

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
DEPARTAMENTO DE ENGENHARIA DE PETRÓLEO

Dissertação Apresentada à
Faculdade de Engenharia Mecânica
Como Requisito Parcial à Obtenção do
Título de Mestre em Engenharia de Petróleo

GOMES

**SIMULAÇÃO NUMÉRICA DE POÇOS HORIZONTAIS
EM RESERVATÓRIOS COM FLUXO MULTIFÁSICO,
USANDO REFINAMENTO LOCAL.**

*Este exemplar corresponde a edição final da tese defendida pelo
Enge. José Adilson Tenório Gomes e aprovada pela comissão julgadora
em 23/11/90*

Autor : José Adilson Tenório Gomes

Orientador : Fernando Rodriguez de la Garza

Prof. Dr. Fernando Rodriguez de la Garza

50/90

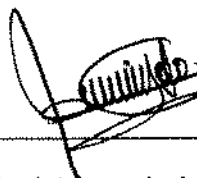
Novembro de 1990

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
DEPARTAMENTO DE ENGENHARIA DE PETRÓLEO

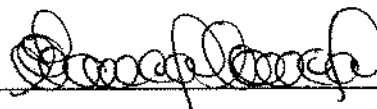
A dissertação "Simulação Numérica de Poços Horizontais em Reservatórios com Fluxo Multifásico, usando Refinamento Local", elaborada por José Adilson Tenório Gomes e aprovada por todos os membros da Banca Examinadora foi aceita pela Sub-Comissão de Pós-Graduação em Engenharia de Petróleo como requisito parcial a obtenção do título de Mestre em Engenharia de Petróleo.

Campinas, 23 de Novembro de 1990

Banca Examinadora



Fernando Rodriguez de la Garza, Ph.D.



Antonio Cláudio de França Corrêa, Ph.D.



Lideniro Alegre, Ph.D.

À Dora

Thiago e Ligia

À minha mãe

E ao meu pai (*em memória*)

AGRADECIMENTOS

Ao Dr. Fernando Rodriguez, pela orientação na elaboração deste trabalho.

Ao professor Dr. Antonio Cláudio e ao Dr. Osvaldo Pedrosa pelos comentários sobre o desenvolvimento inicial do trabalho.

Aos colegas de trabalho da RPSE/DIRES pela oportunidade de participação e o apoio durante o curso.

À Walter Pereira Formosinho por tornar possível a participação do mestrado.

Ao colega Celso Cesar Moreira Branco por sua valiosa contribuição.

Aos colegas de mestrado, professores e funcionários do CEPETRO pelo apoio necessário ao bom andamento do curso.

À todas as pessoas que direta ou indiretamente contribuíram para a realização desta tese.

Conteúdo

1	INTRODUÇÃO	1
2	REVISÃO BIBLIOGRÁFICA	3
2.1	Poço Horizontal	3
2.2	Refinamento Local	6
3	DEFINIÇÃO DO PROBLEMA	9
4	FORMULAÇÃO DO MODELO NUMÉRICO	11
4.1	Formulação Geral	11
4.1.1	Condições Iniciais	12
4.1.2	Condições de Contorno	13
4.2	Discretização das Equações de Fluxo	13
4.2.1	Termo de Acumulação	13
4.2.2	Termo de Fluxo	14
4.2.3	Tratamento das Interfaces	19
4.3	Equações de Fluxo na Forma de Diferenças	21
5	SOLUÇÃO DO MODELO NUMÉRICO	23
5.1	Descrição do Problema	23
5.2	Solução do Sistema não-Linear de Equações	24
5.3	Ordenação da Malha e Solução da Matriz	25
5.3.1	Modelo Totalmente Acoplado	25
5.3.2	Modelo Desacoplado	29
5.4	Solução do Sistema Linear de Equações	33
5.5	Condições de Produção	34
5.6	Derivadas	36
5.7	Controle Automático do Tempo	36

6	VALIDAÇÃO DO MODELO	38
6.1	Solução Analítica	38
6.2	Simulação dos Testes	43
6.2.1	Comportamento da Pressão no Drawdown	43
6.2.2	Determinação da Permeabilidade	46
7	ANÁLISE DOS RESULTADOS	54
7.1	Modelo Totalmente Acoplado	54
7.1.1	Comparação entre os Métodos de Solução	54
7.1.2	Refinamento Radial	56
7.1.3	Refinamento Angular	57
7.2	Comparação entre o Acoplado e os Desacoplados	58
8	APLICAÇÃO DO MODELO	72
9	CONCLUSÕES E RECOMENDAÇÕES	81
10	REFERÊNCIAS	83
10.1	Poços Horizontais	83
10.2	Refinamento local	85
11	APÊNDICE A - Derivadas das Equações de Fluxo	87
12	APÊNDICE B - Dados Utilizados na Simulação	102
13	APÊNDICE C - Resultados	105
13.1	Poço com 300m de comprimento	105
13.2	Poço com 100m de comprimento	112
14	APÊNDICE D - Programas Computacionais	120

Lista de Figuras

3.1	Representação do reservatório	10
4.1	Representação das linhas de fluxo próximo ao poço	14
4.2	Bloco cilíndrico para cálculo da posição do ponto nodal	18
5.1	Modelo físico e sistema de coordenadas	24
5.2	Ordenação da malha	26
5.3	Matriz Jacobiana do sistema totalmente acoplado	28
5.4	Matriz Jacobiana referente a malha cartesiana	30
5.5	Matriz Jacobiana referente a malha cilíndrica	30
5.6	Fluxogramas para os métodos desacoplados	31
5.7	Representação dos termos de fluxo	32
6.1	Representação do modelo físico	39
6.2	Representação das linhas de fluxo dos regimes do transiente	39
6.3	Comparação entre a solução analítica e a numérica	45
6.4	Gráfico de Δp vs $\ln t$, para o radial inicial	50
6.5	Gráfico de Δp vs $\ln t$, para o pseudo radial tardio - $k = 1.97 \times 10^{-14} m^2$	51
6.6	Gráfico de Δp vs $\ln t$, para o pseudo radial tardio - $k = 9.87 \times 10^{-15} m^2$	52
6.7	Gráfico de Δp vs $t^{1/2}$, para o linear tardio	53
7.1	Fator de sobre-relaxação versus tempo	60
7.2	Refinamento radial (caso 1) - p_{wf} versus tempo	61
7.3	Refinamento radial (caso 1) - Q_o versus tempo	62
7.4	Refinamento radial (caso 2) - p_{wf} versus tempo	63
7.5	Refinamento radial (caso 2) - Q_o versus tempo	64
7.6	Refinamento radial (caso 2) - Q_w versus tempo	65
7.7	Refinamento radial (caso 2) - RWO versus tempo	66
7.8	Refinamento angular - p_{wf} versus tempo	67
7.9	Refinamento angular - Q_o versus tempo	68
7.10	Comparação entre os modelos - p_{wf} versus tempo	69

7.11	Comparação entre os modelos - Q_o versus tempo	70
7.12	Comparação entre os modelos - Q_w versus tempo	71
8.1	Posição do poço - p_{wf} versus tempo	74
8.2	Posição do poço - Q_o versus tempo	75
8.3	Posição do poço - Q_w versus tempo	76
8.4	Comprimento do poço - p_{wf} versus tempo	77
8.5	Comprimento do poço - Q_o versus tempo	78
8.6	Comprimento do poço - Q_w versus tempo	79
8.7	Perfil vertical de saturação de água	80

Lista de Tabelas

6.1	Parâmetros usados na validação	44
6.2	Características do reservatório e malha	46
6.3	Duração, em horas, dos regimes transiente.	46
7.1	Parâmetros Usados na Simulação	55
B.1	Propriedades PVT do óleo saturado	102
B.2	Propriedades PVT do óleo sub-saturado	102
B.3	Propriedades PVT da água	102
B.4	Propriedades PVT do gás	103
B.5	Permeabilidades relativas óleo - água	103
B.6	Curva de pressão capilar óleo - água	104

LISTA DE SÍMBOLOS

A	= Área da Seção Transversal (m^2)
a	= Dimensão do reservatório perpendicular ao poço (m)
B	= Fator volume de formação (m^3_{res}/m^3_{std})
b	= Dimensão de reservatório paralela ao poço (m)
c	= Compressibilidade (Pa^{-1})
d	= Menor dimensão: poço/limite do reservatório (m)
D	= Maior dimensão: poço/limite do reservatório (m)
D	= Profundidade em relação a um plano de referência (m)
$\vec{e}$	= Vetor unitário
FF	= Fator de forma
h	= Espessura da formação (m)
k	= Permeabilidade (m^2)
L	= Comprimento do poço (m)
m	= Massa por unidade de volume (kg/m^3)
$\dot{m}$	= Fluxo de massa ($kg/m^3/s$)
n	= Número de blocos
p	= Pressão (Pa)
Q	= Vazão em condições de superfície (m^3_{std}/d)
q	= Vazão em condições de reservatório (m^3/d)
$\tilde{q}$	= Termo fonte ($kg/m^3/s$)
R	= Razão de solubilidade (m^3_{std}/m^3_{std})
r	= Raio (m)
S	= Saturação (fr)
s	= Skin
t	= Tempo (s)
T	= Transmissibilidade
$\vec{u}$	= Velocidade média (m/s)
V	= Volume (m^3)
w	= Ponderador ou fator de sobre-relaxação

LETRAS GREGAS

Δ	= Operador de diferenças
∂	= Operador diferencial parcial
∇	= Operador divergente
ϕ	= Porosidade (<i>fr</i>)
Φ	= Potencial de fluxo
μ	= Viscosidade (<i>Pa.s</i>)
ρ	= Massa Específica (<i>kg/m³</i>)
γ	= Peso Específico (<i>kgf/m³</i>)
θ	= Direção angular

ÍNDICES INFERIORES

<i>e</i>	= Externo
<i>end</i>	= Final de um regime
<i>l</i>	= Fase: óleo, água ou gás
<i>o</i>	= Óleo
<i>p</i>	= Poroso
<i>ph</i>	= Poço horizontal
<i>r</i>	= Rocha ou relativa ou direção radial
<i>start</i>	= Início de um regime
<i>t</i>	= Total
<i>w</i>	= Água ou poço
<i>x, i</i>	= Direção <i>x</i>
<i>y, j</i>	= Direção <i>y</i>
<i>z, k</i>	= Direção <i>z</i>

ÍNDICES SUPERIORES

<i>m</i>	= Iteração anterior do sistema linear de equações
<i>m + 1</i>	= Iteração atual do sistema linear de equações
<i>n</i>	= <i>Time step</i> anterior
<i>n + 1</i>	= <i>Time step</i> atual
ν	= Iteração anterior do sistema não linear de equações
$\nu + 1$	= Iteração atual do sistema não linear de equações
*	= Nível de iteração intermediário

RESUMO

Com o aprimoramento tecnológico dos métodos de perfuração e completção e a necessidade de otimização do volume de óleo recuperável, nota-se um crescente interesse na perfuração de poços horizontais para a drenagem de reservatórios de petróleo. Estes poços são altamente atrativos, principalmente para reservatórios com pequena espessura de óleo e presença de aquífero e/ou capa de gás, para os quais são grandes os riscos de formação de cones de água e/ou gás, quando se utilizam poços verticais para sua drenagem.

O objetivo deste trabalho, é o desenvolvimento de uma técnica de simulação numérica, na qual possa ser utilizada uma geometria de refinamento local para o poços horizontais, que represente adequadamente a geometria do fluxo multifásico ao redor do poço.

Para melhor representar a geometria do fluxo, considera-se que nas proximidades do poço, o fluxo é predominantemente radial, tendendo para linear, conforme se afasta do poço. Então, para a representação numérica utiliza-se uma malha cilíndrica nas proximidades do poço e uma malha cartesiana para a região do reservatório mais afastada do poço.

Foram desenvolvidos dois simuladores, com as equações que regem o fluxo dos fluidos no meio poroso, discretizadas numericamente por diferenças finitas, totalmente implícito, sendo o primeiro considerando um sistema radial/cartesiano acoplado e resolvidos simultaneamente. O segundo considera os dois sistemas desacoplados, com o acoplamento radial/cartesiano feito num *time-step* ou numa iteração Newtoniana.

A validação foi feita através da comparação dos resultados numéricos com as soluções analíticas desenvolvidas para análises de testes para o regime transiente.

São feitas análises entre os métodos de acoplamento, entre os métodos de solução do sistema linear de equações e são apresentados alguns exemplos de aplicação dos modelos.

ABSTRACT

Recent developments in completion and drilling technology have increased the interest of the oil industry in horizontal wells. Such wells are primarily indicated for thin oil reservoirs associated with a gas cap or a bottom water zone, where vertical wells have productivity impaired by the formation of water or gas cones.

The main purpose of this work is to present a finite difference numerical simulator for horizontal wells, based on the use of a local refinement technique. The radial multiphase flow close to the well is more realistically represented through the use of a cylindrically oriented grid blocks. The outer region of the reservoir, where the flow regime is predominately linear, is modelled by regular cartesian blocks.

The connection between the well and reservoir regions was implemented considering two approaches: in the coupled approach the radial/cartesian system is solved simultaneously, while in the uncoupled the reservoir region is solved first, followed by the solution of the well region at each time-step or at each Newtonian iteration.

The validation of the model was done through comparison of the numerical results with analytical ones in the transient flow regimes.

The coupling and matrix solution methods as well as some results of the simulation runs are discussed in this work.

1 INTRODUÇÃO

A crescente valorização do petróleo, a partir da década de setenta, exigiu que as companhias petrolíferas de todo o mundo, investissem mais em pesquisas visando a otimização da produção e recuperação final dos campos de petróleo.

Como consequência, houve grande avanço tecnológico em diversas áreas, principalmente na perfuração e completação de poços direcionais, permitindo que se tornasse cada vez mais viável a perfuração, completação e utilização de poços horizontais para a produção de petróleo. A idéia da utilização de poços horizontais não é nova, pois até os anos 50, mais de uma centena já haviam sido perfurados, mas com índices de sucesso tanto para a perfuração quanto para a produção pouco satisfatórios. Somente a partir do início dos anos 80 é que os poços horizontais começaram a ter resultados mais promissores.

Os princípios físicos que regem o fluxo de fluidos no meio poroso mostram que os poços horizontais oferecem muitas vantagens sobre os verticais. Dentre estas, podemos destacar a grande diferença entre a produtividade dos dois tipos de poços, pois esta depende da extensão do poço em contato com a formação produtora, que para os poços horizontais não é limitada pela espessura da formação como é para os poços verticais. A recuperação final é maior, devido a melhor eficiência de varrido. Para reservatórios com grande extensão areal, o número de poços necessário para sua drenagem é bem menor. Este aspecto é muito interessante para campos *offshore*, explotados com poços perfurados a partir de plataformas. Devido a grande extensão atingida pelo poço horizontal, há maior contribuição para um melhor conhecimento geológico lateral do reservatório, pois em poços verticais isto fica restrito a um ponto. Os poços horizontais podem também oferecer soluções para problemas da produção de óleo e gás em reservatórios onde os poços verticais convencionais teriam resultados poucos satisfatórios. Como exemplo, podemos citar: reservatórios com problemas de formação de cones de água ou gás, reservatórios com fraturas verticais, reservatórios com baixa permeabilidade ou com anisotropia e em reservatórios com baixa eficiência de varrido ou com baixa injetividade.

Naturalmente que todas estas vantagens tem seus limites e o projeto que servirá

de base para a escolha do tipo de poço mais adequado, deverá ser cercado de todos os cuidados, mesmo porque o custo de um poço horizontal ainda é maior que um poço vertical ou direcional. Assim, para que todas as variáveis possam ser cuidadosamente analisadas, minimizando os riscos, muitas pesquisas tem sido feitas. No início, estas pesquisas eram voltadas mais para a determinação de fórmulas analíticas para a previsão da produtividade dos poços horizontais e sua comparação com os verticais. Mais recentemente, com o grande avanço da informática, muito se tem estudado em relação a poços horizontais. Também as novas técnicas em solução de sistemas matriciais tem permitido avanços na área de refinamento de malha e decomposição de domínios, que tem aplicação direta em estudos de poços horizontais.

Este trabalho tem por principal objetivo, a utilização de técnicas de refinamento de malha, aplicadas a diferenças finitas para o desenvolvimento de um simulador que possa representar adequadamente a geometria de fluxo no reservatório drenado por um poço horizontal. Também, estudar as diversas formas de solução dos sistemas gerados e diferentes formas de acoplamento entre as malhas, para que possam ser aplicadas com diferentes objetivos, seja para o estudo de poço ou para o estudo global de um reservatório.

2 REVISÃO BIBLIOGRÁFICA

2.1 Poço Horizontal

A idéia da utilização de poços horizontais para a produção de petróleo não é recente, já que em algumas publicações ^(3,14) encontramos referências de que até os anos 50 e 60, mais de uma centena de poços horizontais já haviam sido perfurados, principalmente na URSS, mas com resultados até então pouco satisfatórios em termos de produção de petróleo. Até esta época, os poucos estudos para a análise do comportamento dos poços horizontais eram realizados em laboratórios, com o auxílio de modelos analógicos elétricos e se concentravam principalmente na comparação entre a produtividade do poço horizontal com a do vertical. Devido as limitações físicas dos modelos utilizados, poucas variáveis podiam ser examinadas, entre elas, o comprimento do poço e a espessura da formação. É citado ^(7,14) que nesta mesma época foi desenvolvida por *Merkulov, V.P. (1958)*, a primeira fórmula analítica para o cálculo do IP de poços horizontais.

A partir do final dos anos 70, devido à crescente valorização do petróleo, iniciou-se uma incessante corrida da indústria petrolífera, para a descoberta de novas reservas e otimização da produção e recuperação final dos campos em produção. Como consequência, houve um grande avanço em diversas áreas, principalmente nas técnicas de perfuração, completação e perfilagem de poços direcionais, tornando-se a partir daí, tecnicamente viável, a utilização de poços horizontais para a produção de petróleo. É a partir de então que começam a ser publicados vários artigos sobre a tecnologia de poços horizontais.

A partir de 1978 - principalmente com a Elf Aquitaine e o Instituto Francês de Petróleo, que financiados pela comunidade econômica européia, iniciaram estudos para a perfuração de quatro poços horizontais - é que começaram a ser publicados vários artigos sobre o comportamento de poços horizontais. Giger ⁽¹⁾ apresentou fórmulas analíticas para o cálculo do IP de poços horizontais e vazão crítica para a formação de *cristas* de água. Apresentou também os critérios para a escolha entre poços horizontais e verticais conven-

cionais. Critérios que levavam em consideração as dimensões do reservatório, mecanismos de manutenção de pressão, anisotropia e características dos fluidos. Em outro artigo Giger ⁽²⁾ analisou as vantagens da produção através de poços horizontais em reservatórios com heterogeneidades tais como: variações da permeabilidade lateral e vertical e com fraturas. Analisou também, os efeitos dos mecanismos de alimentação e o comprimento do poço horizontal. Após os primeiros cinco anos de produção dos poços perfurados pela Elf Aquitaine e o Instituto Francês de Petróleo, Reiss ⁽³⁾ apresentou os principais resultados, comparando-os com as análises teóricas desenvolvidas para o estudo do IP, eficiência de varrido e produção em reservatórios fraturados.

Com a utilização de um modelo hodográfico, Giger ⁽⁴⁾ desenvolveu fórmulas analíticas para o cálculo da deformação vertical dos contatos óleo/água e gás/óleo e a vazão crítica para a formação de *crista* de água. Aplicou as fórmulas para reservatórios com capa de gás e aquífero não atuante, influxo de água de fundo e lateral. Chaperon ⁽⁵⁾ fez um estudo das vantagens da produção com poços horizontais em reservatórios com camadas isoladas de baixa permeabilidade. Utilizando as fórmulas analíticas desenvolvidas por Giger, Kacher ⁽⁷⁾ estudou o problema da produtividade dos poços horizontais em reservatórios fraturados e com anisotropia. Joshi ⁽¹⁰⁾ desenvolveu equações para o IP de poços horizontais e fez estudos para determinação da influência da anisotropia do reservatório, espessura, área de drenagem do poço e da excentricidade na produtividade. Estudou também a tendência da formação de *cristas* de gás e água. Babu et al ⁽¹⁵⁾, mantendo a forma inicial proposta por Giger, desenvolveram o cálculo do IP para reservatórios limitados com poço parcialmente penetrante.

Os estudos desenvolvidos até então, se prendiam ao desenvolvimento de fórmulas analíticas, sem referências à utilização de simuladores numéricos para o estudo do comportamento de poços horizontais. Sherrard ⁽⁶⁾ apresentou um estudo para poços horizontais no qual fez uso de um simulador *black-oil*, totalmente implícito com o poço horizontal sendo representado por vários blocos, numa região em que a malha tinha um grande refinamento, que se estendia por todo o reservatório, em todas as direções. Uma análise

dos cuidados na utilização de fórmulas para o cálculo do IP aplicadas para simuladores numéricos convencionais por diferenças finitas, foi feita por Lee ⁽⁸⁾. Para evitar resultados errados recomendou um grande refinamento e que o poço fosse representado por vários blocos. Kossack ⁽⁹⁾ também usou um simulador *black-oil* totalmente implícito para simular o comportamento da produção através de poços horizontais, em reservatórios com várias fácies de diferentes permeabilidades. Chang ⁽¹⁴⁾ usou o BOAST para fazer a simulação do poço horizontal ou inclinado, através da decomposição do poço em três componentes centrados e paralelos às faces dos blocos e o cálculo do IP para cada componente usando o método das imagens.

Todas as fórmulas desenvolvidas para o cálculo do IP assumiram até agora o comportamento do regime permanente, ou seja, davam estimativa para a produtividade ou injetividade para poços, para os quais, a queda de pressão entre o poço e a formação permanece constante, tais como nos processos de injeção de água.

O comportamento do fluxo de um fluido incompressível em poços horizontais no regime transiente, para reservatórios com espessura finita e infinito horizontalmente foi estudado por Lara ⁽¹²⁾ e Clonts ⁽¹⁷⁾. Goode e Thambynayagam ⁽¹¹⁾ descreveram as repostas de pressão no poço horizontal para um fluido levemente compressível num reservatório semi-infinito: limitado verticalmente e na direção perpendicular ao poço, desprezando o efeito gravitacional. Ozkan et al ⁽¹⁶⁾ desenvolveram curvas tipos para testes de poços horizontais em reservatório infinito. Odeh e Babu ⁽¹³⁾ apresentaram análises do comportamento no regime transiente para reservatório limitado em todas as direções e para poço parcialmente ou totalmente penetrante, localizado paralelamente a duas faces, em qualquer ponto do reservatório. Também para fluido levemente compressível sem o efeito gravitacional.

2.2 Refinamento Local

Durante a simulação numérica de reservatórios é desejável que as regiões em que ocorrem fortes variações nas características do fluxo, sejam representadas por uma malha com refinamento tal, que possa mostrar com precisão estas variações. Estas regiões na maioria dos casos são as regiões dos poços, que é uma região relativamente pequena quando comparada com todo o reservatório. Portanto, é interessante que somente esta região tenha um grau de refinamento maior, deixando as outras partes de reservatório representada por blocos maiores, o que não provoca nenhuma perda de precisão e contribui enormemente para o aumento da velocidade de simulação e economia de memória computacional.

Na simulação numérica de reservatórios com discretização por diferenças finitas e uso de coordenadas cartesianas, devem ser tomados alguns cuidados na implementação de refinamentos local, para que não ocorram erros que comprometam os resultados. Um dos primeiros pesquisadores nesta área foi Rosenberg ⁽¹⁹⁾, que apresentou um método de refinamento local onde prevê quatro diferentes configurações para a aplicação de diferenças finitas. Mostra que o uso destas configurações são efetivas e mais econômicas. Quandalle e Besset ⁽²⁰⁾, com o objetivo de melhor representar reservatórios com falhas, discordâncias geológicas e outras anomalias, fizeram a aplicação do refinamento local em vários níveis, com o uso de múltiplos *time-steps* para diferentes grupos de blocos, num simulador *black-oil* implícito na pressão e na saturação. Concluíram que o uso deste procedimento proporcionava resultados bem próximos dos obtidos com a utilização de uma malha totalmente refinada, mas com grande redução no tempo de computação.

Heinemann et al ⁽²¹⁾ sugeriram a aplicação do refinamento local dinâmico em simulador de múltiplas aplicações, com o objetivo de fazer uma grande redução no número de blocos necessários à representação do reservatório. As equações foram formuladas para diferentes graus de implicitude, que dependiam dos níveis de variação da pressão, saturação e da temperatura do reservatório. Também foi feita a seleção dinâmica do refinamento e do tamanho do *time-step* e uma análise do erro de discretização devido a este

refinamento.

Forsyth e Sammon ⁽²⁰⁾ aplicaram o refinamento local flexível para a simulação de reservatórios com falhas e *pinchouts*, fazendo o uso de diferentes graus de implicitude para diferentes regiões do reservatório. Devido a esparsidade do sistema Jacobiano resultante, este foi resolvido por método iterativo.

A malha híbrida proposta por Pedrosa ⁽²²⁾, representava a região próxima ao poço através de uma malha cilíndrica ou elíptica - a depender das características do reservatório - e a região mais afastada do poço através de uma malha cartesiana. Esta configuração adaptava perfeitamente à geometria do fluxo próximo e longe do poço, podendo dentro da malha cilíndrica, os blocos serem sucessivamente menores, conforme se aproxima do poço. Com esta configuração, desenvolveu simuladores com graus de implicitudes diferentes para as duas regiões; a região do poço totalmente implícita e a região do reservatório num esquema IMPES. As duas regiões foram resolvidas independentemente, sendo as condições dos blocos cartesianos vizinhos aos cilíndricos, usadas como condições de fronteira para os cilíndricos, num processo que poderia ser iterativo ou não. O processo não iterativo traz bons resultados, pois segundo Akbar et al ⁽¹⁸⁾ a convergência num processo iterativo é atingida numa única iteração.

Em outro artigo, Quandalle e Besset ⁽²⁴⁾ analisaram o efeito do refinamento de malha em diferentes níveis, e para eliminar efeitos indesejáveis, propuseram esquemas mais sofisticados para diferenças finitas envolvendo cinco e nove pontos, ao invés de dois, aplicável a qualquer composição de malha tridimensional. Han ⁽²⁵⁾ apresentou um esquema de refinamento dinâmico para qualquer região do reservatório, onde o refinamento é sempre de um bloco para um número par de vizinhos. Wasserman ⁽²⁶⁾ propôs a aplicação do refinamento estático em vários níveis para simuladores tridimensionais, usando para a solução o conceito de decomposição de domínios. Ewing et al ⁽²⁷⁾ discutiram as várias estratégias para o refinamento de malha e o uso da decomposição de domínios. Heine-mann et al ⁽²⁸⁾ propuseram o método de discretização volume finito ou balanço, que fica entre a diferença finita e o elemento finito e se caracteriza pela formação dos blocos pela

ligação entre as linhas perpendiculares aos *grid points*, fazendo com que cada bloco tenha uma geometria irregular e com seis vizinhos. Brand e Heinemann ⁽²⁹⁾, apresentaram novas formas de fatoração da matriz esparsa gerada quando do uso de refinamento local e apresentaram também alguns métodos eficientes de ordenação da malha.

3 DEFINIÇÃO DO PROBLEMA

O principal objetivo deste trabalho, é o estudo das técnicas e desenvolvimento de um simulador numérico, com malha que represente adequadamente a geometria de fluxo, para o estudo do comportamento de reservatórios com produção através de poços horizontais.

As equações de fluxo foram desenvolvidas para a construção de um simulador tridimensional, bifásico (óleo e água), totalmente implícito e utilização do conceito de malha híbrida, ou seja, a região próxima ao poço - onde ocorrem grandes variações de pressão e saturação e onde a geometria do fluxo é predominantemente radial - será representada por uma malha cilíndrica. A região mais afastada do poço será representada por uma malha cartesiana com o acoplamento entre as duas regiões através de dois processos distintos:

- Sistema totalmente acoplado:

Considera as duas malhas sendo resolvidas simultaneamente no mesmo sistema linear de equações, ou seja, no mesmo Jacobiano;

- Sistemas desacoplados:

Considera as malhas resolvidas separadamente, com posterior acoplamento através da transferência dos termos de fluxo em cada *time-step* ou em cada iteração Newtoniana.

São feitas comparações entre os resultados gerados pelos diferentes métodos de acoplamento das regiões e também são feitas análises em relação aos métodos de solução do sistema linear de equações.

O reservatório a ser estudado é ilustrado pela figura 3.1.

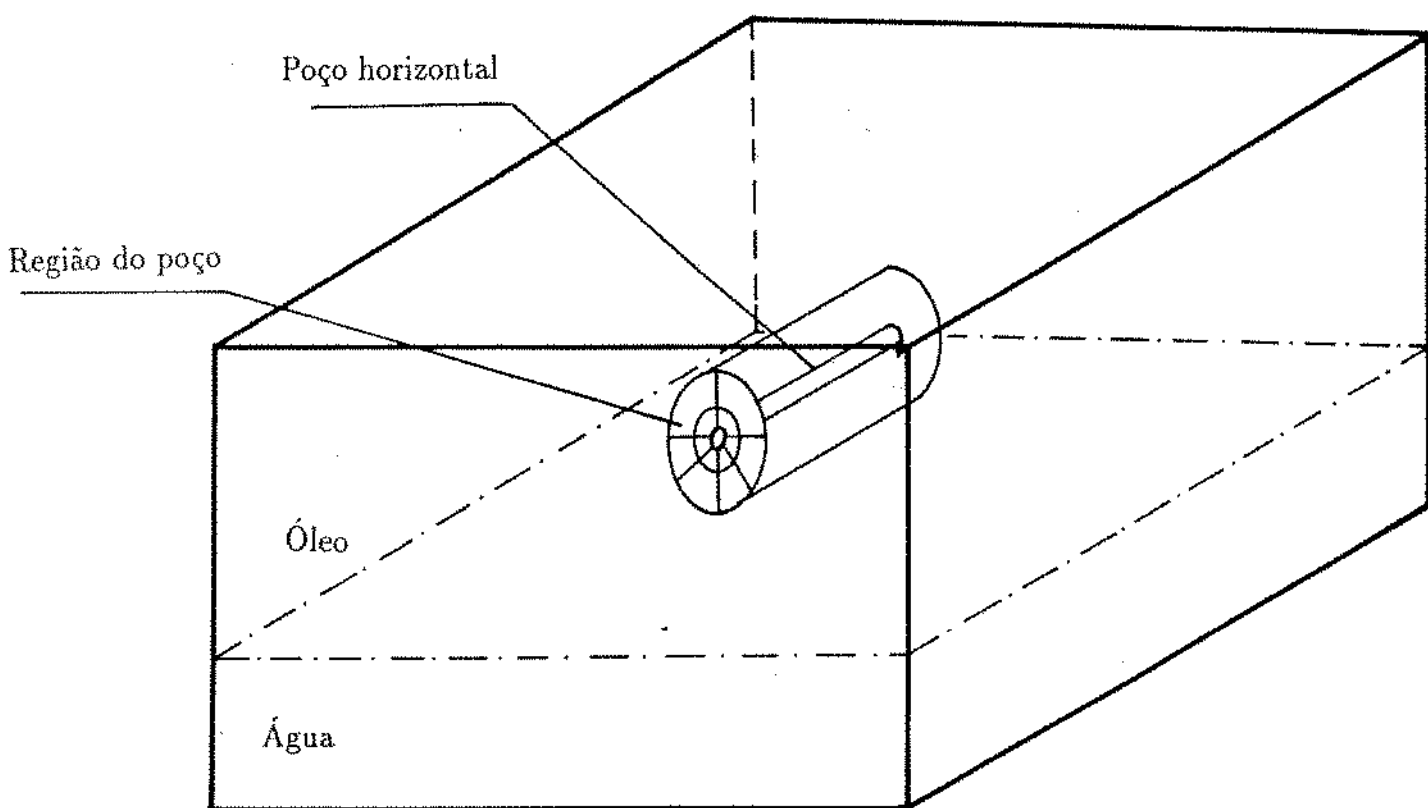


Figura 3.1: Representação do reservatório

4 FORMULAÇÃO DO MODELO NUMÉRICO

4.1 Formulação Geral

O modelo desenvolvido é um simulador *black-oil*, totalmente implícito, tridimensional, bifásico (óleo e água), com as regiões que representam o reservatório e o poço podendo ser totalmente acopladas ou desacopladas.

As equações que regem o fluxo dos fluidos no meio poroso são derivadas a partir da equação da conservação de massa para o fluxo de uma fase,

$$-\nabla \cdot \vec{m} = \frac{\partial m}{\partial t} + \tilde{q} \quad (4.1)$$

onde : $\vec{m}$ = fluxo de massa através da fronteira;

m = massa por unidade de volume;

$\tilde{q}$ = termo fonte.

Fazendo a substituição do fluxo de massa e a massa no meio poroso em termos de densidade média, porosidade e saturação, teremos para a fase l :

$$-\nabla \cdot \rho_l \vec{u}_l = \frac{\partial}{\partial t}(\rho_l \phi S_l) + \tilde{q}_l \quad (4.2)$$

Usando a lei de Darcy para o fluxo multifásico, que é dada por:

$$\vec{u}_l = -k \frac{k_{rl}}{\mu_l} (\nabla p_l - \gamma_l \nabla D) \quad (4.3)$$

e o conceito de fator volume de formação, podemos derivar a equação que rege o fluxo para cada fase, conforme:

$$\nabla \cdot [\lambda_l (\nabla p_l - \gamma_l \nabla D)] = \frac{\partial}{\partial t} \left[\frac{\phi S_l}{B_l} \right] + q_l \quad (4.4)$$

onde, l é o índice que representa a fase (óleo ou água) e λ_l é a mobilidade, que é definida como:

$$\lambda_l = k \frac{k_{rl}}{\mu_l B_l} \quad (4.5)$$

Segundo a formulação apresentada por Pedrosa⁽²²⁾ esta equação pode ser integrada sobre um volume de controle V .

Definindo a diferença de potencial como, $\nabla \Phi_l = (\nabla p_l - \gamma_l \nabla D)$ e admitindo fluxo permanente através das fronteiras, a equação 4.4 pode ser reescrita e integrada como:

$$\int_V \nabla \cdot [\lambda_l \nabla \Phi_l] dV = \int_V \frac{\partial}{\partial t} \left[\frac{\phi S_l}{B_l} \right] dV + q_l \quad (4.6)$$

Usando o teorema do divergente (ou teorema de Gauss), que transforma uma integral de volume em uma integral de superfície, esta equação pode ser reescrita, com o lado esquerdo modificado para uma integral do fluxo de massa sobre uma superfície A , assim:

$$\int_A \lambda_l \nabla \Phi_l \cdot \vec{n} dA = \int_V \frac{\partial}{\partial t} \left[\frac{\phi S_l}{B_l} \right] dV + q_l \quad (4.7)$$

onde : dA = elemento de área atravessada pelo fluxo;

$\vec{n}$ = vetor unitário normal a superfície;

V = volume de controle.

O lado esquerdo da equação acima, é o termo de fluxo e representa o fluxo de massa sobre toda superfície de controle e o lado direito é o termo de acumulação que representa a taxa de expansão e contração do volume de controle.

Para a solução do problema são necessárias as seguintes relações adicionais:

$$\Sigma S_l = S_o + S_w = 1 \quad (4.8)$$

$$P_{cow} = p_o - p_w = f(S_w) \quad (4.9)$$

4.1.1 Condições Iniciais

O reservatório se apresenta inicialmente saturado com óleo e água conata, podendo ou não ter presença de água livre (aquífero). Quando houver aquífero, considera-se que as fases estão em equilíbrio gravitacional e capilar. A pressão original, p_i , é dada no contato óleo/água ou numa profundidade de referência arbitrária.

4.1.2 Condições de Contorno

Como condição de contorno externa considera-se que não há fluxo através das fronteiras externas, ou seja, o reservatório é fechado nas três direções.

Como condição de contorno interna considera-se a produção através de um poço horizontal com condutividade infinita, podendo produzir com vazão de óleo (Q_o) ou pressão de fluxo de fundo (p_{wf}) constante.

4.2 Discretização das Equações de Fluxo

As equações anteriores são discretizadas por diferenças finitas para formulação totalmente implícita, com esquema de bloco centrado com cinco pontos.

4.2.1 Termo de Acumulação

Considerando diferenças regressivas no tempo, o termo de acumulação pode ser aproximado por:

$$\int_V \frac{\partial}{\partial t} \left[\frac{\phi S_l}{B_l} \right] dV = \frac{V_{ijk}}{\Delta t} \Delta_t \left[\frac{\phi S_l}{B_l} \right]_{ijk} \quad (4.10)$$

onde Δ_t é o operador de diferenças no tempo, definido como:

$$\Delta_t \left[\frac{\phi S_l}{B_l} \right] = [(\phi S_l b_l)^{n+1} - (\phi S_l b_l)^n] \quad (4.11)$$

onde n representa o nível de tempo e b_l é o inverso do fator volume de formação B_l , ou seja, $b_l = \frac{1}{B_l}$.

Sabendo que:

$$\phi^{n+1} = \phi^n [1 + c_r (p_o^{n+1} - p_o^n)] \quad (4.12)$$

as equações discretizadas dos termos de acumulação para o óleo e a água, como função das variáveis primárias p_o e S_w , podem ser escritas repectivamente como:

$$\frac{V_{ijk}}{\Delta t} \Delta_t \left[\frac{\phi S_o}{B_o} \right] = \frac{V_{p_{ijk}}}{\Delta t} \{ [1 + c_r(p_o^{n+1} - p_o^n)] (1 - S_w^{n+1}) b_o^{n+1} - (S_o b_o)^n \}_{ijk} \quad (4.13)$$

$$\frac{V_{ijk}}{\Delta t} \Delta_t \left[\frac{\phi S_w}{B_w} \right] = \frac{V_{p_{ijk}}}{\Delta t} \{ [1 + c_r(p_o^{n+1} - p_o^n)] (S_w b_w)^{n+1} - (S_w b_w)^n \}_{ijk} \quad (4.14)$$

Para a equação do óleo foi usada a relação adicional:

$$S_o + S_w = 1 \quad (4.15)$$

4.2.2 Termo de Fluxo

O termo de fluxo é fortemente dependente da geometria da malha, e esta tem que ser levada em conta na discretização das equações. Para a representação do poço horizontal adotou-se a hipótese de que próximo ao poço, as linhas de fluxo são aproximadamente radiais com as linhas equipotenciais circulares, conforme ilustrado na figura 4.1, o que levou à adoção de uma malha híbrida (Pedrosa⁽²²⁾), ou seja, cilíndrica para a região próxima ao poço e cartesiana para regiões mais afastadas.

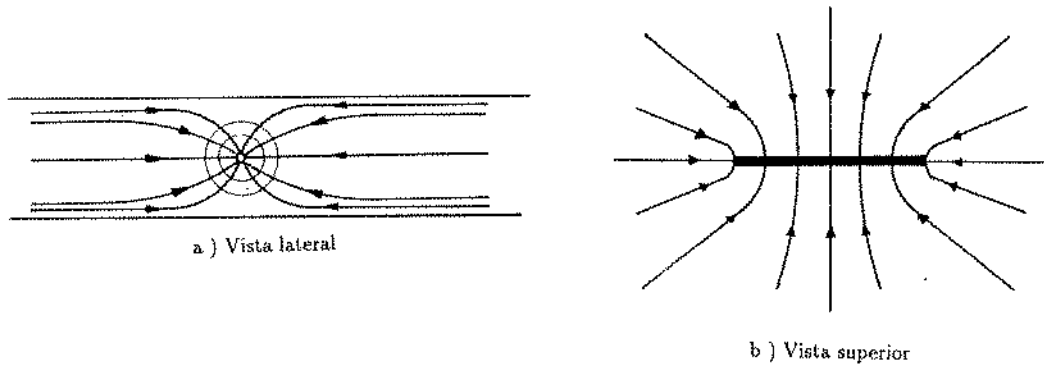


Figura 4.1: Representação das linhas de fluxo próximo ao poço

Portanto, para a discretização do termo de fluxo tem que ser levado em consideração o sistema de coordenadas e a área atravessada pelo fluxo. Para um determinado

volume de controle, este termo representa o somatório do fluxo volumétrico através das faces do bloco, ou seja:

$$\int_A k \frac{k_{rl}}{B_l \mu_l} \nabla \Phi_l dA = \sum_{faces} Q_l \quad (4.16)$$

O potencial de fluxo pode ser decomposto em parcelas referentes a cada direção em cada sistema de coordenadas, ou seja:

Coordenadas cilíndricas

$$\nabla \Phi_l = \frac{\partial \Phi_l}{\partial r} \vec{e}_r + \frac{1}{r} \frac{\partial \Phi_l}{\partial \theta} \vec{e}_\theta + \frac{\partial \Phi_l}{\partial z} \vec{e}_z \quad (4.17)$$

Coordenadas cartesianas

$$\nabla \Phi_l = \frac{\partial \Phi_l}{\partial x} \vec{e}_x + \frac{\partial \Phi_l}{\partial y} \vec{e}_y + \frac{\partial \Phi_l}{\partial z} \vec{e}_z \quad (4.18)$$

onde, $\vec{e}_r$, $\vec{e}_\theta$, $\vec{e}_x$, $\vec{e}_y$ e $\vec{e}_z$ são vetores unitários para cada direção.

Assim quando aplicado a uma determinada direção as parcelas referentes às outras direções se anulam. Por exemplo, para a direção x teremos:

$$Q_{lface} = \int_A k \frac{k_{rl}}{B_l \mu_l} \frac{\partial \Phi_l}{\partial x} dA \quad (4.19)$$

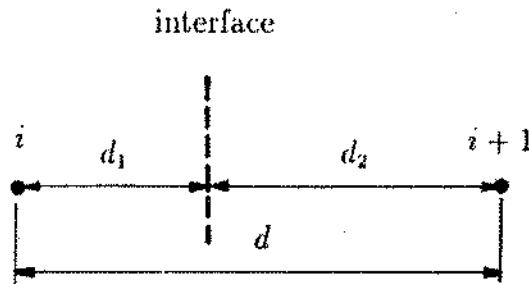
Considerando que $\frac{k_{rl}}{B_l \mu_l}$ pode ser obtido como um valor médio, que dA pode ser determinada de acordo com a geometria e que o termo $k_x \frac{\partial \Phi_l}{\partial x}$ seja função somente da direção considerada a integral pode ser determinada, chegando-se a seguinte forma:

$$Q_{lface} = (FF \frac{k_{rl}}{\mu_l B_l})_{face} \Delta \Phi_l \quad (4.20)$$

O termo $(FF \frac{k_{rl}}{\mu_l B_l})_{face}$, representa a transmissibilidade e cujas parcelas são calculadas para as faces, como a seguir:

- k_{rl} - A permeabilidade relativa para a face é calculada usando o esquema de um ponto a montante, para isto tem que ser considerado o sentido do fluxo das fases;

- μ_l e B_l - A viscosidade e o fator volume de formação na face são determinados através da ponderação dos valores nos pontos nodais dos blocos vizinhos, levando-se em consideração as distâncias destes à face, ou seja:



ponderador : $w = \frac{d_1}{d}$, portanto:

$$\mu_{l_{face}} = (1 - w)\mu_{li} + w\mu_{li+1} \quad (4.21)$$

$$B_{l_{face}} = (1 - w)B_{li} + wB_{li+1} \quad (4.22)$$

As propriedades nos pontos nodais, são funções das pressões nos mesmos.

- FF - Fator de forma, que leva em consideração a geometria da face atravessada pelo fluxo e portanto definidos para cada tipo de malha (cartesiana ou cilíndrica) e para cada direção, conforme a formulação que é apresentada por Pedrosa⁽²²⁾:

– Malha Cilíndrica

* Direção Radial - r

$$FF_{li+1/2} = \left[\frac{\Delta\theta_j \Delta D_k}{\frac{1}{k_{rijk}} \ln \frac{r_{i+1/2}}{r_i} + \frac{1}{k_{ri+1jk}} \ln \frac{r_{i+1}}{r_{i+1/2}}} \right] \quad (4.23)$$

onde k_{rijk} é a permeabilidade absoluta na direção radial.

* Direção Angular - θ

$$FF_{lj+1/2} = \left[\frac{\ln \frac{r_{i+1/2}}{r_{i-1/2}} \Delta D_k}{\frac{\theta_{j+1/2} - \theta_j}{k_{\theta ijk}} + \frac{\theta_{j+1} - \theta_{j+1/2}}{k_{\theta i+1jk}}} \right] \quad (4.24)$$

onde $k_{\theta ijk}$ é a permeabilidade absoluta na direção angular.

* Direção Linear - z

$$FF_{lk+1/2} = \left[\frac{\frac{\Delta \theta}{2} ((r_{i+1/2})^2 - (r_{i-1/2})^2)}{\frac{D_{k+1/2} - D_k}{k_{Dijk}} + \frac{D_{k+1} - D_{k+1/2}}{k_{Dijk+1}}} \right] \quad (4.25)$$

onde k_{Dijk} é a permeabilidade absoluta na direção linear.

– Malha Cartesiana

* Direção - x

$$FF_{li+1/2} = \left[\frac{\Delta Y_j \Delta D_k}{\frac{X_{i+1/2} - X_i}{k_{xijk}} - \frac{X_{i+1} - X_{i+1/2}}{k_{xi+1k}}} \right] \quad (4.26)$$

* Direção - y

$$FF_{lj+1/2} = \left[\frac{\Delta X_i \Delta D_k}{\frac{Y_{j+1/2} - Y_j}{k_{yijk}} - \frac{Y_{j+1} - Y_{j+1/2}}{k_{yij+1k}}} \right] \quad (4.27)$$

* Direção - z

$$FF_{lk+1/2} = \left[\frac{\Delta X_i \Delta Y_j}{\frac{D_{k+1/2} - D_k}{k_{Dijk}} - \frac{D_{k+1} - D_{k+1/2}}{k_{Dijk+1}}} \right] \quad (4.28)$$

para as faces $i - 1/2$, $j - 1/2$ e $k - 1/2$ as fórmulas são similares.

Para a malha cilíndrica é adotado um espaçamento logarítmico, que se caracteriza por ter blocos exponencialmente maiores conforme se afasta do poço. Esta configuração é interessante, pois próximo ao poço, as mudanças de saturação e pressão são mais intensas. Para o cálculo da posição do ponto nodal, considera-se que para o fluxo de um fluido incompressível numa malha cilíndrica, este ponto deve estar localizado no ponto onde a distribuição de pressão para o regime permanente, seja igual a pressão média volumétrica do bloco.

Para o bloco radial, representado pela figura 4.2, a distribuição de pressão para o fluxo permanente é dado por:

$$p - p_{i-1/2} = \frac{q\mu}{\theta kh} \ln\left(\frac{r}{r_{i-1/2}}\right) \quad (4.29)$$

Para o fluxo incompressível no regime permanente a vazão q , através de qualquer seção transversal do bloco é constante. E a pressão média volumétrica é dada por:

$$\bar{p} = \frac{1}{V\phi} \int_V p\phi dV \quad (4.30)$$

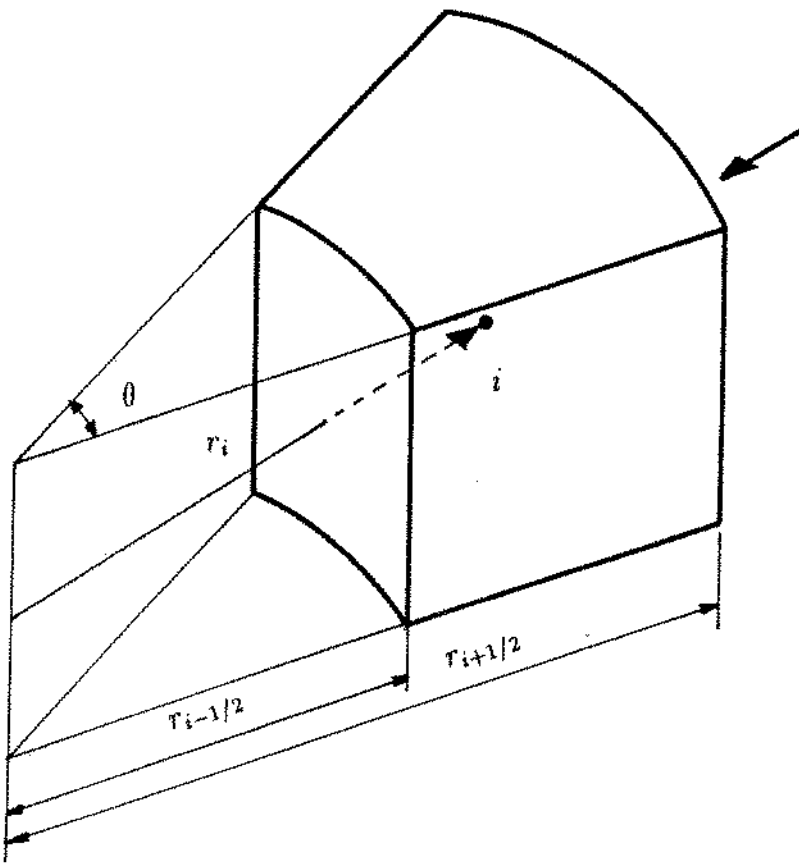


Figura 4.2: Bloco cilíndrico para cálculo da posição do ponto nodal

O volume de um bloco cilíndrico, conforme esquematizado na figura 4.2, é dado por:

$$V = \frac{\theta}{2}(r_{i+1/2}^2 - r_{i-1/2}^2)h \quad (4.31)$$

e

$$dV = \theta h r dr \quad (4.32)$$

Portanto a pressão média volumétrica do bloco pode ser escrita como:

$$\bar{p} = \frac{2}{\theta(r_{i+1/2}^2 - r_{i-1/2}^2)h} \int_{r_{i-1/2}}^{r_{i+1/2}} p \theta h r dr \quad (4.33)$$

Substituindo a equação 4.29 na equação 4.33, teremos:

$$\bar{p} = \frac{2}{\theta(r_{i+1/2}^2 - r_{i-1/2}^2)h} \int_{r_{i-1/2}}^{r_{i+1/2}} [p_{i-1/2} + \frac{q\mu}{\theta kh} \ln \frac{r}{r_{i-1/2}}] \theta h r dr \quad (4.34)$$

que integrada de $r_{i-1/2}$ à $r_{i+1/2}$, resulta em:

$$\bar{p} - p_{i-1/2} = \frac{q\mu}{\theta kh} \left[\frac{r_{i+1/2}^2}{r_{i+1/2}^2 - r_{i-1/2}^2} \ln \frac{r_{i+1/2}}{r_{i-1/2}} - \frac{1}{2} \right] \quad (4.35)$$

se r_i é o ponto nodal, então a pressão (p), neste ponto deve ser igual a pressão média, daí igualando as equações 4.29 e 4.35, teremos a posição deste ponto:

$$r_i = r_{i-1/2} e^{\left[\frac{-\alpha^2}{\alpha^2 - 1} \ln \alpha - \frac{1}{2} \right]} \quad (4.36)$$

onde : $\alpha = \frac{r_{i+1/2}}{r_{i-1/2}}$

e a posição das faces dos blocos é dada por:

$$r_{i+1/2} = r_w e^{i\Delta\rho} \quad (4.37)$$

onde, $\Delta\rho = \frac{\ln \frac{r_e}{r_w}}{N.Blocos}$

4.2.3 Tratamento das Interfaces

A malha cilíndrica possui o mesmo volume e o mesmo número de blocos na direção z , que os blocos cartesianos atravessados pelo poço. Na direção radial o número mínimo de blocos são 2 e estes são exponencialmente crescentes a partir do poço. Na direção angular o número mínimo de blocos são 4.

Para o modelo totalmente acoplado pode haver, na direção radial (r) e linear (z), comunicação entre a malha cilíndrica e a cartesiana. A única diferença que ocorre nas equações de fluxo dos blocos de interfaces são os fatores de forma. Estes são dados por:

1. Direção Radial - r

$$FF_{i+1/2} = \left[\frac{\Delta\theta\Delta D}{\frac{1}{k_{rnz}} \ln\left(\frac{r_s}{r_{nz}}\right) + \frac{1}{k_{rcv}} \ln\left(\frac{d_{cx}}{r_e}\right)} \right] \quad (4.38)$$

2. Direção Linear - z

$$FF_{lk+1/2} = \left[\frac{\Delta\theta((r_{i+1/2})^2 - (r_{i-1/2})^2)}{\Delta D(\frac{1}{k_{zrv}} - \frac{1}{k_{zcv}})} \right] \quad (4.39)$$

Nestas equações temos:

- r_e = raio externo da malha cilíndrica;
- r_{nx} = ponto nodal do bloco mais externo da malha cilíndrica;
- d_{cv} = distância entre ponto nodal do bloco mais externo da malha cilíndrica e o do bloco cartesiano vizinho;
- k_{rnx} = permeabilidade absoluta do bloco cilíndrico mais externo na direção radial;
- k_{rcv} = permeabilidade absoluta do bloco cartesiano na direção do bloco radial vizinho;
- k_{znx} = permeabilidade absoluta do bloco cilíndrico mais externo na direção linear;
- k_{zcv} = permeabilidade absoluta do bloco cartesiano na direção do bloco radial vizinho;

Para os modelos desacoplados não existem termos de interfaces, pois a comunicação cartesiano/cilíndrico é determinada pela substituição do termo de fluxo referente as faces externas dos blocos cilíndricos, pelos termos de fluxo dos blocos cartesianos vizinhos, calculados numa iteração ou num *time step*. No próximo capítulo serão dados mais detalhes sobre este acoplamento.

4.3 Equações de Fluxo na Forma de Diferenças

De acordo com as discretizações do termo de acumulação e termo de fluxo, feitas nas seções anteriores, as equações para o fluxo do óleo e água no meio poroso, num esquema totalmente implícito na pressão e na saturação, podem ser escritas da seguinte forma:

Equação para o óleo:

$$\sum_m \{ T_{o_{m+1/2}}^{n+1} [(p_{m+1}^{n+1} - p_m^{n+1}) - \gamma_{o_{m+1/2}}^{n+1} (D_{m+1} - D_m)] + \\ - T_{o_{m-1/2}}^{n+1} [(p_m^{n+1} - p_{m-1}^{n+1}) - \gamma_{o_{m-1/2}}^{n+1} (D_m - D_{m-1})] \} = \\ \frac{V_{p_{ijk}}}{\Delta t} ([1 + c_r(p_o^{n+1} - p_o^n)](1 - S_w^{n+1})b_o^{n+1} - (S_o b_o)^n)_{ijk} + q_o \quad (4.40)$$

Equação para a água:

$$\sum_m \{ T_{w_{m+1/2}}^{n+1} [(p_{m+1}^{n+1} - p_m^{n+1}) - (P_{cow_{m+1}}^{n+1} - P_{cow_m}^{n+1}) - \gamma_{w_{m+1/2}}^{n+1} (D_{m+1} - D_m)] + \\ - T_{w_{m-1/2}}^{n+1} [(p_m^{n+1} - p_{m-1}^{n+1}) - (P_{cow_m}^{n+1} - P_{cow_{m-1}}^{n+1}) - \gamma_{w_{m-1/2}}^{n+1} (D_m - D_{m-1})] \} = \\ \frac{V_{p_{ijk}}}{\Delta t} ([1 + c_r(p_o^{n+1} - p_o^n)](S_w b_w)^{n+1} - (S_w b_w)^n)_{ijk} + q_w \quad (4.41)$$

Nas equações acima, m representa as três direções (i, j, k), o *time step* anterior (*old time step level*), é dado por n e o *time step* atual (*new time step level*), por $n + 1$. A transmissibilidade e o peso específico dos fluidos são calculados nas faces dos blocos.

Estas equações podem ser escritas simplificadaamente utilizando-se a notação definida por Aziz e Settari³⁰, da seguinte forma:

$$\Delta [T(\Delta p_l - \gamma_l \Delta D)]_{ijk}^{n+1} = \frac{V_{ijk}}{\Delta t} \Delta_t \left[\frac{\phi S_l}{B_l} \right]_{ijk} \quad (4.42)$$

Onde o primeiro membro significa:

$$\begin{aligned}
\Delta[T_l(\Delta p_l - \gamma_l \Delta D)]_{ijk}^{n+1} = & \Delta_x[T_{xl}(\Delta_x p_l - \gamma_l \Delta_x D)]_{ijk}^{n+1} + \\
& \Delta_y[T_{yl}(\Delta_y p_l - \gamma_l \Delta_y D)]_{ijk}^{n+1} + \\
& \Delta_z[T_{zl}(\Delta_z p_l - \gamma_l \Delta_z D)]_{ijk}^{n+1}
\end{aligned} \tag{4.43}$$

Para uma variável U , o operador de diferenças centrais é definido como:

$$(\Delta_x U)_i = U_{i+1/2} - U_{i-1/2}$$

Em termos de p_o e S_w a equação 4.42 pode ser escrita como:

$$\Delta[T_o(\Delta p_o - \gamma_o \Delta D)]_{ijk}^{n+1} = \frac{V_{ijk}}{\Delta t} \Delta_t [\phi b_o (1 - S_w)]_{ijk} \tag{4.44}$$

$$\Delta[T_w(\Delta p_o - \Delta P_{cwo} - \gamma_w \Delta D)]_{ijk}^{n+1} = \frac{V_{ijk}}{\Delta t} \Delta_t [\phi b_w S_w]_{ijk} \tag{4.45}$$

Estas equações escritas para todos blocos ijk , ($i = 1, 2, \dots, nx$, $j = 1, 2, \dots, ny$ e $k = 1, 2, \dots, nz$) no nível de iteração $n+1$ geram um sistema não-linear de $2 \times nx \times ny \times nz$ equações com $2 \times nx \times ny \times nz$ incógnitas.

Os métodos de solução do sistema não linear de equações, assim como os métodos de solução do sistema já linearizado serão abordados no próximo capítulo.

5 SOLUÇÃO DO MODELO NUMÉRICO

O objetivo deste capítulo é fazer a descrição do problema a ser resolvido e todas as hipóteses utilizadas para sua solução.

5.1 Descrição do Problema

As equações desenvolvidas e discretizadas anteriormente serão aplicadas para o estudo do comportamento de um reservatório produzindo através de um poço horizontal num modelo com as seguintes características:

- Totalmente implícito;
- Tridimensional;
- Bifásico (óleo e água);
- Poço com condutividade infinita e com fluxo nas pontas desprezível;
- Produção com vazão ou pressão de fluxo de fundo especificada;
- Uso de malha híbrida, com região do poço em coordenadas cilíndricas e a região do reservatório em coordenadas cartesianas;
- Regiões totalmente acopladas ou desacopladas;
- Consideração do efeito gravitacional;
- Consideração do sentido de fluxo;
- Solução do sistema não linear de equações pelo método de Newton-Raphson;
- Solução do sistema linear de equações por método direto e iterativo.

A esquematização do modelo físico e do sistema de coordenadas utilizadas são apresentados na figura 5.1.

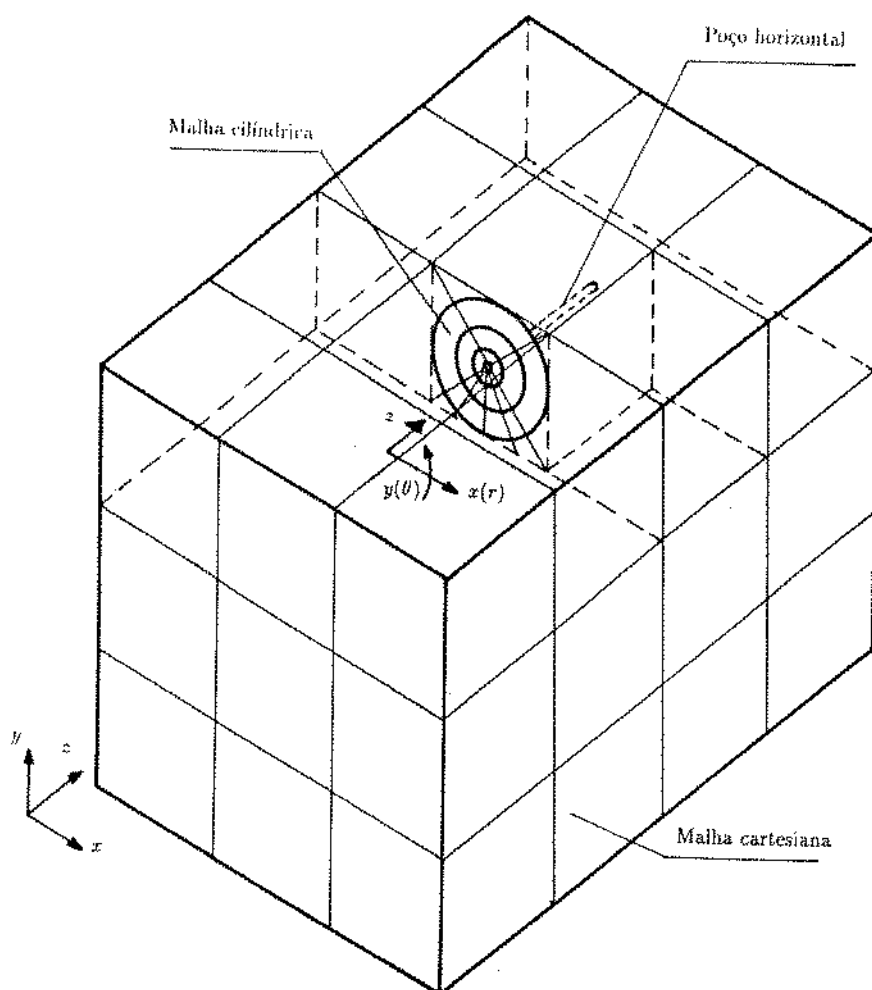


Figura 5.1: Modelo físico e sistema de coordenadas

5.2 Solução do Sistema não-Linear de Equações

Para o esquema totalmente implícito, o método de Newton Raphson será aplicado para resolver o sistema não linear de equações, nas incógnitas p_o e S_w , que é gerado com a aplicação das equações desenvolvidas anteriormente a todos os blocos que formam o reservatório.

O esquema iterativo do Newton-Raphson é gerado a partir da expansão das funções de resíduos de cada fase, F_l ($l = o, w$), em torno da iteração (ν), e retenção dos termos de menor ordem.

Genéricamente o método de Newton-Raphson pode ser escrito como:

$$-\vec{F}_l^{(\nu+1)} = \left(\frac{\partial F_l}{\partial x} \right)^\nu \delta \vec{x}^{\nu+1} \quad (5.1)$$

onde:

- F_l são as funções de resíduos dadas pelas equações;

$$F_{oijk}^{n+1} = \Delta [T_o(\Delta p_o - \gamma_o \Delta D)]_{ijk}^{n+1} - \frac{V_{ijk}}{\Delta t} \Delta_t [\phi b_o(1 - S_w)]_{ijk} - q_o = 0 \quad (5.2)$$

$$F_{wijk}^{n+1} = \Delta [T_w(\Delta p_o - \Delta P_{cwo} - \gamma_w \Delta D)]_{ijk}^{n+1} - \frac{V_{ijk}}{\Delta t} \Delta_t [\phi b_w S_w]_{ijk} - q_w = 0 \quad (5.3)$$

- $\left(\frac{\partial F}{\partial x} \right)^\nu$, representa a matriz Jacobiana, que é formada pelas derivadas das funções resíduos, em relação as variáveis primárias (p_o e S_w) para todos os blocos e seus vizinhos. Para o nosso problema, que é tridimensional e bifásico a matriz resultante é heptadiagonal blocada e os blocos 2x2, com a seguinte composição:

$$\begin{pmatrix} \frac{\partial F_w}{\partial p_o} & \frac{\partial F_w}{\partial S_w} \\ \frac{\partial F_o}{\partial p_o} & \frac{\partial F_o}{\partial S_w} \end{pmatrix}$$

- $\delta \vec{x}^{\nu+1}$, é o vetor solução dado por $\delta x^{\nu+1} = x^{\nu+1} - x^\nu$, sendo x as variáveis p_o e S_w e ν o nível de iteração Newtoniana.

5.3 Ordenação da Malha e Solução da Matriz

O acoplamento entre a região do poço (malha cilíndrica) e a região do reservatório (malha cartesiana) será resolvido de duas formas distintas: 1) de forma totalmente acoplada e 2) de forma desacoplada, como será discutido a seguir:

5.3.1 Modelo Totalmente Acoplado

Neste caso as duas regiões são resolvidas simultaneamente no mesmo sistema de equações, para o esquema totalmente implícito na pressão e na saturação.

A figura 5.2 ilustra o tipo de ordenação da malha adotada neste trabalho.

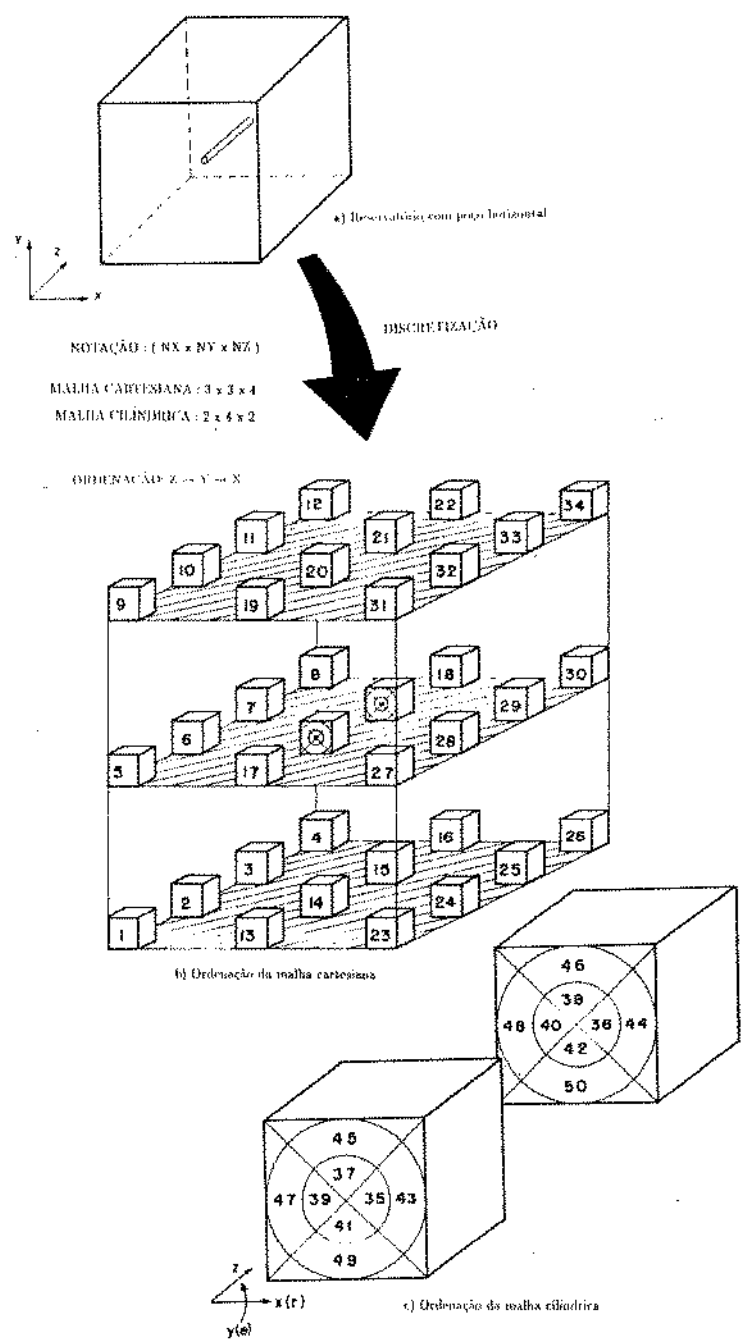


Figura 5.2: Ordenação da malha

Conforme pode-se verificar pela figura 5.2, esta ordenação é caracterizada pela numeração dos blocos primeiramente na direção z, depois na direção y - que é vertical na malha cartesiana e angular na malha cilíndrica -, e finalmente na direção x - que é na horizontal para a malha cartesiana e radial na cilíndrica. O sistema com a matriz Jacobiana gerada apresenta a seguinte forma:

$$\begin{pmatrix} J_{cc} & J_{cr} \\ J_{rc} & J_{rr} \end{pmatrix}^{\nu} \begin{pmatrix} \delta x_c \\ \delta x_r \end{pmatrix}^{\nu+1} = - \begin{pmatrix} G_c \\ G_r \end{pmatrix}^{\nu}$$

onde : J_{cc} é a submatriz de derivadas parciais das funções de resíduos do sistema cartesiano com relação às variáveis primárias do mesmo sistema;

J_{rr} é a submatriz de derivadas parciais das funções de resíduos do sistema cilíndrico com relação às variáveis primárias do mesmo sistema;

J_{cr} é a submatriz de derivadas parciais das funções de resíduos do sistema cartesiano com relação às variáveis primárias do sistema cilíndrico;

J_{rc} é a submatriz de derivadas parciais das funções de resíduos do sistema cilíndrico com relação às variáveis primárias sistema cartesiano;

δx_c é o vetor solução para o sistema cartesiano;

δx_r é o vetor solução para o sistema cilíndrico;

G_c é o vetor de termo independente para o sistema cartesiano;

G_r é o vetor de termo independente para o sistema cilíndrico.

As submatrizes J_{cr} e J_{rc} correspondem aos elementos de interfaces e a distância destes elementos à parte central da matriz depende da ordenação da malha. A figura 5.3 ilustra a matriz Jacobiana completa para uma malha com as seguintes características: cartesiana com 3x3x4, com um total de 34 blocos e uma cilíndrica de 2x4x2, com um total de 16 blocos, conforme a figura 5.2.

Nesta matriz e nas seguintes as letras (de A até G), representam blocos e tem o seguinte significado:

- A = Matriz 2 x 2 formada pelas derivadas das funções resíduos da água e do óleo em relação as variáveis primárias - p_o e S_w - para os blocos i, j, k . Ou seja:

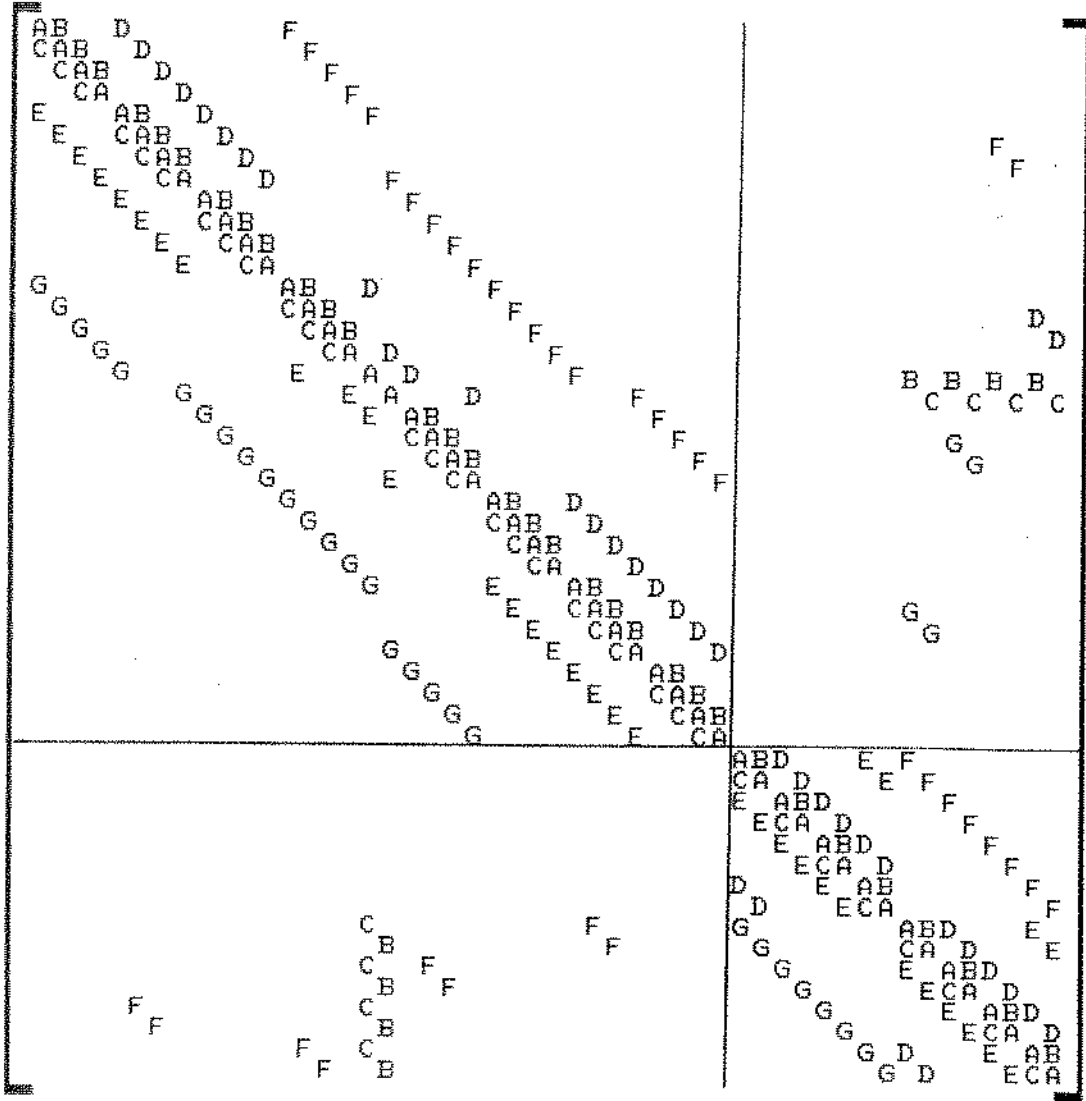


Figura 5.3: Matriz Jacobiana do sistema totalmente acoplado

$$\begin{pmatrix} \frac{\partial F_w}{\partial p_{oijk}} & \frac{\partial F_w}{\partial S_{wijk}} \\ \frac{\partial F_o}{\partial p_{oijk}} & \frac{\partial F_o}{\partial S_{wijk}} \end{pmatrix}$$

- B = Idem, em relação aos blocos $i, j, k + 1$;
- C = Idem, em relação aos blocos $i, j, k - 1$;
- D = Idem, em relação aos blocos $i, j + 1, k$;
- E = Idem, em relação aos blocos $i, j - 1, k$;
- F = Idem, em relação aos blocos $i + 1, j, k$;
- G = Idem, em relação aos blocos $i - 1, j, k$.

5.3.2 Modelo Desacoplado

Por este método as duas regiões são resolvidas separadamente, ou seja, é gerada uma matriz heptadiagonal para a malha cartesiana e outra para a malha cilíndrica.

O sistema de ordenação é o mesmo adotado anteriormente (figura 5.2) que resultará em matrizes heptadiagonais com blocos 2x2, tanto para a região do poço (malha cilíndrica) quanto para a região do reservatório (malha cartesiana), conforme pode ser visto respectivamente nas figuras 5.4 e 5.5. Deve ser observado que as matrizes geradas tem as bandas bem definidas, para as quais existem métodos eficientes de solução.

O acoplamento entre as duas regiões é feito através da transferência de massa de uma malha para outra, no qual o balanço de massa deve ser respeitado. Esta transferência pode ser feita através de dois processos, cujos fluxogramas são apresentados na figura 5.6. No primeiro, os termos de fluxos são transferidos de uma malha para a outra num time-step. Já no segundo, esta transferência é feita dentro de uma iteração Newtoniana.

Esta transferência é representada na figura 5.7, onde os termos q_{uli} , são os termos de fluxo das faces dos blocos cartesianos, e são calculados numa iteração ou num *time*

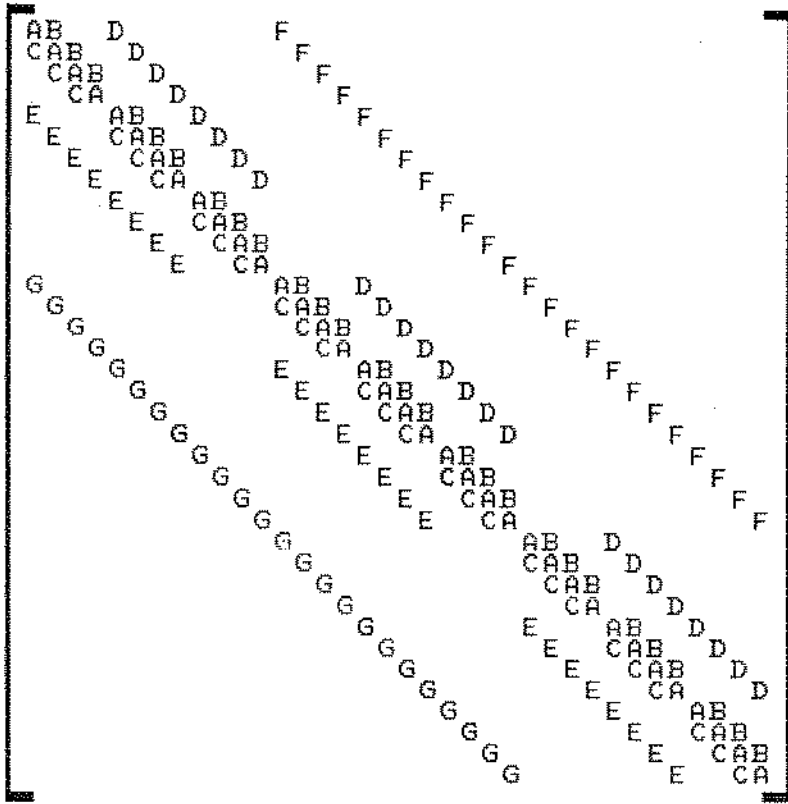


Figura 5.4: Matriz Jacobiana referente a malha cartesiana

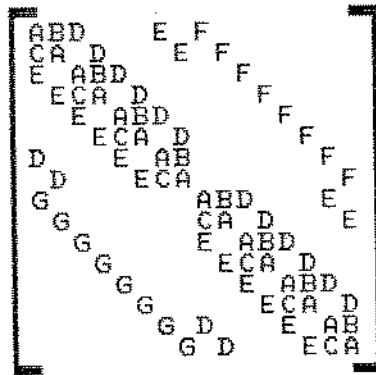


Figura 5.5: Matriz Jacobiana referente a malha cilíndrica

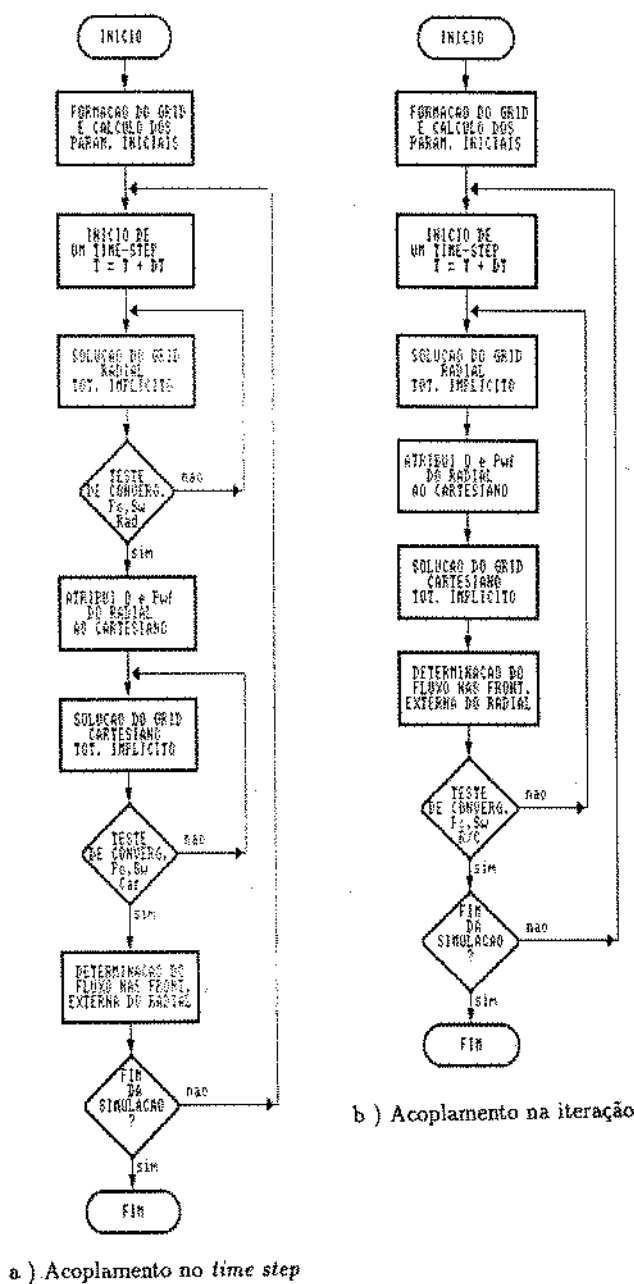


Figura 5.6: Fluxogramas para os métodos desacoplados

step. Estes termos substituem os termos de fluxo das faces externas dos blocos cilíndricos que fazem "contato" com os cartesianos, em qualquer direção.

Quando a malha cilíndrica for mais refinada que a cartesiana, é feita a ponderação dos termos de fluxo. Pela figura 5.6, nota-se que o fluxo que sai (entra) num bloco cartesiano deve ser igual ao somatório dos fluxos que entra (sai) de cada bloco cilíndrico, ou seja:

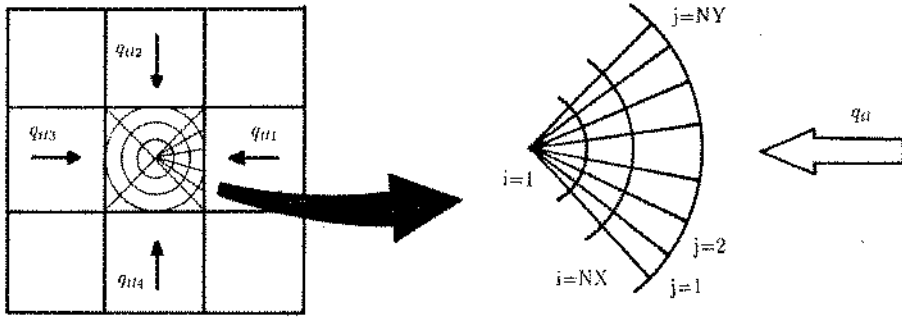


Figura 5.7: Representação dos termos de fluxo

$$q_{i1} = \sum_{j=1}^n q_{ij} \quad (5.4)$$

O fluxo para cada bloco cilíndrico também pode ser calculado pela lei de Darcy:

$$q_{ij} = \lambda_{ij} A_j \left(\frac{\partial \Phi_{ij}}{\partial d} \right)_{front} \quad (5.5)$$

onde, $\lambda_{ij} = k \frac{k_{r,ij}}{\mu_i B_i}$ será calculada no bloco a montante, A_j é a área de interface, $\frac{\partial \Phi_{ij}}{\partial d}$ a variação do potencial entre um bloco cartesiano e um bloco cilíndrico e d a projeção vertical ou horizontal da distância entre os pontos nodais.

Dividindo a equação 5.5 pela 5.4, obtem-se:

$$\frac{q_{ij}}{q_{i1}} = \frac{\lambda_{ij} A_j \left(\frac{\partial \Phi_{ij}}{\partial d} \right)_{front}}{\sum_{j=1}^{ny} \lambda_{ij} A_j \left(\frac{\partial \Phi_{ij}}{\partial d} \right)_{front}} \quad (5.6)$$

que rearranjada, resulta no fluxo na face de cada "fatia", que pode ser vista como uma ponderação do fluxo do bloco cartesiano.

$$q_{lj} = \frac{A_j \left(\frac{\partial \Phi_{lj}}{\partial d} \right)_{front}}{\sum_{j=1}^{n_y} A_j \left(\frac{\partial \Phi_{lj}}{\partial d} \right)_{front}} q_{tl} \quad (5.7)$$

5.4 Solução do Sistema Linear de Equações

Para a solução do sistema matricial podem ser usados métodos diretos ou iterativos. Para casos em que não se exija uma malha muito refinada, é perfeitamente viável a utilização do método direto por eliminação de Gauss. Já para casos mais complexos em que deve haver um grande refinamento da malha é recomendável a utilização de métodos iterativos.

Neste trabalho resolveremos o sistema pelo método direto, com o uso de um algoritmo com otimização do processo de eliminação de Gauss (com o algoritmo para matrizes esparsas, NSPIV³⁴) e também pelo método iterativo BSOR (Aziz e Settari (30)). Este último pode ser representado da seguinte forma:

$$\begin{pmatrix} J_{cc} & J_{cr} \\ J_{rc} & J_{rr} \end{pmatrix} \begin{pmatrix} x_c \\ x_r \end{pmatrix} = \begin{pmatrix} G_c \\ G_r \end{pmatrix}$$

Onde, J_{cc} representa a matriz jacobiana referente a região do reservatório, J_{rr} se refere a região do poço, J_{cr} e J_{rc} se referem às interfaces entre as duas regiões. x_c e x_r são os vetores de incógnitas, G_c e G_r são os termos independentes.

A partir de um valor inicial para x_c e x_r resolve-se, num primeiro passo, o sistema para a incógnita x_r :

$$J_{rc}x_c^m + J_{rr}x_r^* = G_r$$

$$x_r^{m+1} = wx_r^* + (1-w)x_r^m$$

o segundo passo é usar x_r^{m+1} , determinado no passo anterior e resolver o sistema para x_c , com a seguir:

$$J_{cc}x_c^* + J_{cr}x_r^{m+1} = G_c$$

$$x_c^{m+1} = wx_c^* + (1-w)x_c^m$$

o terceiro passo é a verificação da convergência,

$$|x_r^{m+1} - x_r^m| < \text{tolerância}$$

$$|x_c^{m+1} - x_c^m| < \text{tolerância}$$

Aqui m significa o nível de iteração do processo iterativo do sistema linear, dentro de uma iteração Newtoniana e w o fator de sobre-relaxação.

5.5 Condições de Produção

Tanto para o modelo totalmente acoplado, quanto para os desacoplados, a condição de produção poder ser pela especificação da vazão de produção ou da pressão de fluxo de fundo.

1. Vazão Especificada

Quando a vazão for especificada (vazão constante), a pressão de fluxo de fundo em cada iteração é calculada usando as propriedades de rocha e fluidos, da iteração anterior. O cálculo da pressão de fluxo de fundo, é feito com base nas propriedades de um bloco cilíndrico que se comunica com o poço. Devido às dimensões reduzidas destes blocos e o fato do poço ter condutividade infinita, este procedimento mostrou-se perfeitamente aceitável, pois testes para o cálculo de p_{wf} com todos os blocos comunicante com o poço, deram resultados praticamente iguais. Assim, usando o primeiro bloco teremos:

$$p_{wf} = p_{o1,j,k}^\nu - \gamma_{o1/2,j,k}^\nu (D_{ph} - D_{1,j,k}) - \frac{Q_{o1,j,k}}{T_{ox1/2,j,k}^\nu} \quad (5.8)$$

Nesta equação o subíndice $(1,j,k)$ significa que a propriedade é tomada no ponto nodal e $(1/2,j,k)$ na face do poço. $Q_{o1,j,k}$ é a vazão do bloco $(1,j,k)$, que é a vazão total especificada ponderada de acordo com as transmissibilidades das faces produtoras, ou seja, para um determinado j e k :

$$Q_{o1,j,k} = Q_o \frac{T_{ox1/2,j,k}}{\sum_{m=1}^{ny} \sum_{n=1}^{nz} T_{ox1/2,m,n}} \quad (5.9)$$

A transmissibilidade na face do poço é dada por:

$$T_{ox1/2,j,k} = FF_{x1/2,j,k} \frac{k_{ro1,j,k}}{\mu_{o1/2,j,k} B_{o1/2,j,k}} \quad (5.10)$$

e o fator de forma na face do poço é dado por:

$$FF_{x1/2,j,k} = \frac{\Delta\theta_j \Delta D_k}{\frac{1}{k_{1,j,k}} \ln \frac{r_{1,j,k}}{r_w}} \quad (5.11)$$

2. Pressão de Fluxo de Fundo Especificada

Quando a pressão é especificada, a vazão é determinada pelo somatório das vazões de cada bloco em comunicação com o poço:

$$Q_o = \sum_{j=1}^{ny} \sum_{k=1}^{nz} Q_{o1,j,k} \quad (5.12)$$

onde:

$$Q_{o1,j,k} = T_{ox1,j,k} [p_{o1,j,k}^\nu - p_{wf} - \gamma_{o1/2,j,k}^\nu (D_{ph} - D_{1,j,k})] \quad (5.13)$$

é a vazão de óleo de cada bloco cilíndrico em comunicação com o poço. Na equação 5.12 ny e nz são o número de blocos da malha cilíndrica nas direção y e z , respectivamente.

3. Vazão de Gás e Água

Em todos os casos simulados a pressão em qualquer bloco sempre estará acima da pressão de saturação, assim a vazão de gás será dada por:

$$Q_g = \sum_{j=1}^{ny} \sum_{k=1}^{nz} Q_{g1,j,k}$$

$$Q_g = \sum_{j=1}^{ny} \sum_{k=1}^{nz} Q_{o1,j,k} R_{s1,j,k} \quad (5.14)$$

A vazão de água será obtida através das razões de mobilidades:

$$Q_w = \sum_{j=1}^{ny} \sum_{k=1}^{nz} Q_{w1,j,k}$$

$$Q_w = \sum_{j=1}^{ny} \sum_{k=1}^{nz} \frac{\lambda_{w1,j,k}}{\lambda_{o1,j,k}} Q_{o1,j,k} \quad (5.15)$$

as mobilidades do óleo e da água são definidas como:

$$\lambda_{o1,j,k} = \frac{k_{ro1,j,k}}{B_{o1,j,k} \mu_{o1,j,k}} \quad (5.16)$$

$$\lambda_{w1,j,k} = \frac{k_{rwj,k}}{B_{w1,j,k} \mu_{w1,j,k}} \quad (5.17)$$

Este procedimento para o cálculo das condições de produção é usado tanto para o modelo totalmente acoplado como para os modelos desacoplados, isto proporciona a vantagem da não utilização de modelos de poço.

5.6 Derivadas

Conforme visto anteriormente, a solução do sistema com a aplicação do método de Newton Raphson, exige a determinação das derivadas das funções de resíduos do óleo e da água em relação as variáveis primárias para todos os blocos e seus vizinhos. As derivadas das funções de resíduos - termos de fluxo e termos de acumulação - dos termos fontes e todas as hipóteses consideradas para sua determinação são apresentadas no anexo A.

5.7 Controle Automático do Tempo

O tamanho do *time step* deve ser selecionado de forma adequada a garantir a estabilidade da solução e erros de truncamentos aceitáveis, (Aziz e Settari⁽³⁰⁾). Esta seleção pode ser feita com base na comparação entre as máximas variações da pressão e da saturação ocorridas num bloco, com valores limites pré-estabelecidos, de pressão (*DPLIM*) e saturação (*DSLIM*), que são fornecidos como dados de entrada.

No programa são calculadas as variações máximas para os blocos, que podem ser representadas por:

$$DSMAX^n = \max\{\Delta_t S_{ijk}^n\} \quad (5.18)$$

$$DPMAX^n = \max\{\Delta_t p_{ijk}^n\} \quad (5.19)$$

Caso $DSMAX$ e/ou $DPMAX$ excedam os valores limites, um novo valor para Δt^{n+1} será calculado, conforme:

$$\Delta t_s = \Delta t^n \frac{DSLIM}{DSMAX^n} \quad (5.20)$$

ou

$$\Delta t_p = \Delta t^n \frac{DPLIM}{DPMAX^n} \quad (5.21)$$

Assim o tamanho do novo *time step* será:

$$\Delta t^{n+1} = \min\{\Delta t_s, \Delta t_p\} \quad (5.22)$$

Quando as variações de pressão e saturação estão dentro dos limites estabelecidos, o controle do *time step* é feito pelo número de iterações Newtonianas. Assim estabelece-se um número mínimo ($ITMIN$) e um número máximo ($ITMAX$) de iterações.

O novo *time step* será estabelecido da seguinte forma:

$$\text{número de iterações} \leq ITMIN \rightarrow \Delta t^{n+1} = 1.5\Delta t^n$$

$$ITMIN < \text{número de iterações} \leq ITMAX \rightarrow \Delta t^{n+1} = \Delta t^n$$

$$\text{número de iterações} > ITMAX \rightarrow \Delta t^{n+1} = 0.5\Delta t^n$$

6 VALIDAÇÃO DO MODELO

A validação foi feita através da comparação dos resultados obtidos dos modelos desenvolvidos com as soluções analíticas desenvolvidas por Odeh e Babu ⁽¹³⁾ para o drawdown em um poço horizontal com fluxo de uma fase.

6.1 Solução Analítica

As análises para a descrição do comportamento do fluxo monofásico, apresentadas por alguns pesquisadores ⁽¹¹⁻¹⁷⁾, mostram que para um poço horizontal podem ocorrer até quatro regimes de fluxo transientes distintos, que tem sua existência e duração como função das características do reservatório - dimensões e forma - características da rocha, fluido e do poço horizontal - raio, comprimento e localização dentro do reservatório.

As soluções analíticas usadas para a validação, consideram o fluxo de um fluido levemente compressível e sem considerar o efeito gravitacional. Assume um poço horizontal com condutividade infinita, raio r_w e comprimento L , perfurado paralelamente a uma das faces de um reservatório fechado. As dimensões do reservatório são, espessura; h , comprimento (direção x); a , largura (direção y , paralela ao poço); b . O poço horizontal com coordenadas do início e fim definidos respectivamente por (x_0, y_1, z_0) e (x_0, y_2, z_0) . O modelo físico e os parâmetros usados nas fórmulas analíticas estão representados na figura 6.1.

Conforme mencionado anteriormente, o transiente completo para um poço horizontal apresenta quatro regimes distintos: 1) Radial inicial, 2) Linear inicial, 3) Pseudo radial tardio e 4) Linear tardio. A representação esquemática das linhas de fluxo em cada regime são apresentadas na figura 6.2.

As equações que regem o comportamento da pressão do fluxo monofásico para produção com vazão constante, e os tempos de início e fim para os quatros regimes de fluxos , são apresentados a seguir:

1. Regime Radial Inicial

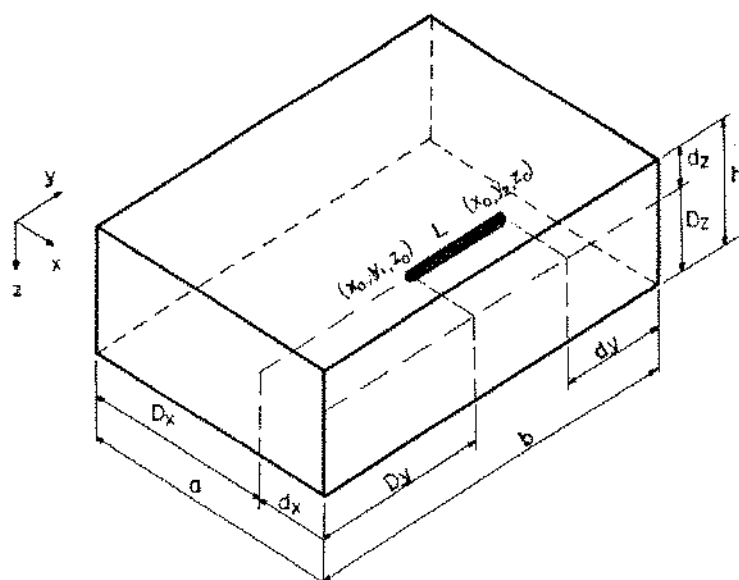


Figura 6.1: Representação do modelo físico

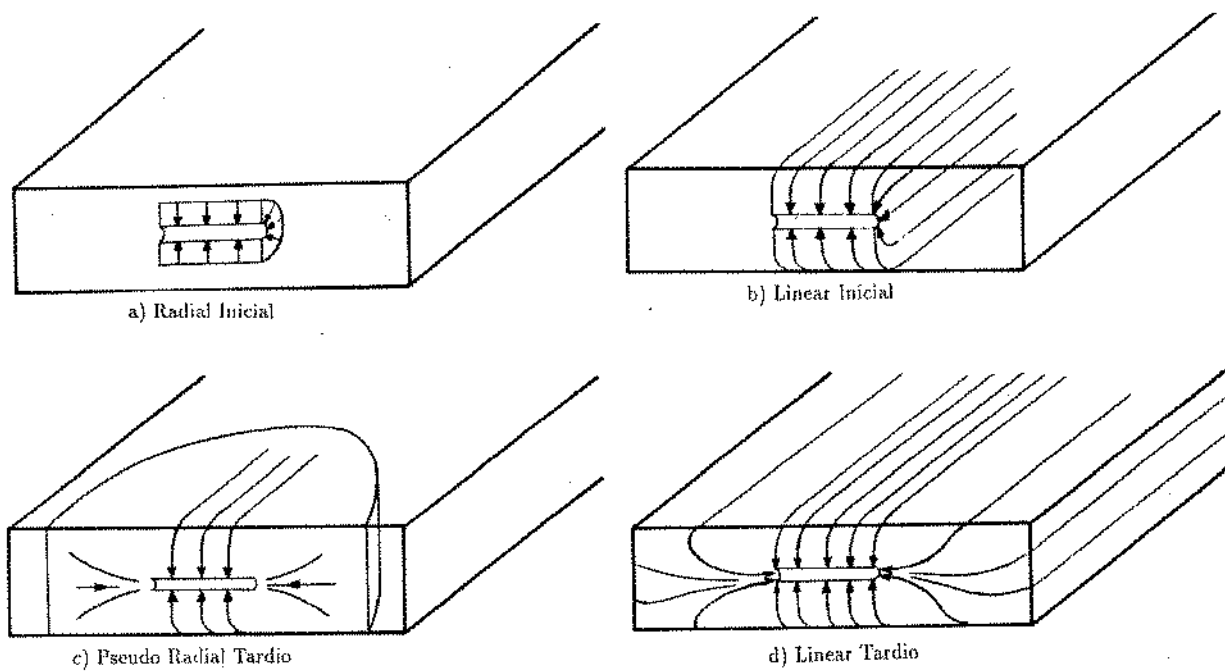


Figura 6.2: Representação das linhas de fluxo dos regimes do transiente

Ocorre imediatamente após o poço ser colocado em produção e é análogo ao período de fluxo radial para um poço vertical de comprimento L , totalmente penetrante. O final deste período é dado pelo menor valor abaixo.

$$t_{end} = 0.4744 d_z^2 \phi \mu c_t / k_z \quad (6.1)$$

ou

$$t_{end} = 0.0329 L^2 \phi \mu c_t / k_y \quad (6.2)$$

Devido à curta duração, este período pode ser mascarado pelos efeitos decorrentes do início do primeiro fluxo. Em adição, a não consideração do efeito gravitacional pode distorcer os resultados.

O comportamento da pressão é dado por:

$$p_{wf} = p_i - \frac{0.1833 q \mu B}{\sqrt{k_x k_z} L} \left(\log \frac{\sqrt{k_x k_z} t}{\phi \mu c_t r_w^2} + 0.3492 + 0.87 s_a \right) \quad (6.3)$$

Para este regime um gráfico semi-log de $[(p_i - p_{wf}) = \Delta p]$ vs t dará uma linha reta com inclinação m , dada por:

$$|m| = \frac{0.1833 q \mu B}{\sqrt{k_x k_z} L} \quad (6.4)$$

2. Regime Linear Inicial

Este regime ocorre no período em que o transiente de pressão é influenciado somente pelos limites inferior e superior do reservatório, e ocorrerá se o comprimento do poço for significativamente maior que a espessura do reservatório, e se a contribuição do fluxo nas pontas do poço for desprezada. A duração deste período é dada aproximadamente por:

$$t_{start} = 0.4744 D_z^2 \phi \mu c_t / k_z \quad (6.5)$$

$$t_{end} = 0.0422L^2\phi\mu c_t/k_y \quad (6.6)$$

O comportamento da pressão é dado por:

$$p_{wf} = p_i - \frac{0.5646q\mu B}{Lh} \left[\sqrt{\frac{t}{\phi\mu c_t k_x}} + \frac{0.2803h}{\sqrt{k_x k_z}} \left(\ln \frac{h}{r_w} + 0.25 \ln \frac{k_x}{k_z} + \right. \right. \\ \left. \left. - \ln \sin \frac{180^\circ z_o}{h} - 1.838 + s_a \right) \right] \quad (6.7)$$

Para este regime um gráfico de $(p_i - p_{wf})$ vs $\sqrt{t}$ dará uma linha reta com inclinação m , dada por:

$$|m| = \frac{0.5646q\mu B}{\sqrt{\phi\mu c_t k_x} Lh} \quad (6.8)$$

3. Regime Pseudo Radial Tardio

Se a espessura do reservatório não for suficientemente grande quando comparada com a largura, este regime pode não ocorrer. Também é necessário que razão de penetração, definida por L/b , seja menor que 0.45. O início deste período ocorre para:

$$t_{start} = 0.3900L^2\phi\mu c_t/k_y \quad (6.9)$$

e o final para o menor valor abaixo:

$$t_{end} = 0.5271\phi\mu c_t(d_y + L/4)^2/k_y \quad (6.10)$$

ou

$$t_{end} = 0.4348\phi\mu c_t d_x^2/k_x \quad (6.11)$$

O comportamento da pressão é dado por:

$$p_{wf} = p_i - \frac{0.1833q\mu B}{h\sqrt{k_x k_y}} \left[\log \frac{k_y t}{\phi \mu c_t L^2} + 1.8192 + 0.87 \sqrt{\frac{k_y}{k_z}} \frac{h}{L} \left(\ln \frac{h}{r_w} + 0.25 \ln \frac{k_x}{k_z} - \ln \sin \frac{180^\circ z_o}{h} - 1.838 + s_a \right) \right] \quad (6.12)$$

Para este regime um gráfico semi-log de $(p_i - p_{wf})$ vs t , é uma linha reta com inclinação m , dada por:

$$|m| = \frac{0.1833q\mu B}{\sqrt{k_x k_y} h} \quad (6.13)$$

4. Regime Linear Tardio

Este período ocorre após o transiente de pressão ter atingido os limites do reservatório nas direções y e z e o comportamento nestas direções se tornará do tipo pseudo permanente. É análogo ao fluxo linear que ocorre para uma fratura vertical, com uma queda de pressão adicional resultante da penetração parcial nas direções y e z . Se o reservatório for infinito este período poderá não ocorrer. O início deste período é dado pelo menor valor abaixo:

$$t_{start} = 1.2650 \phi \mu c_t (D_y + L/4)^2 / k_y \quad (6.14)$$

ou

$$t_{start} = 0.4744 D_z^2 \phi \mu c_t / k_z \quad (6.15)$$

o final é dado por:

$$t_{end} = 0.4348 \phi \mu c_t d_x^2 / k_x \quad (6.16)$$

O comportamento da pressão é dado por:

$$p_{wf} = p_i - \frac{0.5646q\mu B}{bh} \left[\sqrt{t} \phi \mu c_t k_x + \frac{0.2803h}{\sqrt{k_y k_z}} \left(\ln \frac{h}{r_w} + 0.25 \ln \frac{k_x}{k_z} + \right. \right. \\ \left. \left. - \ln \sin \frac{180^\circ z_o}{h} - 1.838 + s_a \right) \right] \quad (6.17)$$

Para este regime um gráfico $(p_i - p_{wf})$ vs $\sqrt{t}$ é uma linha reta com inclinação m , dada por:

$$|m| = \frac{0.5646q\mu B}{\sqrt{\phi \mu c_t k_x} bh} \quad (6.18)$$

Em seu artigo Odeh e Babu⁽¹³⁾ observam que a ocorrência dos quatro regimes de fluxo podem não ser garantida. Assim, devemos nos preocupar em que a duração de um regime particular seja suficientemente longa para que as análises possam ser conclusivas. A permeabilidade, volume drenado, forma do reservatório, o comprimento e a localização do poço, afetam em diferentes graus a duração de cada período de fluxo. Podem ocorrer situações em que não ocorram nenhum dos quatro regimes de fluxo transiente.

6.2 Simulação dos Testes

Para demonstrar a validade do modelo desenvolvido, foi feito primeiramente, a comparação entre os resultados analíticos e numéricos para o comportamento da pressão no drawdown para o caso de um poço horizontal totalmente penetrante. Posteriormente, foi feita a determinação da permeabilidade, individualmente para cada período do regime transiente e comparação com os valores usados na simulação.

Os principais parâmetros utilizados em todos os casos simulados foram os mesmos apresentados por Odeh e Babu⁽¹³⁾ e são apresentados na tabela 6.1.

6.2.1 Comportamento da Pressão no Drawdown

Conforme mencionado anteriormente, foi escolhida uma situação para a comparação do comportamento da pressão obtidos numérica e analiticamente. Esta situação

Parâmetro	Valor
Vazão ($m^3 std/d$)	127.2
Viscosidade ($Pa.s$)	1.0×10^{-3}
Fator volume de formação ($m^3 res/m^3 std$)	1.25
Raio do poço (m)	0.08
Pososidade (fr)	0.20
Compressibilidade total (Pa^{-1})	2.18×10^{-9}
Skin	0.0
Permeabilidade absoluta (m^2)	4.94×10^{-14}

Tabela 6.1: Parâmetros usados na validação

é a simulação da produção de um reservatório produzindo através de um poço totalmente penetrante e cujo objetivo é verificar os resultados numéricos sujeitos à influência da mudança da forma de simulação do fluxo na direção x , ou seja, em direção ao poço, a representação do fluxo passa de linear para radial.

As características do reservatório, malha utilizada e os períodos de ocorrência de cada regime são apresentados nas tabelas 6.2 e 6.3. E a comparação entre a solução analítica com os resultados obtidos pelo modelo numérico é mostrada na figura 6.3.

Da análise deste gráfico, conclui-se que é boa a concordância entre os resultados. Havendo um pequeno desvio entre as curvas logo no início, que é devido aos efeitos iniciais da entrada em produção e a não consideração do efeito gravitacional nas fórmulas analíticas, e outro desvio após o término do regime radial inicial - similar a um pseudo skin - que é causado pela diferença entre os volumes dos blocos da interface da malha cilíndrica com a cartesiana. Este ultimo efeito pode ser reduzido com uma discretização mais refinada e com variação mais suave entre os volumes dos blocos externos da malha cilíndrica e os blocos contíguos da malha cartesiana. Devido as limitações de computador, isto não foi possível neste trabalho.

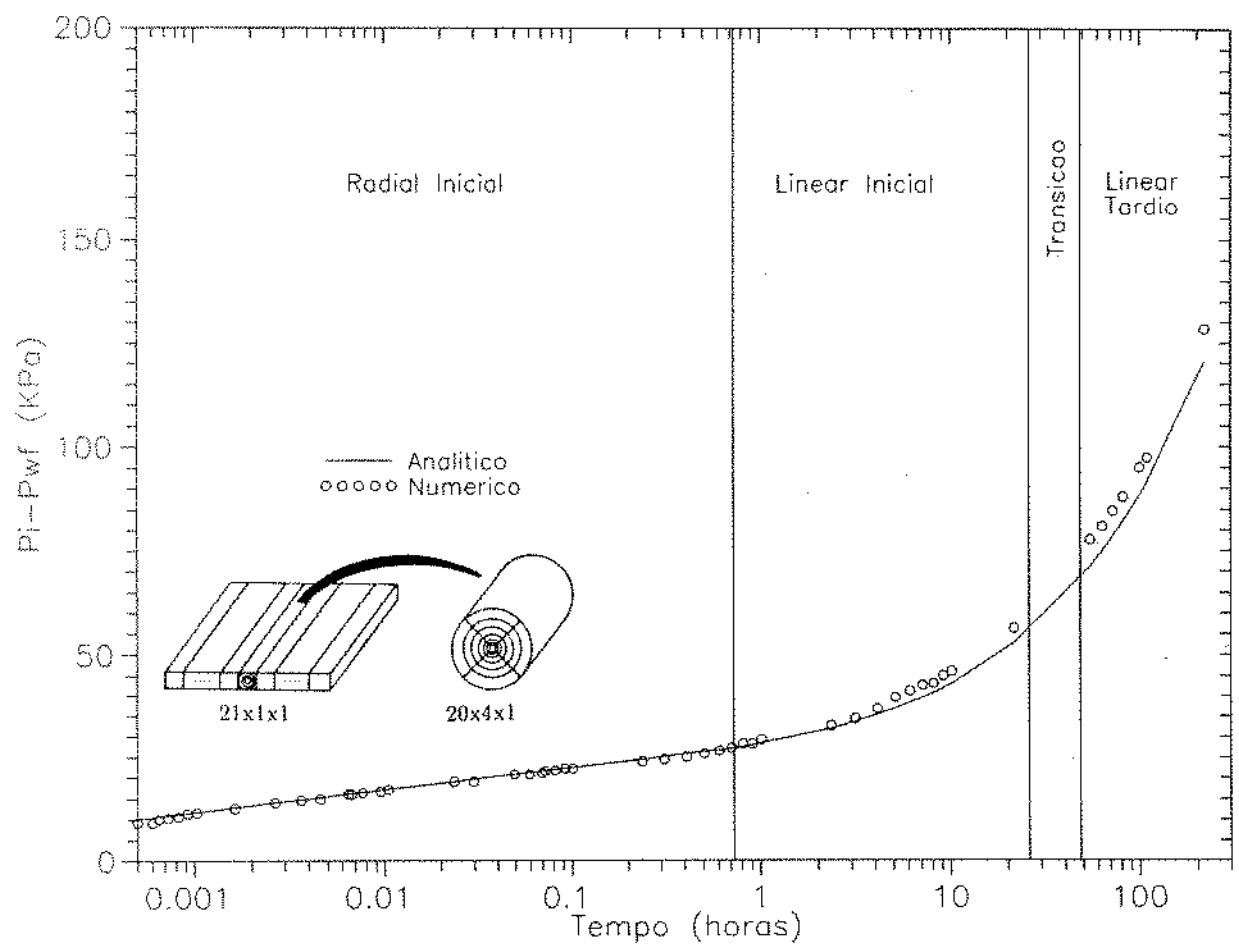


Figura 6.3: Comparação entre a solução analítica e a numérica

Características	
Comprimento do reservatório (m)	1050.0
Largura do reservatório (m)	500.0
Espessura do reservatório (m)	50.0
Comprimento do poço (m)	500.0
Malha cartesiana	
Número de blocos na direção x	21
Número de blocos na direção y	1
Número de blocos na direção z	1
Malha cilíndrica	
Número de blocos na direção r	20
Número de blocos na direção θ	4
Número de blocos na direção z	1

Tabela 6.2: Características do reservatório e malha

Regimes	tempo (hr)	
	início	fm
Radial inicial	—	0.73
Linear inicial	0.73	25.82
Pseudo radial tardio	—	—
Linear tardio	48.41	293.5

Tabela 6.3: Duração, em horas, dos regimes transiente.

6.2.2 Determinação da Permeabilidade

Para a determinação da permeabilidade através dos resultados numéricos, e posterior comparação com os utilizados na simulação, foram escolhidos reservatório com dimensões particulares, nas quais os diversos regimes de fluxo transiente tenham duração suficiente para as análises individuais.

1. Regime Radial Inicial

Os principais parâmetros usados para o regime radial inicial foram:

$$Q = 50 \text{ m}^3 \text{std/d}$$
$$k_x = k_y = k_z = 9.869 \times 10^{-15} \text{m}^2 \text{ (10 md)}$$
$$a = 550 \text{ m}$$
$$b = 500 \text{ m}$$

$$h = 50 \text{ m}$$

$$L = 500 \text{ m}$$

$$d_x = 250 \text{ m}$$

$$d_z = 25 \text{ m}$$

A malha usada foi: cartesiana; 11x1x1 e a cilíndrica; 10x4x1. Para o reservatório com as dimensões acima, segundo a equação 6.1 ou a 6.2 o final do regime radial inicial ocorrerá para $t_{end} = 3.63hr$. O gráfico de Δp_{wf} vs $\log t$ é mostrado na figura 6.4, do qual obtem-se uma reta com inclinação de $m = 26.88 KPa/ciclo$, e através da equação 6.4, obtem-se a permeabilidade de $9.849 \times 10^{-15} \text{ m}^2$ (9.98 md).

2. Regime Linear Inicial

Para o regime linear inicial não foi feita a determinação da permeabilidade, pois a duração deste regime é muito pequena e ocorre para situações muito particulares.

3. Regime Pseudo Radial Tardio

Para a validação do regimes pseudo radial tardio, foi encontrada certa dificuldades, pois o simulador desenvolvido faz o uso intensivo de matrizes de quatro dimensões (três dimensões para as três direções, e uma para cada reservatório; cilíndrico e cartesiano), isto exige grande capacidade de memória de computador e como estamos trabalhando com micro computador ficamos limitados ao refinamento (número de blocos) da malha. O maior refinamento areal da malha conseguido foi de 7x7x1 para o cartesiano e para o cilíndrico. Para se conseguir um tempo suficientemente longo para as análises do regime em questão é necessário uma malha mais refinada nas direções x e y .

Mesmo limitado ao pequeno número de blocos, fizemos dois testes, que mesmo não dando uma grande duração para o regime pseudo radial tardio, mostra que os resultados do modelo são coerentes com as deduções analíticas.

Os principais parâmetros usados foram:

$$Q = 50 \text{ m}^3\text{std}/d$$

$$a = 350 \text{ m}$$

$$b = 350 \text{ m}$$

$$h = 50 \text{ m}$$

$$D_x = D_y = 75 \text{ m}$$

$$D_z = 25 \text{ m}$$

Foi feita a simulação com o uso de dois valores de permeabilidade:

$$9.8690 \times 10^{-15} \text{ m}^2 (10 \text{ md}) \text{ e } 1.9738 \times 10^{-14} \text{ m}^2 (20 \text{ md}).$$

Para estas condições, o regime pseudo radial tardio tem a seguinte duração: $t_{start} = 11.94 \text{ hr}$ e $t_{end} = 163.00 \text{ hr}$. Os gráficos de Δp vs $\log t$, para as permeabilidades de $9.8690 \times 10^{-15} \text{ m}^2 (10 \text{ md})$ e $1.9738 \times 10^{-14} \text{ m}^2 (20 \text{ md})$, são apresentados respectivamente, nas figuras 6.5 e 6.6. Nestes gráficos notamos que, apesar do regime não ter longa duração, se tomarmos a inclinação da reta logo após o término do regime radial inicial (visto que neste caso não há ocorrência do regime linear inicial), serão obtidas as seguintes inclinações: $m = 250.0 \text{ KPa/ciclo}$ e $m = 125 \text{ KPa/ciclo}$. Usando estes valores na equação 6.13 obtem-se aproximadamente permeabilidades de $9.8690 \times 10^{-15} \text{ m}^2 (10 \text{ md})$ e $1.9738 \times 10^{-14} \text{ m}^2 (20 \text{ md})$.

4. Regime Linear Tardio

Para o regime linear tardio os parâmetros usados foram:

$$Q = 50 \text{ m}^3\text{std}/d$$

$$k_x = k_y = k_z = 9.869 \times 10^{-15} \text{ m}^2 (10 \text{ md})$$

$$a = 1350 \text{ m}$$

$$b = 500 \text{ m}$$

$$h = 50 \text{ m}$$

$$L = 500 \text{ m}$$

A malha usada para representar este reservatório foi: cartesiana; $27 \times 1 \times 1$, cilíndrica; $5 \times 4 \times 1$.

Na figura 6.7, é representado o gráfico de ΔP vs $\sqrt{t}$, da qual obtem-se uma reta com inclinação $m = 15.2kPa/hr^{1/2}$, que aplicada a equação 6.18 resulta numa permeabilidade de $9.671 \times 10^{-15}m^2(9.8md)$.

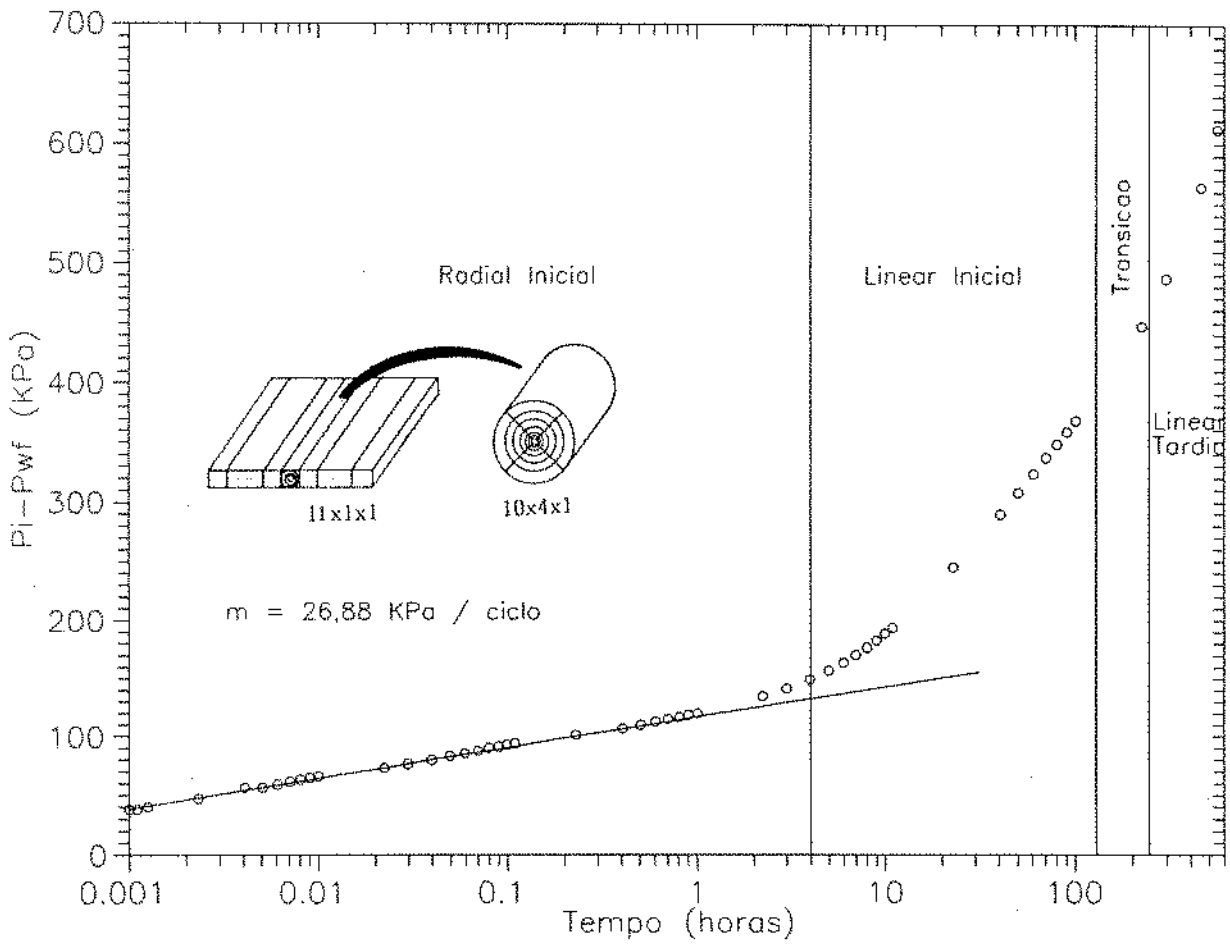


Figura 6.4: Gráfico de Δp vs $\ln t$, para o radial inicial

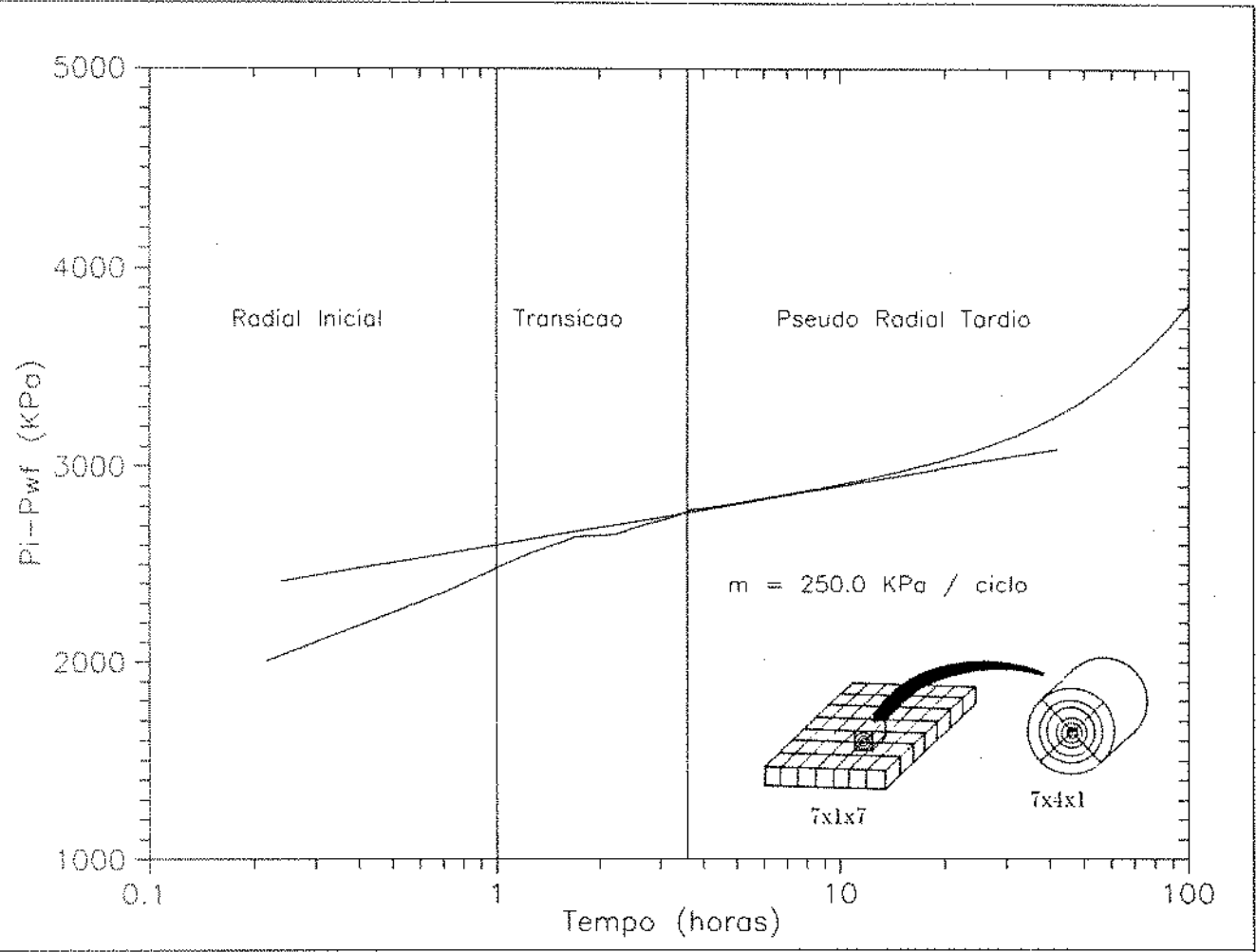


Figura 6.5: Gráfico de Δp vs $\ln t$, para o pseudo radial tardio - $k = 1.97 \times 10^{-14} m^2$

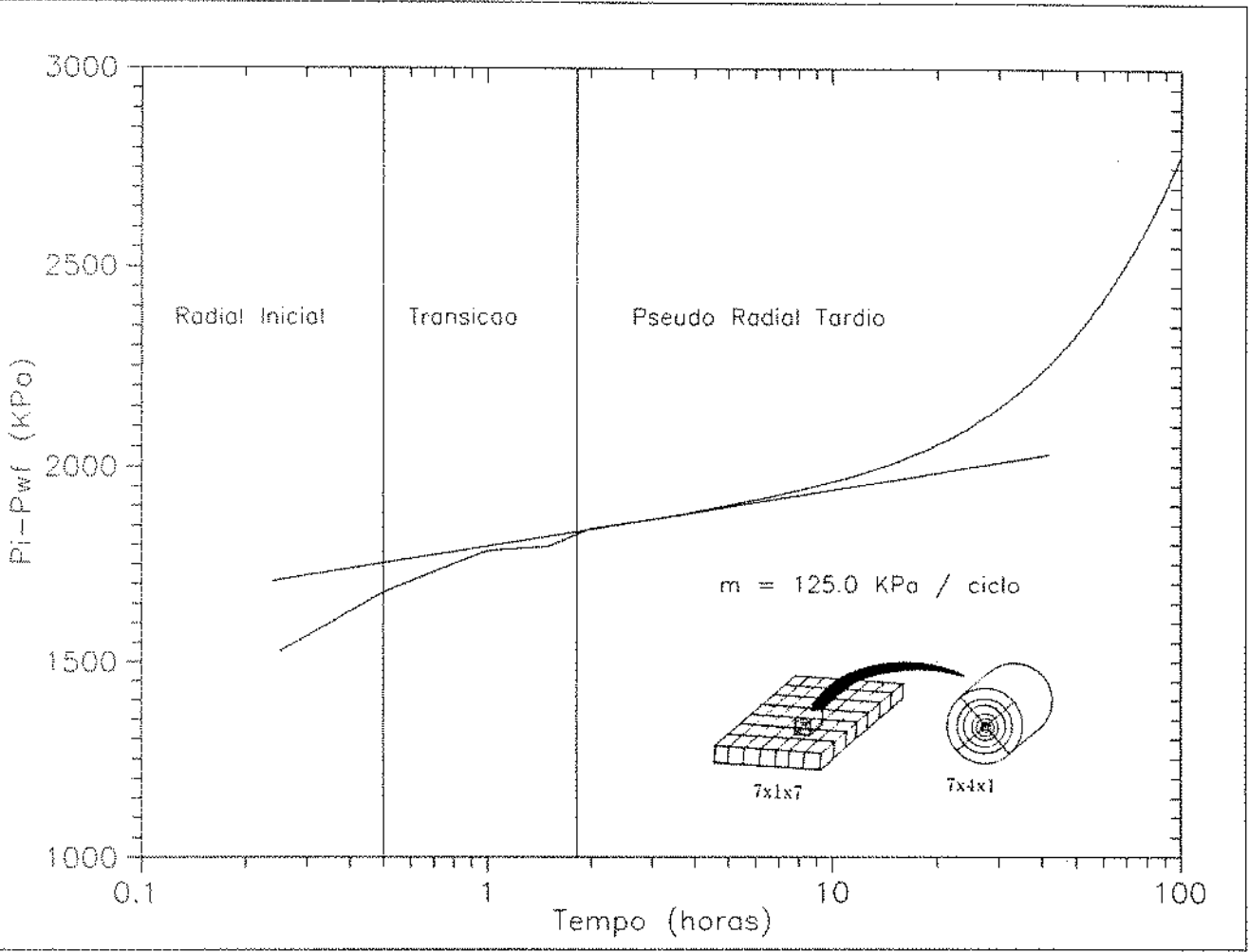


Figura 6.6: Gráfico de Δp vs $\ln t$, para o pseudo radial tardio - $k = 9.87 \times 10^{-15} m^2$

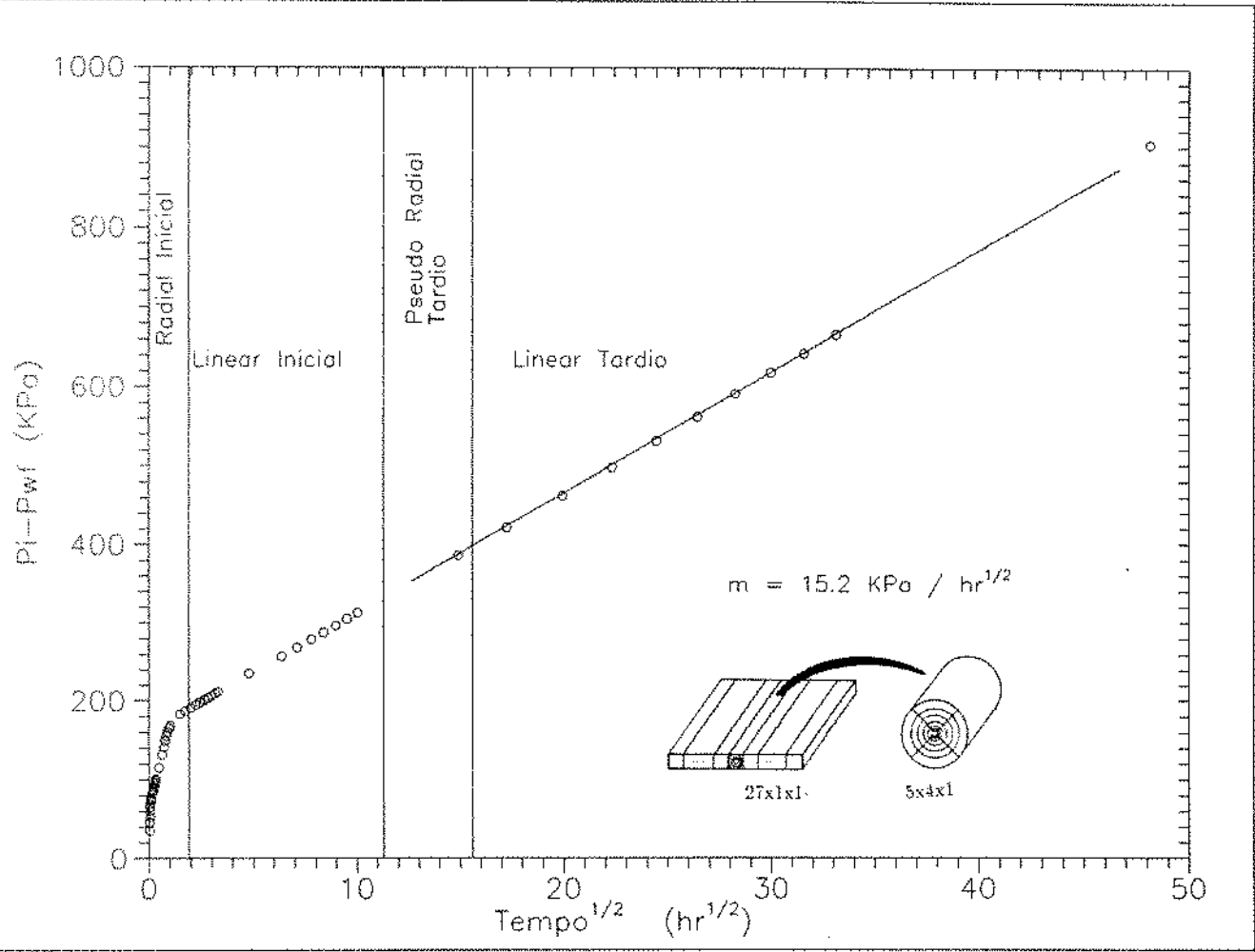


Figura 6.7: Gráfico de Δp vs $t^{1/2}$, para o linear tardio

7 ANÁLISE DOS RESULTADOS

O objetivo deste capítulo é a apresentação e análise dos resultados da simulação do comportamento de um reservatório produzindo através de um poço horizontal e dos métodos usados para esta simulação. Esta análise foi feita em duas partes: na primeira, são apresentados os resultados do simulador totalmente acoplado, com o sistema linear de equações resolvido por método direto e iterativo (BSOR) e a malha cilíndrica que representa a região do poço com diferentes refinamentos angular e radial. Na segunda parte foi feita a comparação entre os resultados obtidos pelos modelos que usam o método totalmente acoplado e os desacoplados.

Os dados de simulação, condições de produção e as características do reservatório que foram usados na simulação são apresentados na tabela 7.1. Enquanto que os dados das análises PVT do óleo, gás e água e os dados das análises petrofísicas são apresentados no anexo B.

Voltamos a salientar que devido ao uso de micro computador as malhas simuladas não tem grande refinamento, mas foram o suficiente para que as análises fossem conclusivas.

7.1 Modelo Totalmente Acoplado

7.1.1 Comparação entre os Métodos de Solução

O programa computacional desenvolvido prevê a solução da matriz Jacobiana por método direto ou iterativo. Para efeito de investigação, a solução com método direto foi feita de duas formas: a primeira forma foi a utilização da eliminação de Gauss normal com a matriz completa - trabalhando inclusive com os elementos nulos - e a segunda é através da utilização do algoritmo (NSPIV³⁴), também baseado na eliminação de Gauss, que trabalha somente com os elementos não nulos. Para os casos simulados pode-se fazer as seguintes observações:

Parâmetros de Simulação	Unidade	Valor
Tolerância para pressão	Pa	3447.40
Tolerância para saturação	fr	0.0010
Máxima variação da saturação num <i>time step</i>	fr	0.0500
Máxima variação da pressão num <i>time step</i>	kPa	689.48
Incremento de tempo máximo	$dias$	10.00
Incremento de tempo mínimo	$dias$	0.0010
Número máximo de iterações no Jacobiano		5
Número mínimo de iterações no Jacobiano		2
Dados de produção e poço		
Vazão de óleo máxima	m^3/dia	300.00
Vazão de óleo mínima	m^3/dia	15.00
Pressão de fluxo mínima no poço	MPa	30.00
Raio do poço	m	0.10
Cota do poço	m	-2560.00
Dados do reservatório		
Saturação inicial de água	fr	0.12
Compressibilidade da rocha	MPa^{-1}	0.440×10^{-3}
Pressão inicial do reservatório	MPa	35.00
Cota do contato óleo/água	m	-2585.00
Permeabilidade absoluta (x, y e z)	m^2	49.3×10^{-13}
Porosidade	fr	0.30

Tabela 7.1: Parâmetros Usados na Simulação

- Tempo de execução - o algoritmo otimizado trouxe uma redução média de até 60% no tempo de simulação, em relação a eliminação de Gauss com a matriz completa;
- Área de memória - À primeira vista pode-se pensar que o método otimizado traria uma redução de memória necessária, mas isso não ocorre, porque apesar de não necessitar de armazenar toda a matriz, o algoritmo trabalha com vetores auxiliares que são tão maiores quanto for a esparsidade da matriz, ou seja, a distância entre os elementos que representam as interfaces e a diagonal principal.

O método iterativo usado foi o de sobre-relaxação sucessiva por blocos (BSOR), e para as situações simuladas notou-se que esse método é desvantajoso, em relação ao método direto, quanto ao tempo de simulação. Para que os resultados fossem aceitáveis, a tolerância usada no processo iterativo do sistema linear de equações não pôde ser menor que 10^{-5} .

A vantagem desse método é em relação a pequena área de memória necessária, assim para problemas com grande número de blocos passa a ser atrativo.

Para a situação de um poço horizontal totalmente penetrante num reservatório de $300 \times 150 \times 150m$, representado por uma malha cartesiana de $3 \times 3 \times 1$ e uma malha cilíndrica de $3 \times 4 \times 1$, o fator de sobre-relaxação que proporcionou um menor número de iterações foi de aproximadamente 1.15, conforme pode ser visto pela figura 7.1

7.1.2 Refinamento Radial

O objetivo é verificar a influência do refinamento da malha cilíndrica na direção radial, sobre o comportamento da produção em duas situações: a primeira, com o poço totalmente penetrante e a segunda com poço parcialmente penetrante.

- Poço totalmente penetrante -

Foi simulado um reservatório de $250 \times 150 \times 600m$, com malha cartesiana de $5 \times 3 \times 1$ e malha cilíndrica de $3,5,10 \times 4 \times 1$.

Nesta primeira situação o refinamento radial influi somente na malha cilíndrica, já que não há mudança do número de interfaces cilindro/cartesiano nas direções x e y .

Os resultados do comportamento da pressão de fluxo e vazão de óleo são apresentados respectivamente nas figuras 7.2 e 7.3. Observa-se que para o comportamento da pressão para vazão constante, os resultados são iguais para os três refinamentos. Há uma diferença entre os tempos de chegada a pressão de fluxo e a vazão mínima; quanto maior o refinamento mais rápido serão atingido estes tempos.

A produção de água em todos os casos foi insignificante, e como não houve diferenças para este caso, torna-se desnecessário a apresentação dos gráficos.

- Poço parcialmente penetrante:

As dimensões do reservatório foram: 300 x 150 x 300m com malha cartesiana de 3 x 3 x 3 e malha cilíndrica de 2,3,5 x 4 x 1.

Nesta situação há variação do número de interfaces na direção z . Quanto maior o refinamento, maior será o número de elementos de interfaces na matriz Jacobi-ana. Os resultados do comportamento do reservatório com a produção de um poço horizontal centrado com 100 m de comprimento, são apresentados nas figuras 7.4; p_{wf} versus tempo, figura 7.5; vazão de óleo versus tempo, figura 7.6; vazão de água versus tempo e figura 7.7; razão água/óleo versus o tempo. Pelo gráfico de p_{wf} vs t nota-se que há uma pequena diferença entre as pressões para os diferentes refinamentos. Nesta situação, a produção de água já é significativa, sendo maior para malha com menor refinamento, conforme pode ser visto pela figura 7.6 e 7.7. Isto pode ser explicado pelo fato de que para um maior refinamento a distribuição da saturação de água é mais homogênea.

7.1.3 Refinamento Angular

Para este caso, foi estudado o comportamento da produção de um poço totalmente penetrante com a malha cilíndrica sendo dividida em 4 e 8 partes na direção angular. Quando

o número de divisões é maior que 4, há - na direção x e/ou y - a comunicação de 1 bloco cartesiano com mais de 1 cilíndrico, assim pode-se analisar a sua influência nos resultados. As dimensões do reservatório foram: 250 x 150 x 600m, com a malha cartesiana de 5 x 3 x 1.

Na figura 7.8 são apresentadas as curvas de p_{wf} vs t . Nota-se que o comportamento é igual em todos os casos. O comportamento da vazão de óleo com tempo é apresentado na figura 7.9, na qual nota-se que há uma tendência de redução do tempo de produção para malhas mais refinadas, quando a vazão de produção é variável e pequena. Neste caso a produção de água, também foi insignificante.

Deve ser observado, que se ocorresse a produção significativa de uma segunda fase, haveria diferenças no comportamento para os diferentes refinamentos angulares, pois a distribuição de saturações nos dois casos seriam diferentes.

7.2 Comparação entre o Acoplado e os Desacoplados

Um dos principais objetivos desse trabalho é a comparação entre os resultados do simulador totalmente acoplado e os desacoplados - que fazem o acoplamento numa iteração ou num *time step*. Pelos casos simulados, podem ser feitas as seguintes observações:

- tolerância - Para os desacoplados, deve-se tomar certos cuidados com uma tolerância "folgada" no processo de convergência do Newton-Raphson. Pois como é um processo que usa os termos de fluxo, da iteração ou do *time step* anterior da malha cartesiana, como influxo nas fronteiras externas da malha cilíndrica, pode haver problemas de balanço de materiais nos blocos da malha cilíndrica.
- Tempo - O tempo de simulação é menor para os modelos desacoplados.
- Capacidade de memória - A grande vantagem dos modelos desacoplados é em relação à menor área de memória necessária ao processamento. São armazenadas somente as matrizes Jacobianas referentes às duas malhas - que a depender do algoritmo

utilizado, são transformadas em vetores que armazenam os elementos não nulos. Não há elementos de interfaces. E estas matrizes são heptadiagonais normais que podem ser resolvidas por métodos rápidos. No caso foi usada a *subroutine GBAND* (Aziz e Settari⁽³⁰⁾) .

Os resultados para um reservatório de 250 x 250 x 600m com uma malha cartesiana 5x3x1 e malha cilíndrica de 3x4x1, para um poço totalmente penetrante, podem ser vistos nas figuras 7.10, 7.11 e 7.12. Através destes gráficos nota-se que no período em que a vazão de óleo é constante, os resultados dos desacoplados coincidem e existe uma pequena diferença deles para o totalmente acoplado. A outra diferença notada, é que quando o acoplamento é feito num *time step* há tendência de ser um pouco maior o tempo de produção , quando a pressão de fundo é constante e a vazão é baixa.

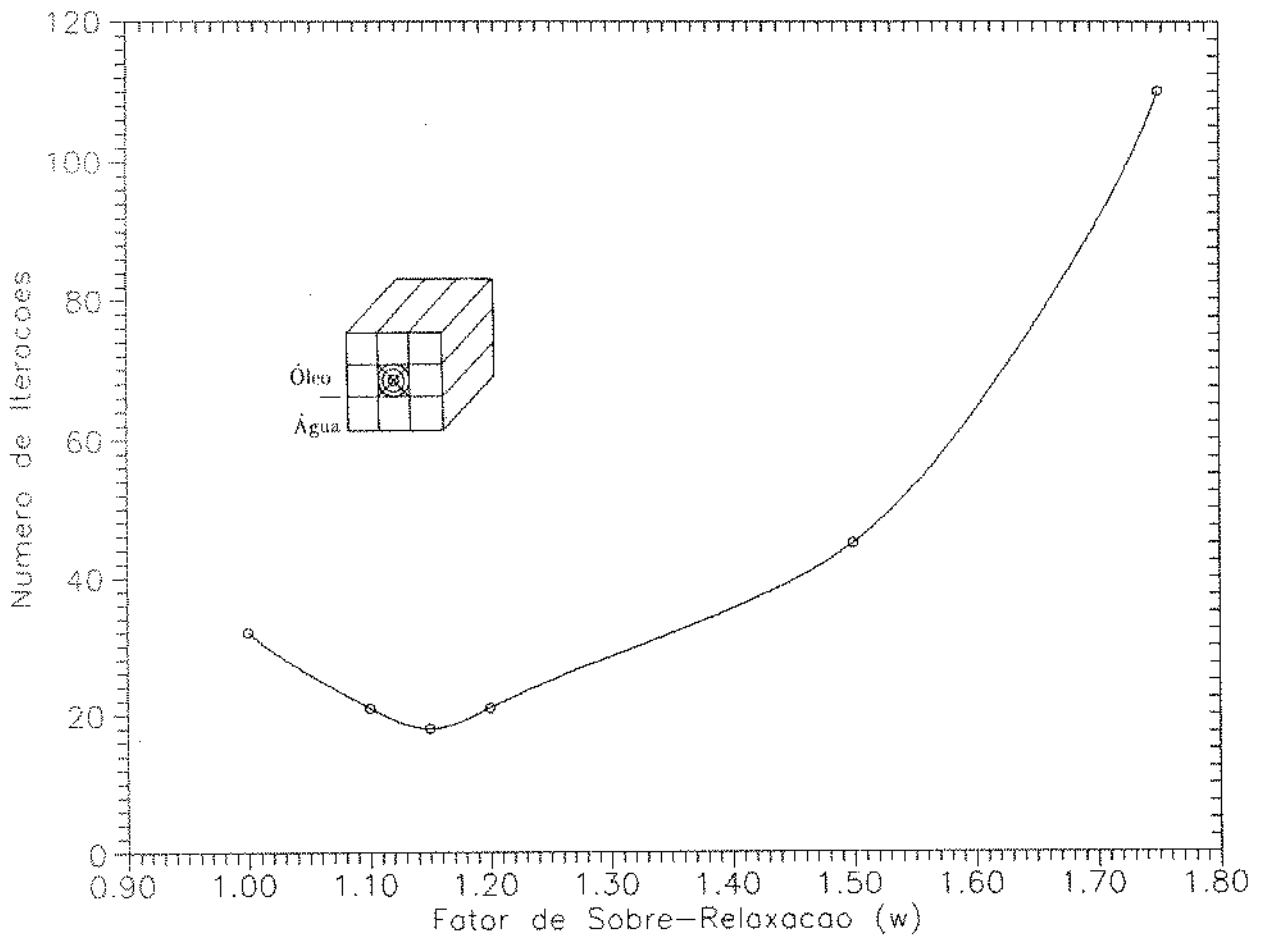


Figura 7.1: Fator de sobre-relaxação versus tempo

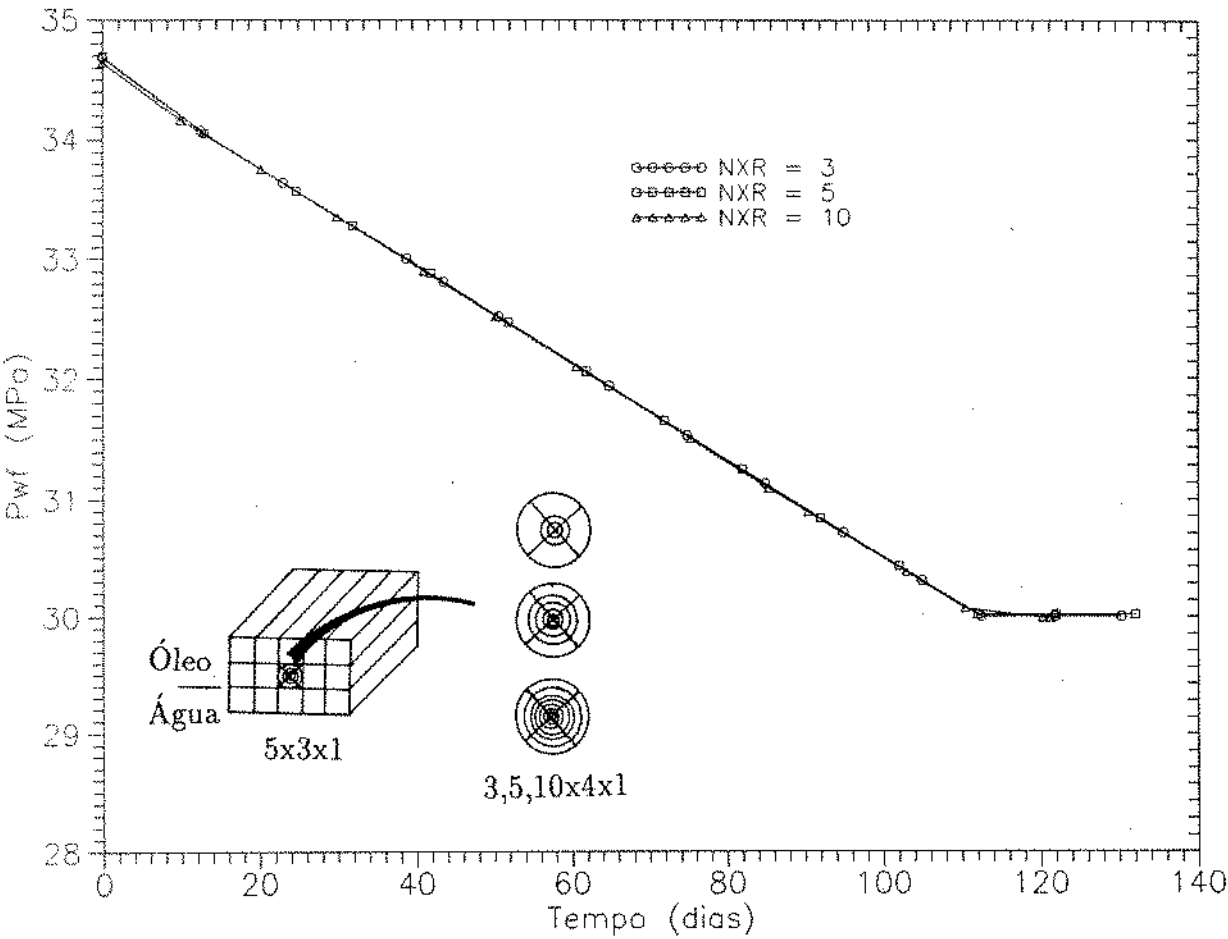


Figura 7.2: Refinamento radial (caso 1) - P_{wf} versus tempo

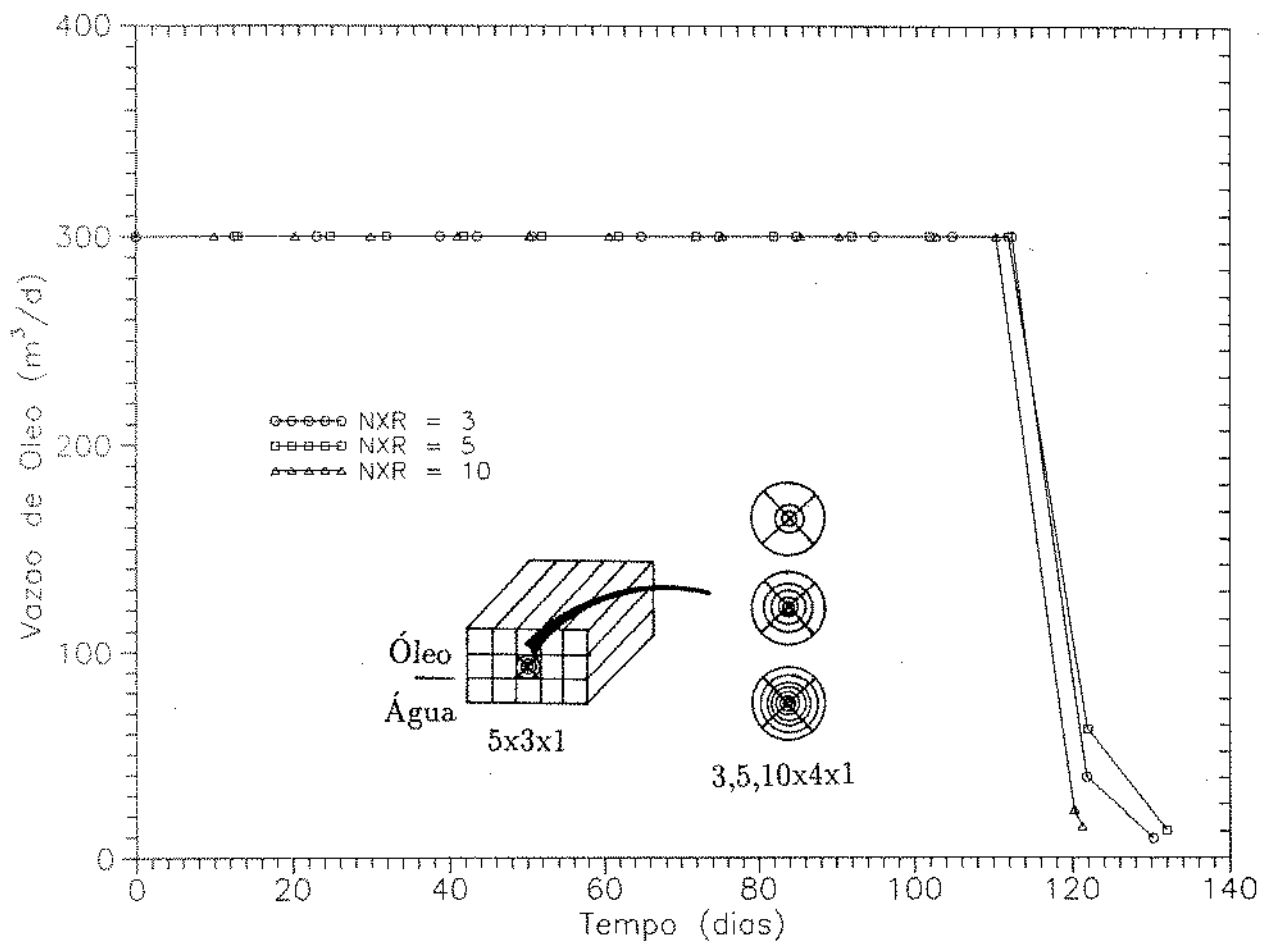


Figura 7.3: Refinamento radial (caso 1) - Q_o versus tempo

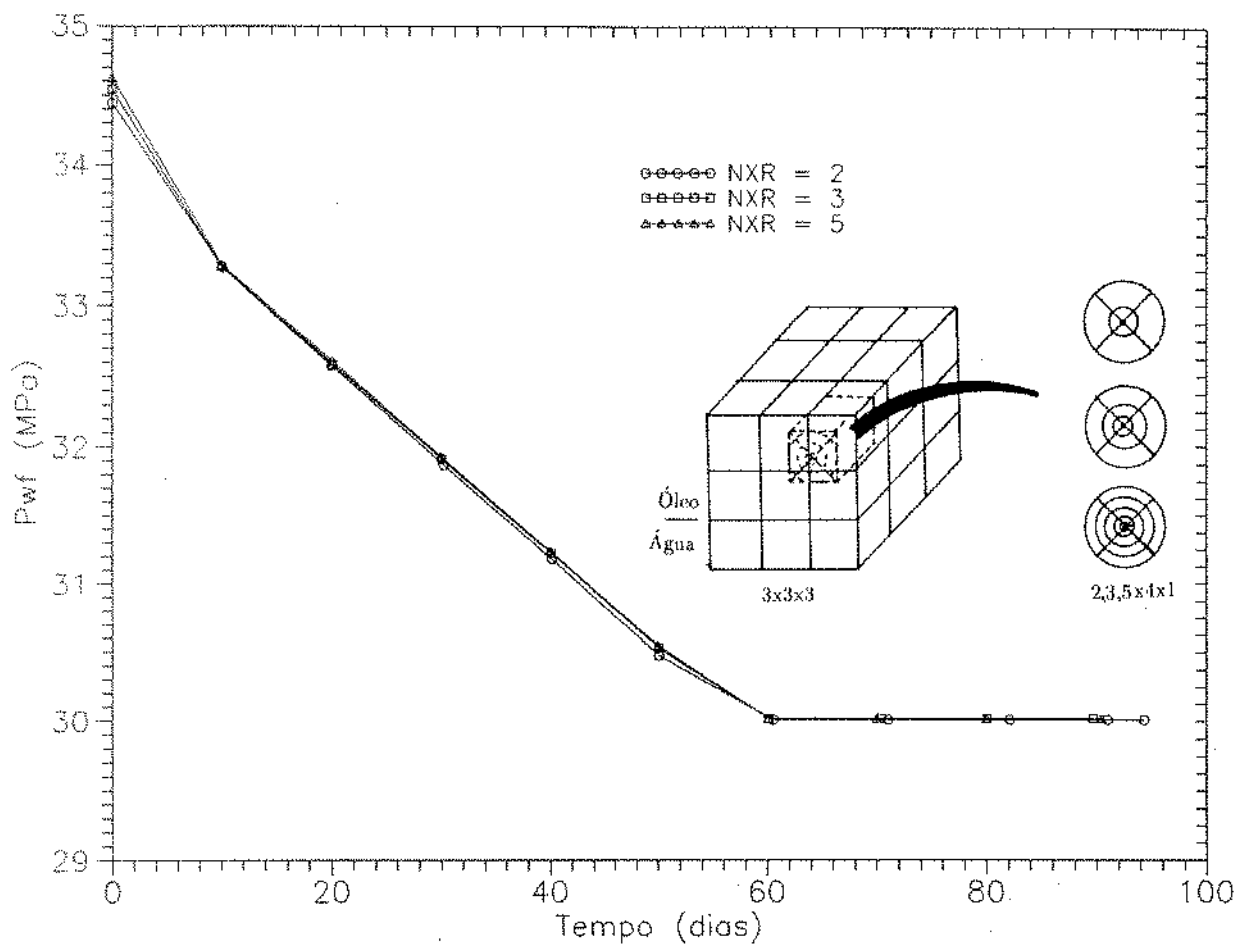


Figura 7.4: Refinamento radial (caso 2) - P_{wf} versus tempo

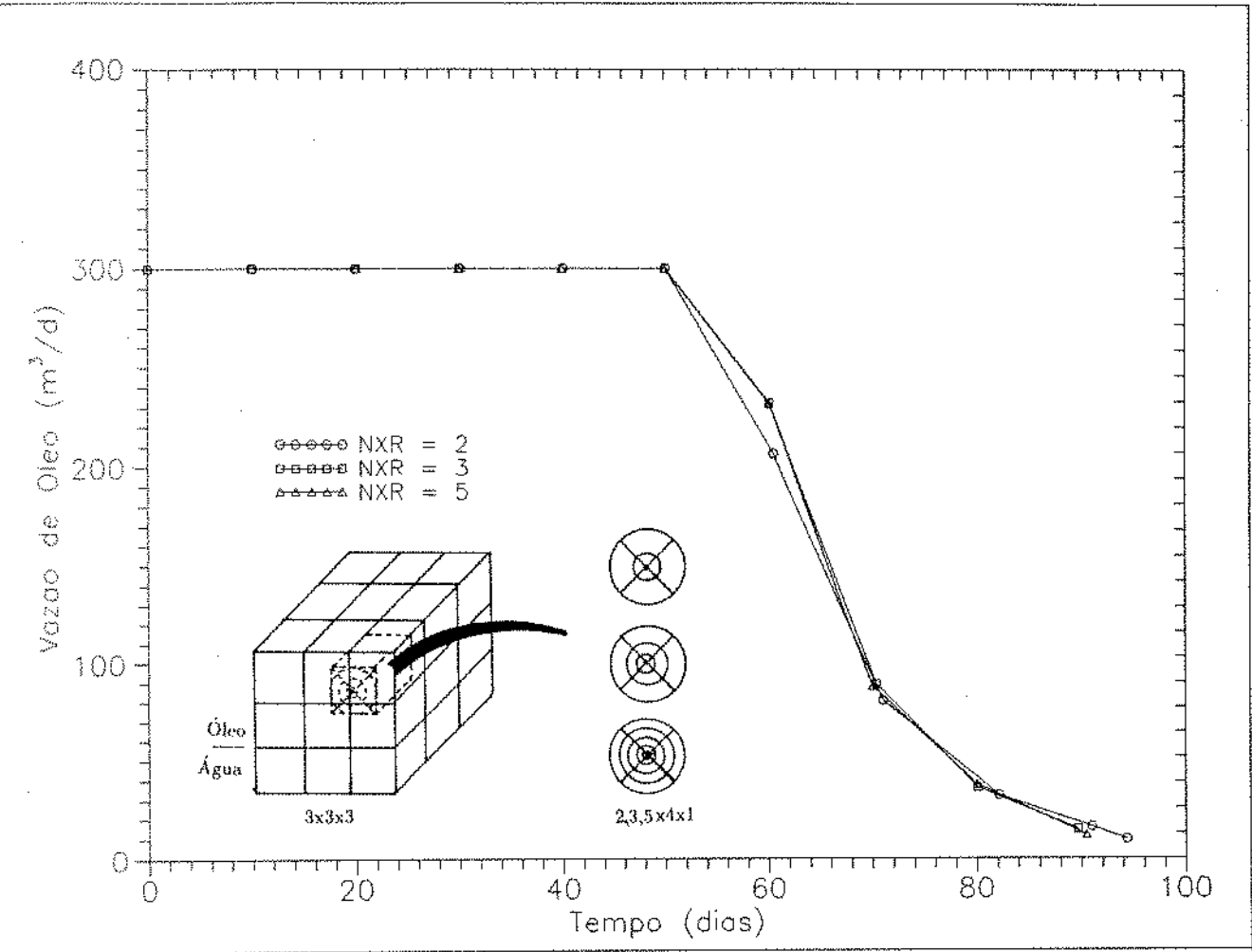


Figura 7.5: Refinamento radial (caso 2) - Q_o versus tempo

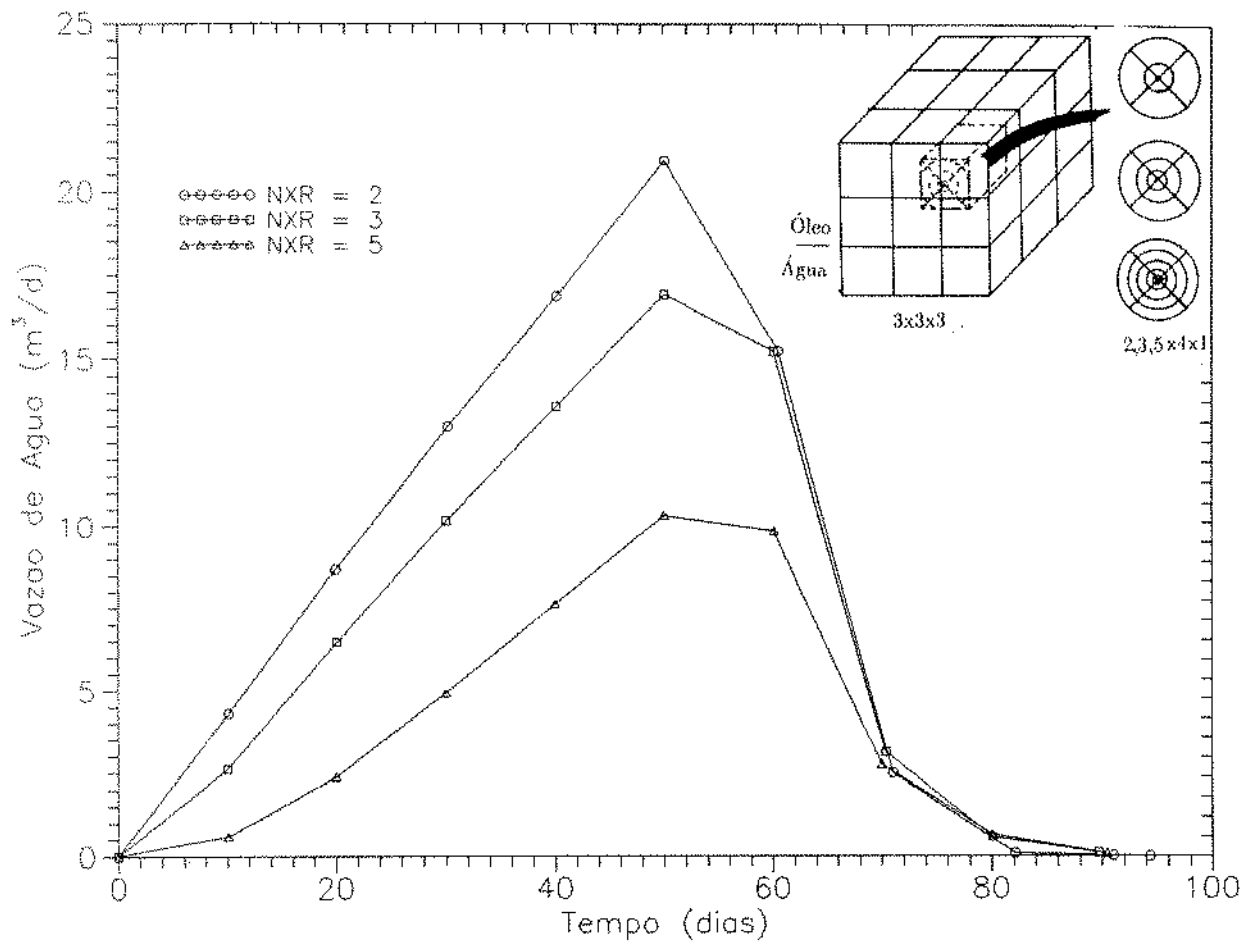


Figura 7.6: Refinamento radial (caso 2)- Q_w versus tempo

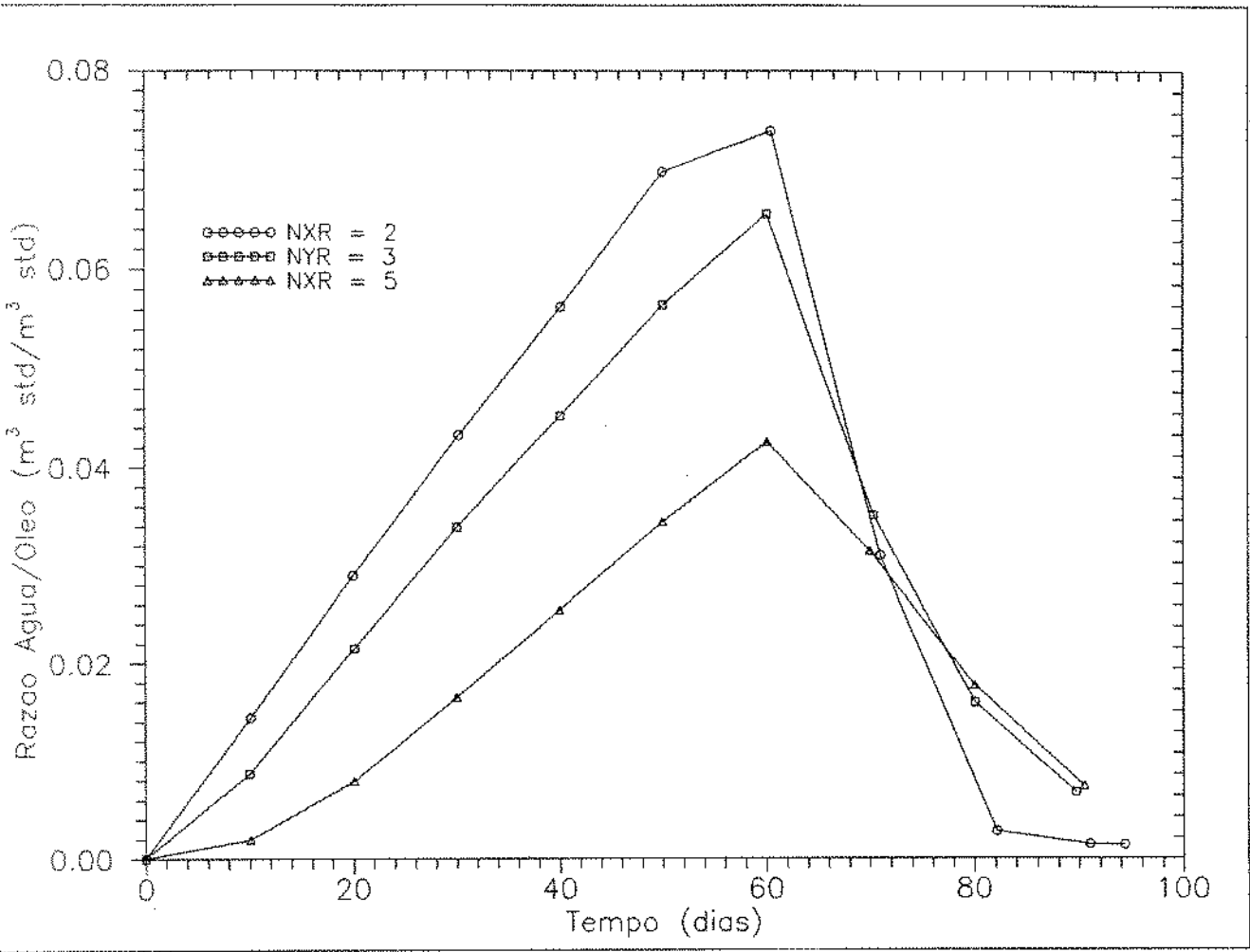


Figura 7.7: Refinamento radial (caso 2)- *RWO* versus tempo

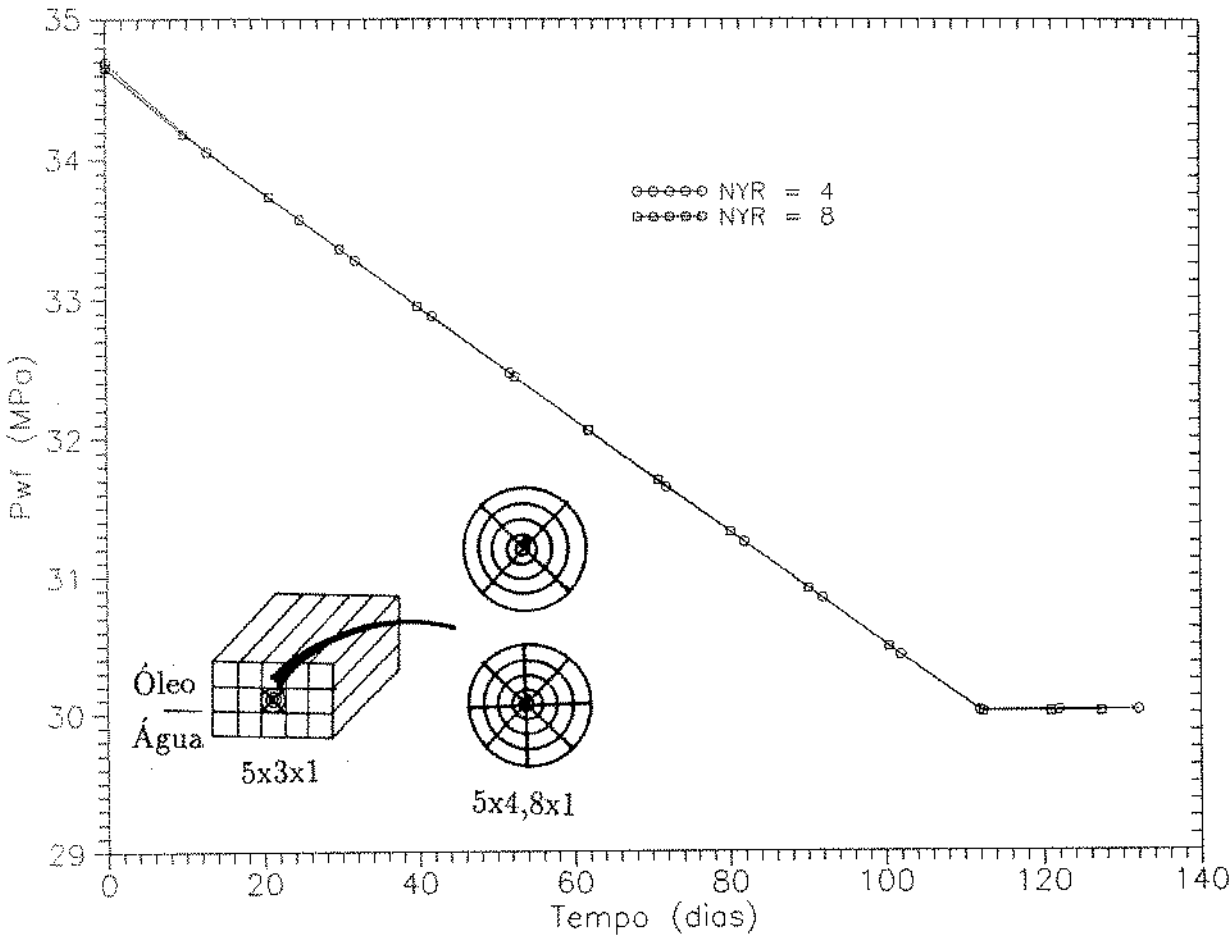


Figura 7.8: Refinamento angular - P_{wf} versus tempo

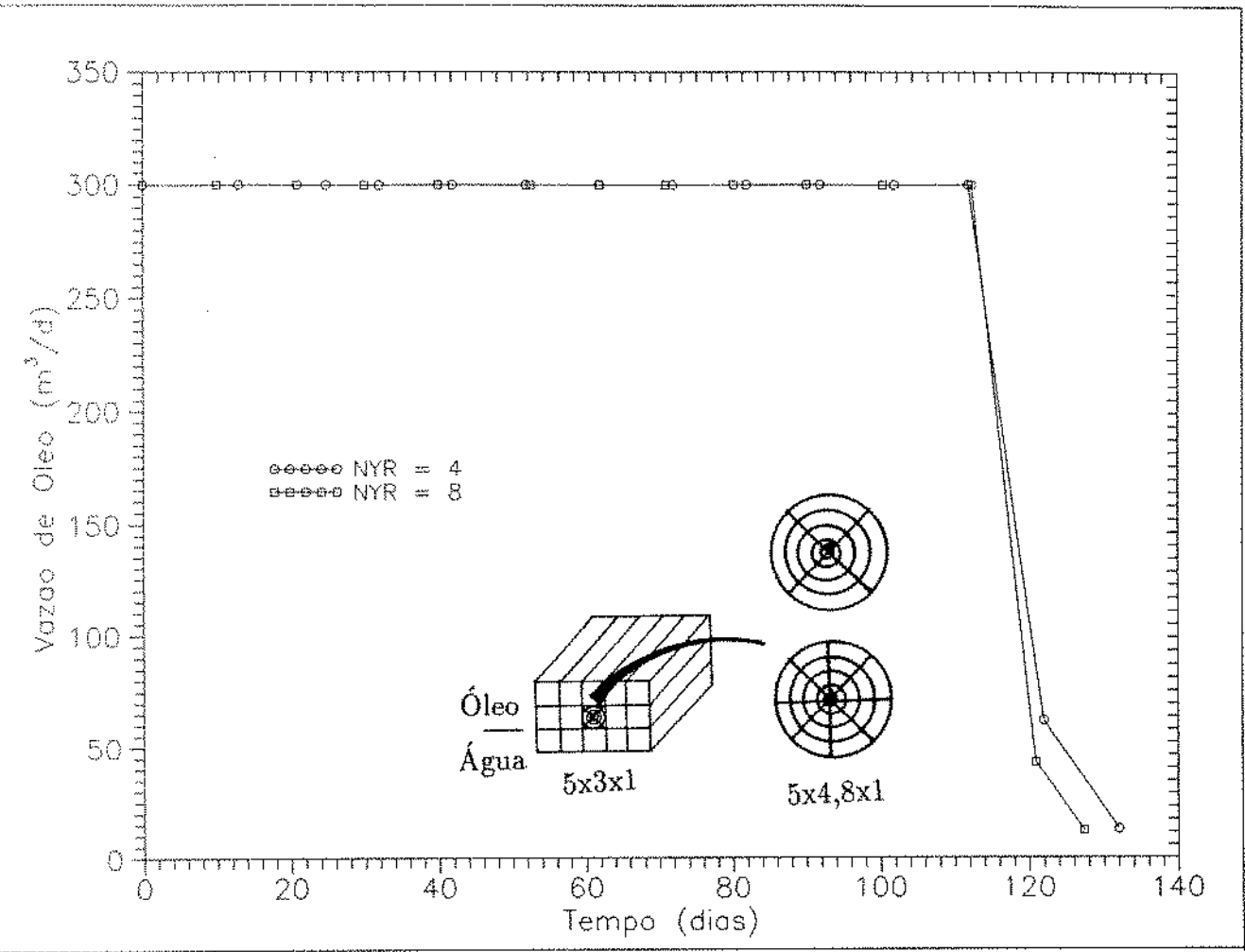


Figura 7.9: Refinamento angular - Q_o versus tempo

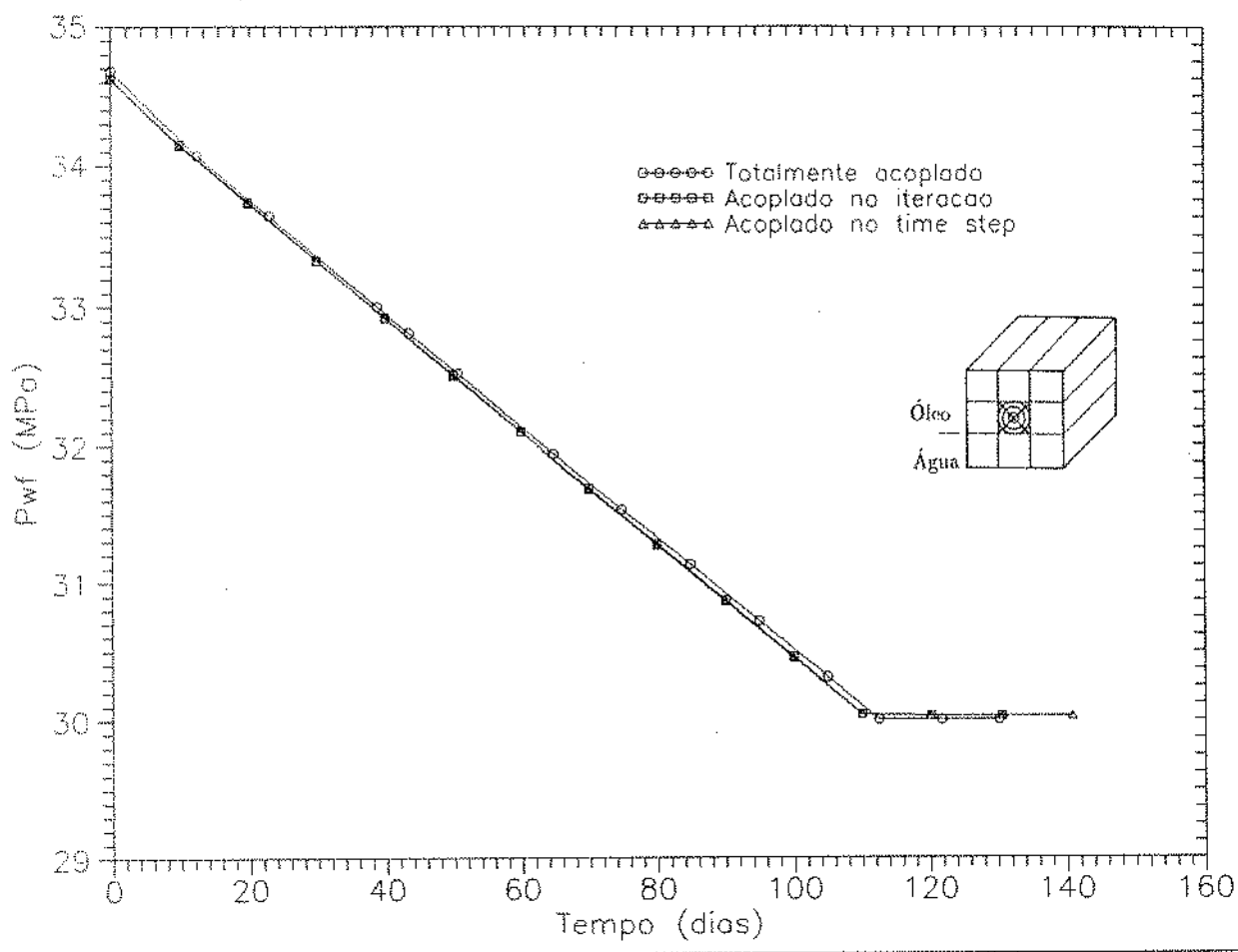


Figura 7.10: Comparação entre os modelos - P_{wf} versus tempo

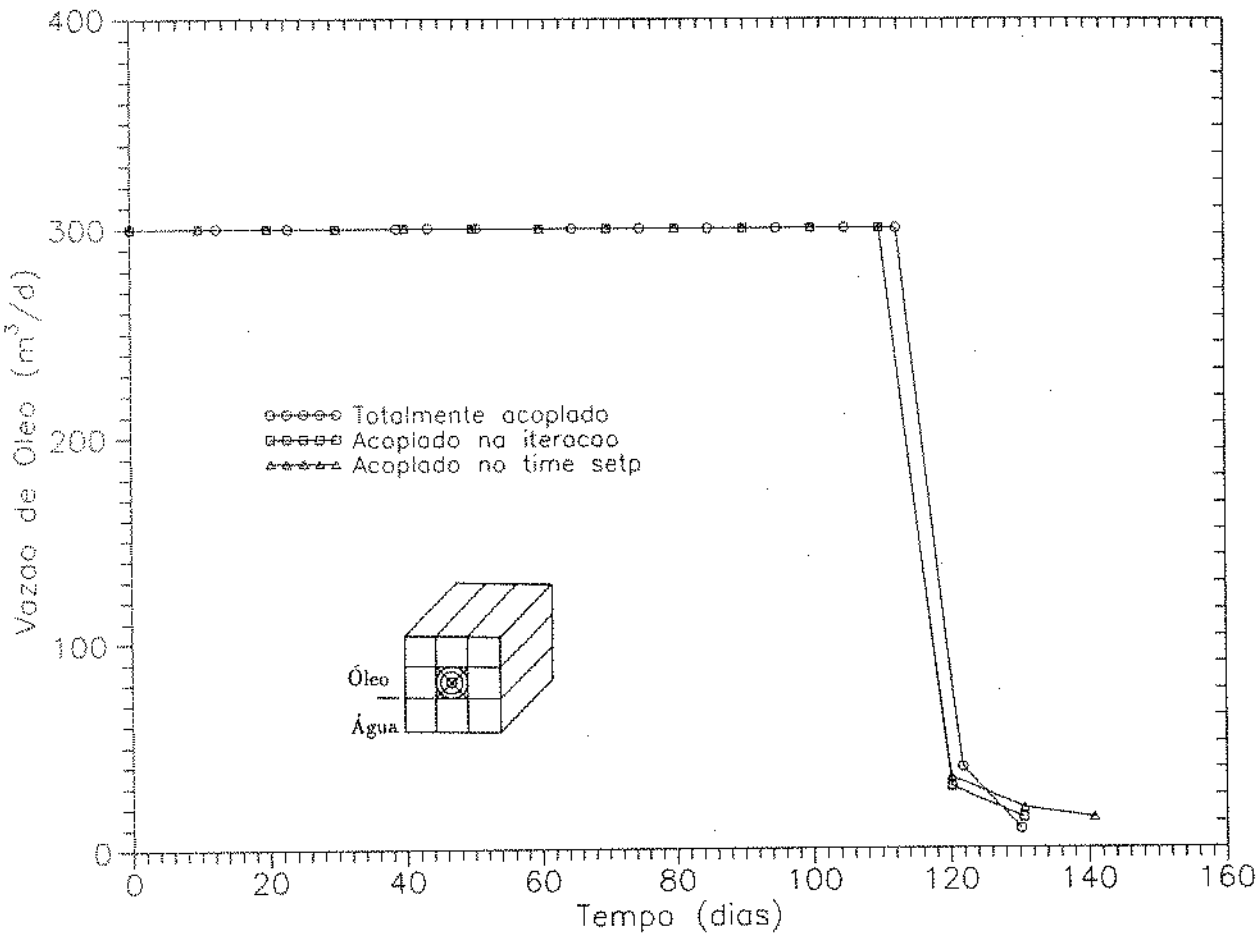


Figura 7.11: Comparação entre os modelos - Q_o versus tempo

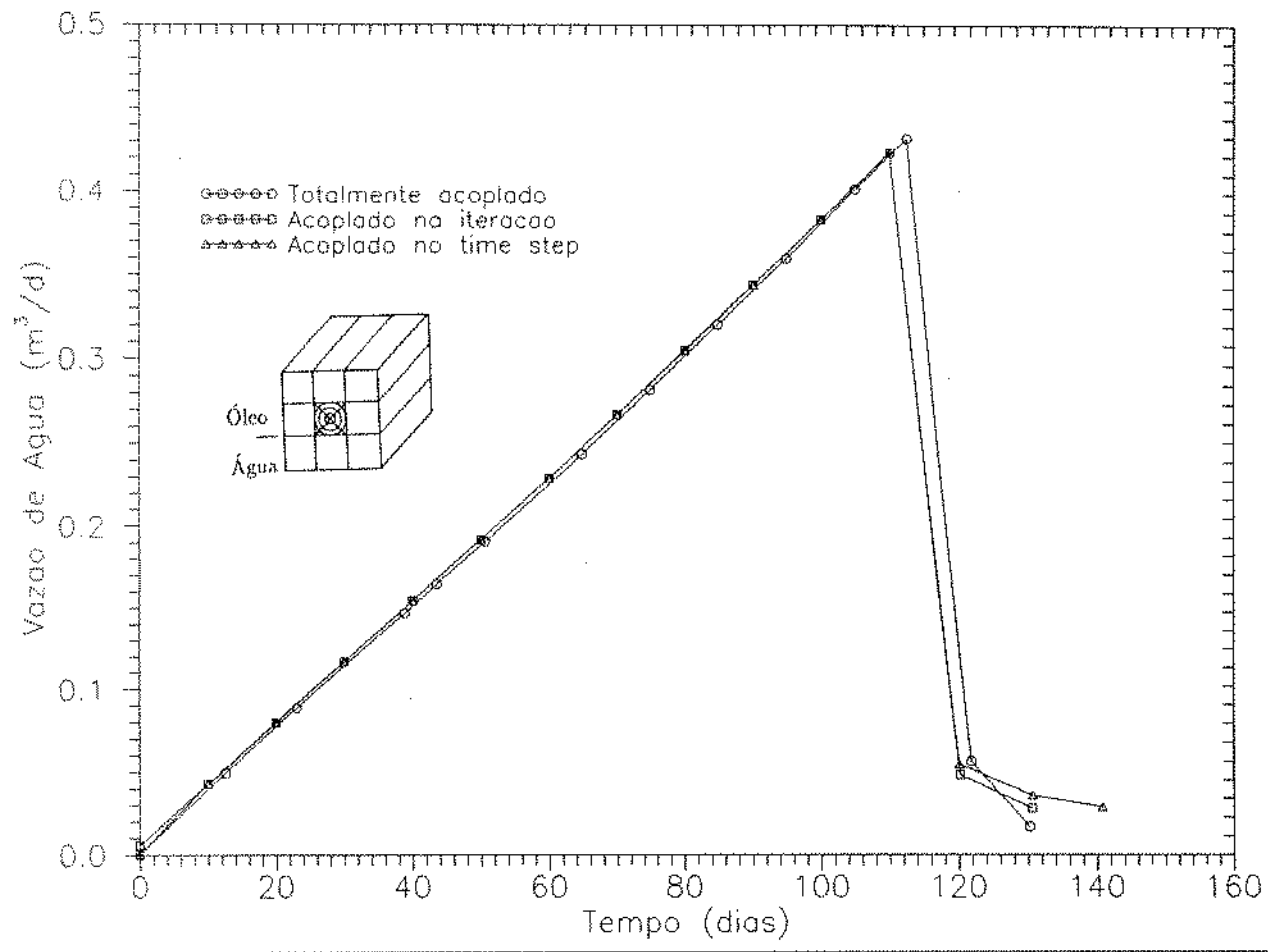


Figura 7.12: Comparação entre os modelos - Q_w versus tempo

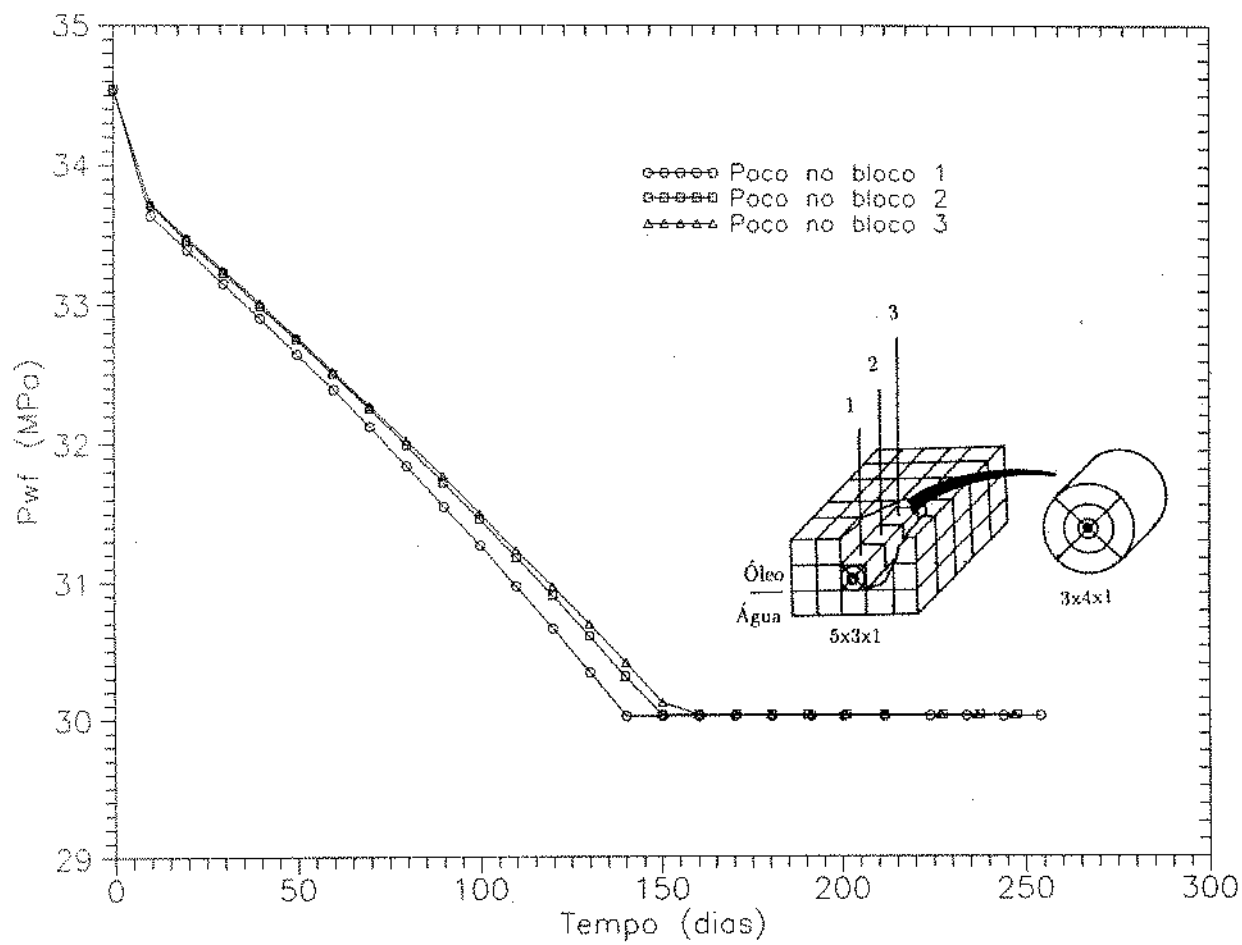


Figura 8.1: Posição do poço - P_{wf} versus tempo

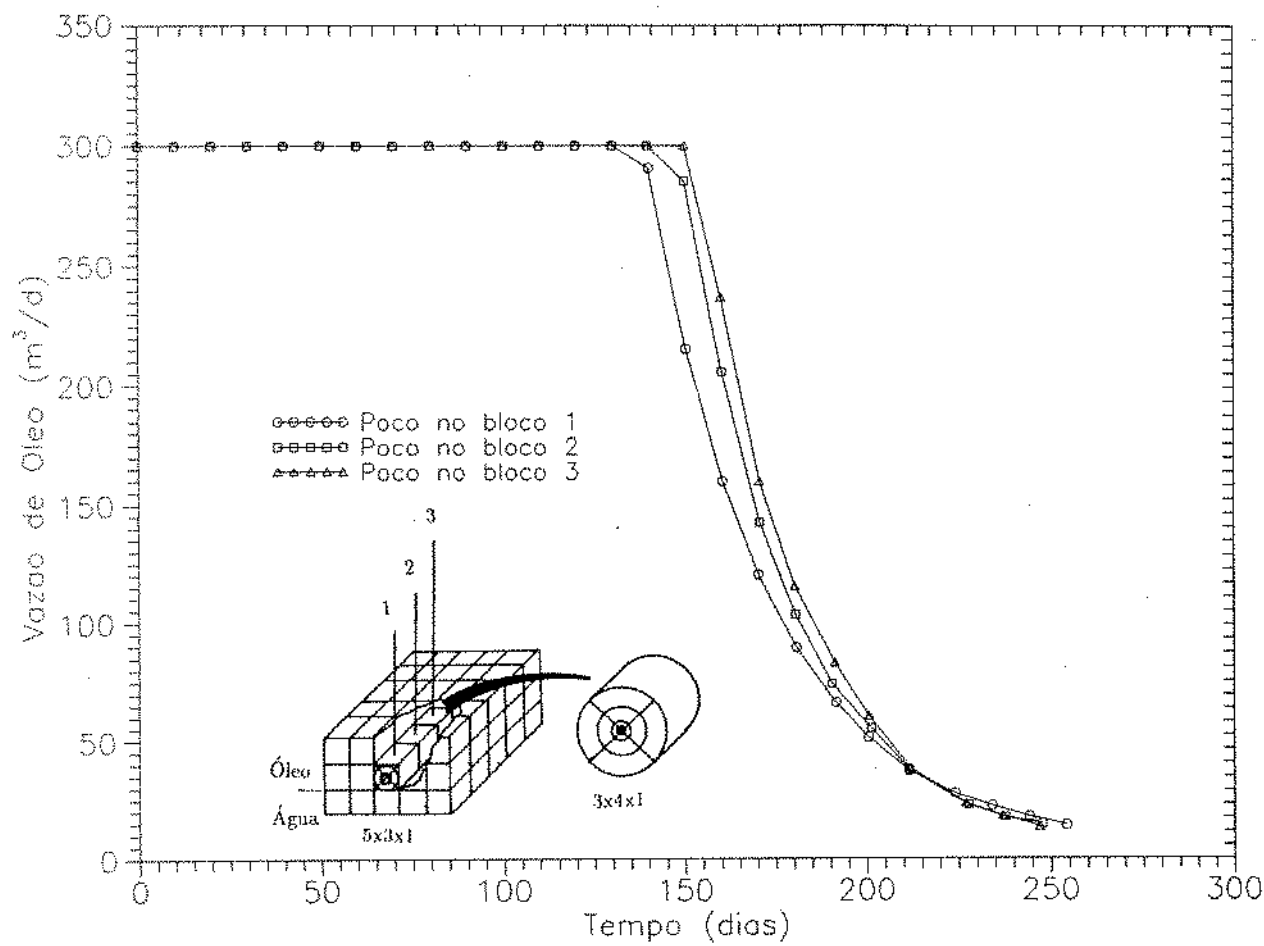


Figura 8.2: Posição do poço - Q_o versus tempo

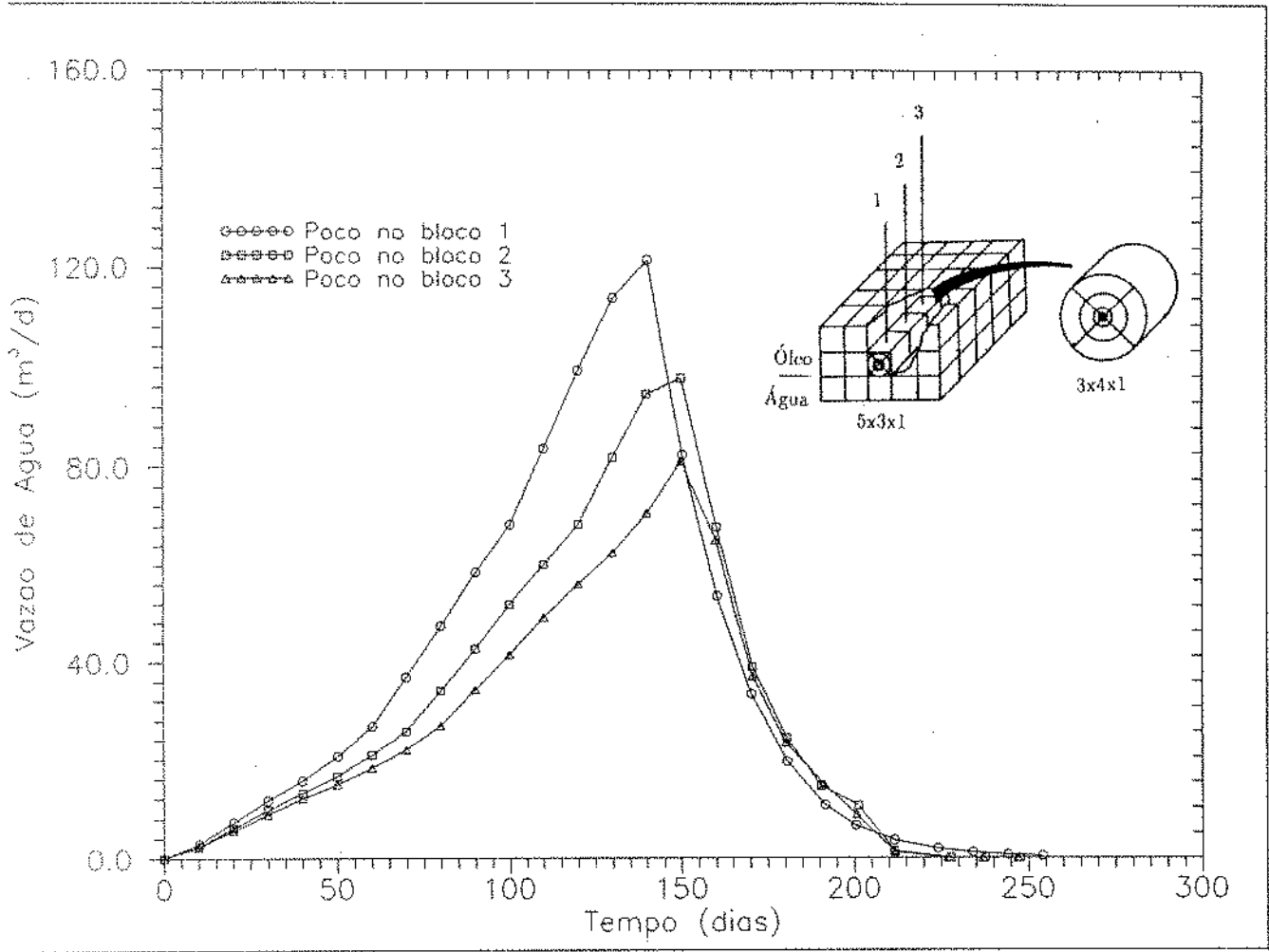


Figura 8.3: Posição do poço - Q_w versus tempo

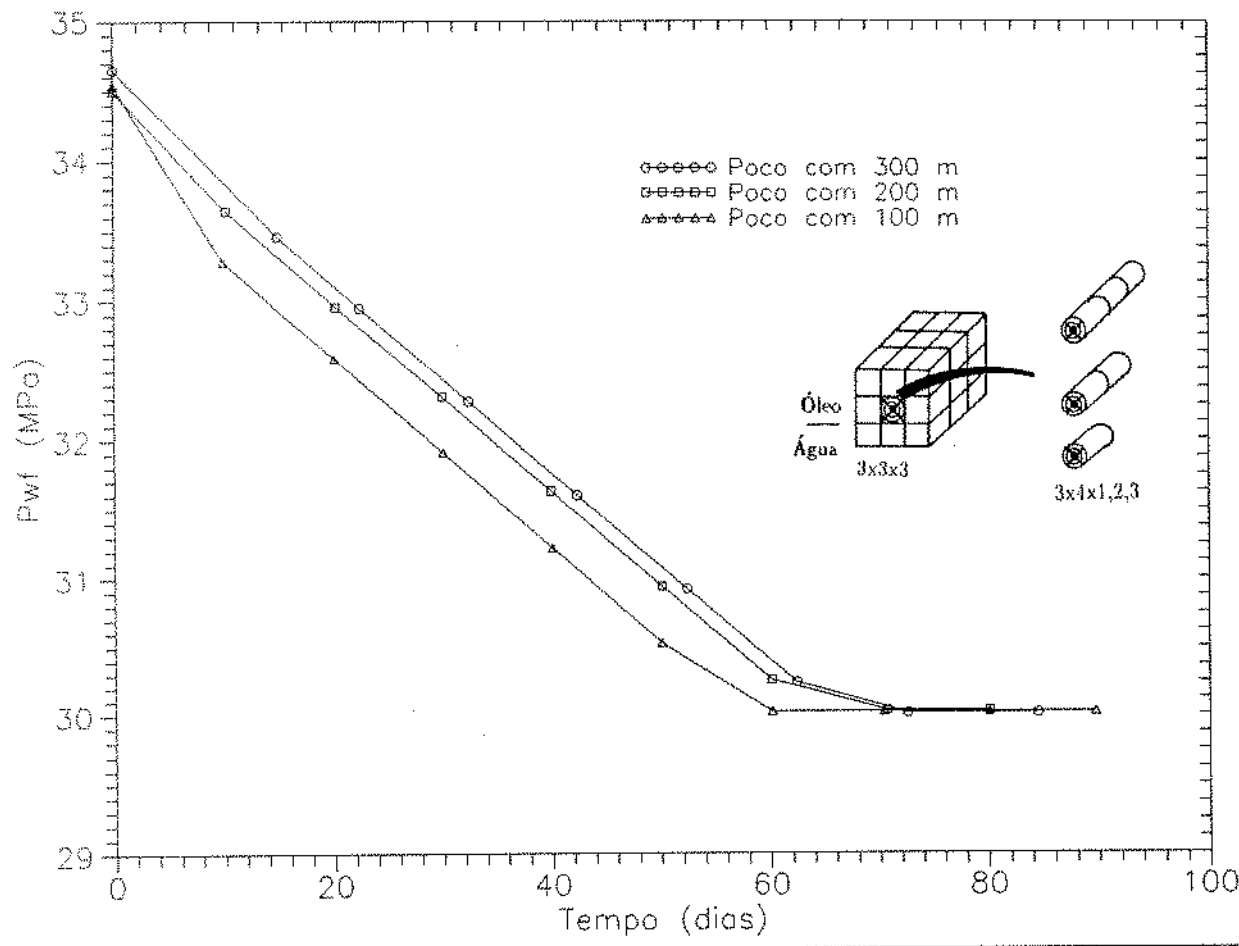


Figura 8.4: Comprimento do poço - P_{wf} versus tempo

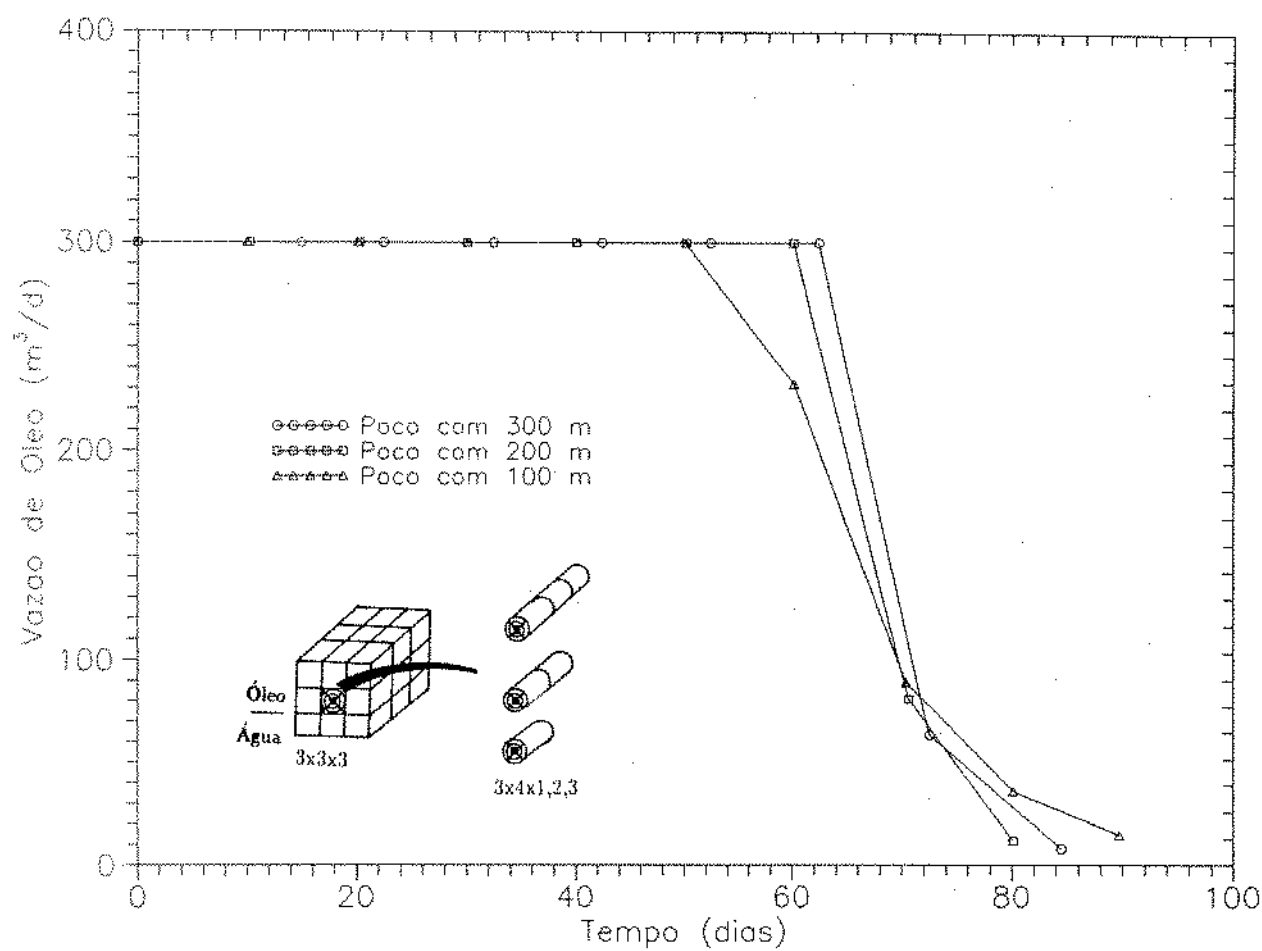


Figura 8.5: Comprimento do poço - Q_o versus tempo

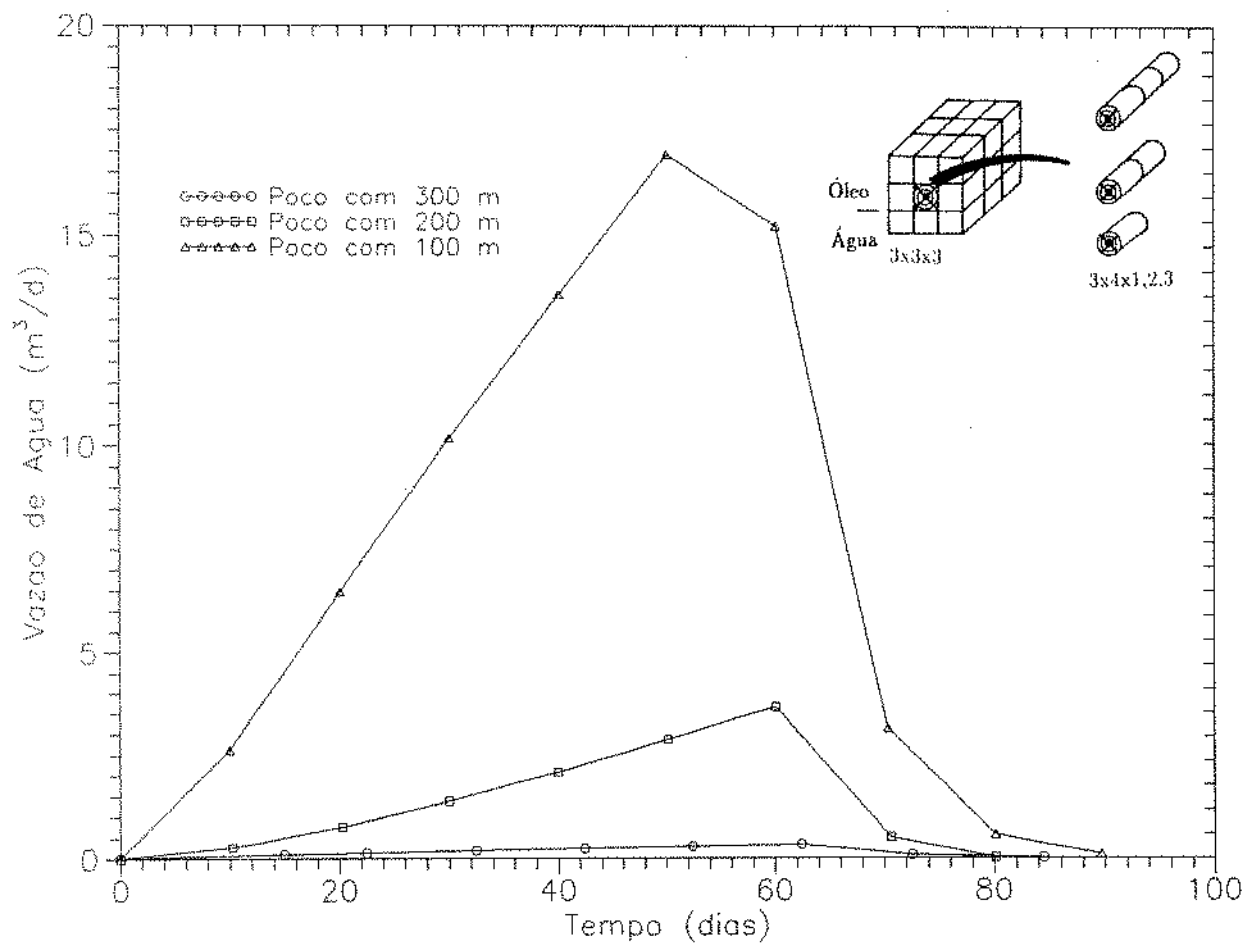


Figura 8.6: Comprimento do poço - Q_w versus tempo

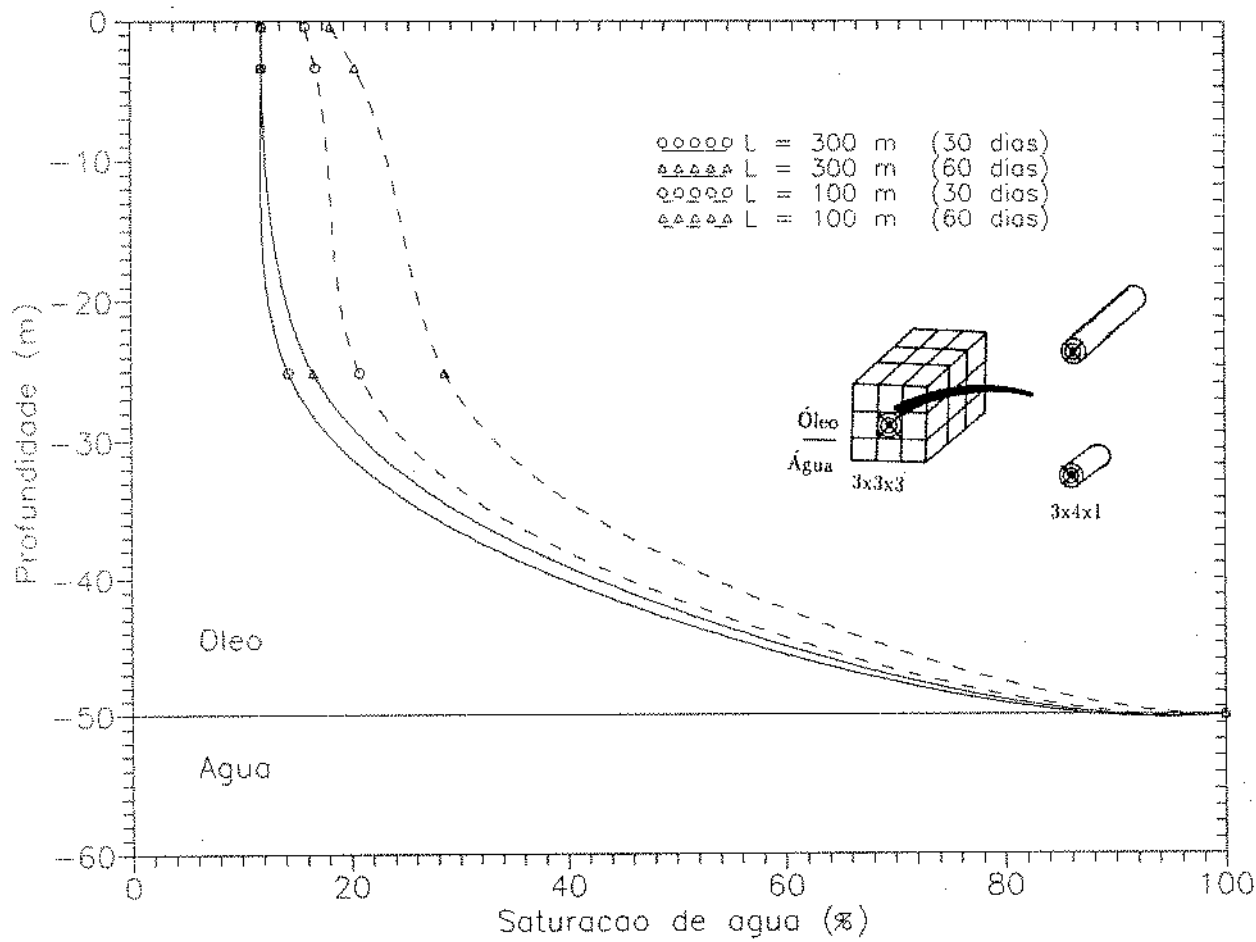


Figura 8.7: Perfil vertical de saturação de água

9 CONCLUSÕES E RECOMENDAÇÕES

As principais conclusões obtidas no desenvolvimento deste trabalho foram:

- Poço horizontal:
 - Com o avanço das técnicas de perfuração e completação, a utilização de poços horizontais surge como alternativa viável para a drenagem de reservatórios, onde a utilização de poços verticais convencionais é comprovadamente ineficaz.
- Refinamento local de malha e decomposição de domínios:
 - São ferramentas muito úteis, que podem ser utilizadas para economizar tempo e memória computacional, não só para a simulação de poços horizontais mas também para a representação de qualquer fenômeno localizados.
- Modelos desenvolvidos:
 - Tanto o modelo totalmente acoplado quanto os desacoplados - com acoplamento na iteração e no *time step* - fazem a representação do comportamento de reservatório com poço horizontal e da geometria do fluxo ao redor deste, de forma adequada e econômica;
 - Nos modelos desacoplados deve-se ter atenção especial com as tolerâncias usadas no processo de convergência;
 - O modelo totalmente acoplado tem maior aplicação em problemas que podem ser representados por malhas grosseiras;
 - Os modelos desacoplado podem ser implementados em simuladores convencionais.
- Métodos de solução do sistema linear de equações:
 - A utilização de métodos de solução de matrizes esparsas que trabalhem somente com os elementos não nulos trazem significativa economia de tempo;

- Os métodos iterativos para a solução da matriz Jacobiana são econômicos em termos de memória, porém muito mais lentos.

Como recomendações para o aprimoramento deste trabalho, sugerimos:

- Adaptação dos modelos desenvolvidos para computadores de grande porte, para simulação de malhas com maior refinamento;
- Adaptação de malhas esféricas para a representação do fluxo nas pontas do poço;
- Pesquisa de algoritmo mais otimizado para a solução de matrizes esparsas com elementos de interface de malhas.

10 REFERÊNCIAS

10.1 Poços Horizontais

1. Giger, F. M., Reiss, L. H. and Jourdan, A. P., "The Reservoir Engineering Aspects of Horizontal Drilling", paper SPE 13024 presented at the 59th Annual Technical Conference and Exhibition, Houston, September, 1984.
2. Giger, F. M., "Horizontal Wells Production Techniques in Heterogeneous Reservoirs", paper SPE 13710 presented at the Middle East Oil Technical Conference and Exhibition, Bahrain, March, 1985.
3. Reiss, L. H., "Horizontal Wells' Production After Five Years", paper SPE 14338 presented at the 60th Annual Technical Conference and Exhibition of the SPE, Las Vegas, September, 1985.
4. Giger, F. M., "Analytic 2-D Models of Water Cresting Before Breakthrough for Horizontal Wells", paper SPE 15378 presented at the 61th SPE Annual Technical Conference and Exhibition, New Orleans, October, 1986.
5. Chaperon, I., "Theoretical Study of Coning Toward Horizontal and Vertical Wells in Anisotropic Formations: Subcritical and Critical Rates", paper SPE 15377 presented at the 61th SPE Annual Technical Conference and Exhibition, New Orleans, October, 1986.
6. Sherrard, D. W., Brice, B. W. and MacDonald, D.G., "Application of Horizontal Wells at Prudhoe Bay", paper SPE 15376 presented at the 61th SPE Annual Technical Conference and Exhibition, New Orleans, October, 1986.
7. Karcher, B. J., Giger, F. M. and Combe, J., "Some Practical Formulas to Predict Horizontal Well Behavior", paper SPE 15430 presented at the 61th SPE Annual Technical Conference and Exhibition, New Orleans, October, 1986.

8. Lee, S. H., "Analysis of Productivity of Inclined Wells and its Implication on Finite Difference Reservoir Simulation", paper SPE 16002 presented at the Ninth Symposium on Reservoir Simulation, San Antonio, February, 1987.
9. Kossack, C. A., Kleppe, J. and Aasen, T., "Oil Production from the Troll Field: A Comparison of Horizontal and Vertical Wells", paper SPE 16869 presented at the 62th Annual Technical Conference and Exhibition of SPE, Dallas, September, 1987.
10. Joshi, S. D., "Augmentation of Well Productivity With Slant and Horizontal Wells", paper SPE 15375 presented at the 61th SPE Annual Technical Conference and Exhibition, New Orleans, October, 1986.
11. Goode, P. A., Thambynayagam, R. K. M., "Pressure Drawdown and Buildup Analysis Wells in Anisotropic Media", paper SPE 14250 presented at the 1985 SPE Annual Technical and Conference and Exhibition in Las Vegas, Sept. 22-25.
12. Lara, A. Q., "Estudo do Comportamento Transiente de Pressão em Poços Horizontais", tese de mestrado, Universidade Federal de Ouro Preto, março, 1986.
13. Odeh, S. A., "Transient Flow Behavior of Horizontal Wells: Pressure Drawdown and Buildup Analysis", paper SPE 18802 presented at the 1989 SPE California Regional Meeting, April 5-7.
14. Chang, M. M., "Simulation of Production from Wells With Horizontal-Slanted Laterals", Final Report, IIT Research Institute National for Petroleum and Energy Research, Bartlesville, Oklahoma, March, 1989.
15. Babu, D. K., Odeh, A. S., "Productivity of a Horizontal Well", paper SPE 18298, presented at the SPE Annual Technical Conference and Exhibition, Houston, October, 1988.
16. Ozkan, E., et al, "Horizontal-Well Pressure Analysis", paper SPE 16378, presented at the 1987 SPE California Regional Meeting, April, 1987.

17. Clonts, M. D., Ramey, H. J., " Pressure Transiente Analysis for Wells with Horizontal Drainholes ", paper SPE presented at the 56th California Regional Meeting in Dokland, C.A., April, 1986.

10.2 Refinamento local

18. Akbar, A., et al, "Numerical Simulation of Individual Wells in a Field Simulation Model", paper SPE 4073 presented at the SPE-AIME, 47th Annual Fall Meeting, San Aantonio, October 1972.
19. Rosenberg, D., "Local Mesh Refinement fo Finite Diference Methods", paper SPE 10974 presented at the 57th Fall Technical Conference and Exhibition of the SPE, New Orleans, 1982.
20. Quandale, P. and Besset, P., "The Use of Flexible for Reservoir Modeling", paper SPE 12239 presented at the SPE Symposium on Reservoir Simulation , San Francisco, November, 1983.
21. Heinemann, Gerken, G. and HantelmannM, G., "Using Local Grid Refinement in a Multiple-Application Reservoir Simulator", paper SPE 12255 presented at the SPE Symposium on Reservoir Simulation , San Francisco, November, 1983.
22. Pedrosa, O. A., " Use of Hybrid Grid in Reservoir Simulation " Tese de Doutorado, Stanford University, December, 1984.
23. Forsyth, P. A. and Sammon, P. H., "Local Mesh Refinement and Modelling of Faults and Pinchouts", paper SPE 13524 presented at the SPE 1985 Reservoir Simulation Symposium, Dallas, February, 1985.
24. Quandale, P. and Besset P., "Reduction of Grid Effects Due to Local Sub-Gridding in Simulations Using a Composite Grid", paper SPE 13527 presented at the SPE 1985 Reservoir Simulation Symposium, Dallas, February, 1985.

25. Han, D. K., Yan, C.Z. and Peng, L.T., "A More Flexible Approach of Dynamic Local Refinement for Reservoir Modeling", paper SPE 16014 presented at the Ninth SPE Symposium on Reservoir Simulation, San Antonio, February, 1987.
26. Wasserman, M. L., "Local Grid Refinement for Three-Dimensional Simulators", paper SPE 16013 presented at the Ninth SPE Symposium on Reservoir Simulation, San Antonio, February, 1987.
27. Ewing, R. E., Boyet, B. A. and Babu, D. K., "Efficient Use of Locally Refined Grids for Multifase Reservoir Simulation", paper SPE 18413 presented at the SPE Symposium on Reservoir Simulation, Houston, February, 1989.
28. Heinemann, Z. E., Brand, C., Munka, M. and Chen, Y. M., "Modeling Reservoir Geometry With Irregular Grids", paper SPE 18412 presented at the SPE Symposium on Reservoir Simulation, Houston, February, 1989.
29. Brand, C., Heinemann, Z. E., "A New Iterative Solution Technique for Reservoir Simulation Equation on Locally Refined Grids", paper SPE 18410, presented at the SPE Symposium on Reservoir Simulation in Houston, February, 1989.
30. Aziz, K. and Settari, A., "*Petroleum Reservoir Simulation*", Applied Science Publisher, 1979.
31. Peaceman, D. W., "*Fundamentals of Numerical Reservoir Simulation*", Elsevier Scientific Publishing Co., Amsterdam, 1977.
32. Crichlow, H. B., "*Modern Reservoir Engineering - a Simulation Approach*", Prentice Hall Inc., Englewood Cliffs, NJ, 1977.
33. Dake, L. P., "*Fundamentals of Reservoir Engineering*", Elsevier Scientific Publishing Co., Amsterdam, 1978.
34. Sherman, A. H., "*NSPIV - A Fortran for Sparse Gaussian Elimination with Partial Pivoting*", The University of Texas at Austin.

11 APÊNDICE A - Derivadas das Equações de Fluxo

Neste apêndice, são apresentadas todas as derivadas e as hipóteses usadas para a solução do problema na forma totalmente implícita.

As equações para o fluxo do óleo e da água são:

Equação para o óleo:

$$\begin{aligned} & T_{o_{m+1/2}} \underbrace{[(p_{m+1} - p_m) - \gamma_{o_{m+1/2}}(D_m - D_{m+1})]}_{POTOM2} + \\ & -T_{o_{m-1/2}} \underbrace{[(p_m - p_{m-1}) - \gamma_{o_{m-1/2}}(D_{m-1} - D_m)]}_{POTOM1} = \\ & \underbrace{\frac{V_{p_{ijk}}}{\Delta t} ([1 + cr(po - po^n)][1 - Sw]bo) - (Sobo)^n}_{ACO}_{ijk} + Q_o \end{aligned} \quad (A.1)$$

Equação para a água:

$$\begin{aligned} & T_{w_{m+1/2}} \underbrace{[(p_{m+1} - p_m) - (Pc_{m+1} - Pc_m) - \gamma_{w_{m+1/2}}(D_m - D_{m+1})]}_{POTWM2} + \\ & -T_{w_{m-1/2}} \underbrace{[(p_m - p_{m-1}) - (Pc_m - Pc_{m-1}) - \gamma_{w_{m-1/2}}(D_{m-1} - D_m)]}_{POTWM1} = \\ & \underbrace{\frac{V_{p_{ijk}}}{\Delta t} ([1 + cr(po - po^n)](Swbw) - (Sobo)^n)}_{ACW}_{ijk} + Q_w \end{aligned} \quad (A.2)$$

Como as equações acima estão escritas de forma simplificadas deve-se observar que:

- Os termos de fluxo devem ser tomados em todas direções, portanto, as transmisibilidades devem ser calculadas para todas direções, assim nesta representação m , representa os blocos i, j e k ,
 $m + 1/2$ representa as faces $i + 1/2, j + 1/2$ e $k + 1/2$ e
 $m - 1/2$ representa as faces $i - 1/2, j - 1/2$ e $k - 1/2$.

- $POTIM2$ representa a diferença de potencial de fluxo entre os blocos m e $m + 1$.
 $POTIM1$ representa a diferença de potencial de fluxo entre os blocos m e $m - 1$.
 l , pode ser óleo (o) ou água (w).
- M , representa as direções x , y ou z .
- Como é usado o esquema totalmente implícito, a maioria das variáveis são calculadas no time step $n + 1$. Para simplificar as equações, somente serão colocados os expoentes quando elas estiverem no time step n .
- ACI é o termo de acumulação para o óleo ou a água.
- p é a pressão na fase óleo.
- P_c é a pressão capilar óleo/água.

Portanto usando a nomenclatura simplificada as funções de resíduos podem ser escritas:

Óleo:

$$Fo = To_{m+1/2}POTOM2 - To_{m-1/2}POTOM1 - ACO - Qo \quad (A.3)$$

Água:

$$Fw = Tw_{m+1/2}POTWM2 - Tw_{m-1/2}POTWM1 - ACW - Qw \quad (A.4)$$

Derivadas das Funções Resíduos do Óleo - F_o

1. Em Relação a Pressão (Equação Geral)

$$\begin{aligned} \frac{\partial F_o}{\partial p} = & POTOM2 \frac{\partial T_{o_{m+1/2}}}{\partial p} + T_{o_{m+1/2}} \frac{\partial POTOM2}{\partial p} - POTOM1 \\ & \frac{T_{o_{m-1/2}}}{\partial p} - T_{o_{m-1/2}} \frac{\partial POTOM1}{\partial p} - \frac{\partial ACO}{\partial p} - \frac{\partial Q_o}{\partial p} \end{aligned} \quad (A.5)$$

2. Em Relação a p_{m-1}

$$\frac{\partial F_o}{\partial p_{m-1}} = -POTOM1 \frac{T_{o_{m-1/2}}}{\partial p_{m-1}} - T_{o_{m-1/2}} \frac{\partial POTOM1}{\partial p} \quad (A.6)$$

3. Em Relação a p_m

$$\begin{aligned} \frac{\partial F_o}{\partial p_m} = & POTOM2 \frac{\partial T_{o_{m+1/2}}}{\partial p_m} + T_{o_{m+1/2}} \frac{\partial POTOM2}{\partial p_m} - POTOM1 \\ & \frac{T_{o_{m-1/2}}}{\partial p_m} - T_{o_{m-1/2}} \frac{\partial POTOM1}{\partial p_m} - \frac{\partial ACO}{\partial p_m} - \frac{\partial Q_o}{\partial p_m} \end{aligned} \quad (A.7)$$

4. Em Relação a p_{m+1}

$$\frac{\partial F_o}{\partial p_{m+1}} = POTOM2 \frac{\partial T_{o_{m+1/2}}}{\partial p_{m+1}} + T_{o_{m+1/2}} \frac{\partial POTOM2}{\partial p_{m+1}} \quad (A.8)$$

1. Em Relação a Saturação (Equação Geral)

$$\frac{\partial F_o}{\partial S_w} = POTOM2 \frac{\partial T_{o_{m+1/2}}}{\partial S_w} + T_{o_{m+1/2}} \frac{\partial POTOM2}{\partial S_w} - POTOM1 \frac{T_{o_{m-1/2}}}{\partial S_w} - T_{o_{m-1/2}} \frac{\partial POTOM1}{\partial S_w} - \frac{\partial ACO}{\partial S_w} - \frac{\partial Q_o}{\partial S_w} \quad (A.9)$$

2. Em Relação a Sw_{m-1}

$$\frac{\partial F_o}{\partial Sw_{m-1}} = -POTOM1 \frac{T_{o_{m-1/2}}}{\partial Sw_{m-1}} \quad (A.10)$$

3. Em Relação a Sw_m

$$\frac{\partial F_o}{\partial Sw_m} = POTOM2 \frac{\partial T_{o_{m+1/2}}}{\partial Sw_m} - POTOM1 \frac{T_{o_{m-1/2}}}{\partial Sw_m} - \frac{\partial ACO}{\partial Sw_m} + \frac{\partial Q_o}{\partial Sw_m} \quad (A.11)$$

4. Em Relação a Sw_{m+1}

$$\frac{\partial F_o}{\partial Sw_{m+1}} = POTOM2 \frac{\partial T_{o_{m+1/2}}}{\partial Sw_{m+1}} \quad (A.12)$$

Derivadas das Funções Resíduos do Água - Fw

1. Em Relação a Pressão (Equação Geral)

$$\frac{\partial Fw}{\partial p} = POTWM2 \frac{\partial Tw_{m+1/2}}{\partial p} + Tw_{m+1/2} \frac{\partial POTWM2}{\partial p} - POTWM1 \frac{Tw_{m-1/2}}{\partial p} - Tw_{m-1/2} \frac{\partial POTWM1}{\partial p} - \frac{\partial ACW}{\partial p} - \frac{\partial Qw}{\partial p} \quad (A.13)$$

2. Em Relação a p_{m-1}

$$\frac{\partial Fw}{\partial p_{m-1}} = -POTWM1 \frac{Tw_{m-1/2}}{\partial p_{m-1}} - Tw_{m-1/2} \frac{\partial POTWM1}{\partial p_{m-1}} \quad (A.14)$$

3. Em Relação a p_m

$$\frac{\partial Fw}{\partial p_m} = POTWM2 \frac{\partial Tw_{m+1/2}}{\partial p_m} + Tw_{m+1/2} \frac{\partial POTWM2}{\partial p_m} - POTWM1 \frac{Tw_{m-1/2}}{\partial p_m} - Tw_{m-1/2} \frac{\partial POTWM1}{\partial p_m} - \frac{\partial ACW}{\partial p_m} - \frac{\partial Qw}{\partial p_m} \quad (A.15)$$

4. Em Relação a p_{m+1}

$$\frac{\partial Fw}{\partial p_{m+1}} = POTWM2 \frac{\partial Tw_{m+1/2}}{\partial p_{m+1}} + Tw_{m+1/2} \frac{\partial POTWM2}{\partial p_{m+1}} \quad (A.16)$$

1. Em Relação a Saturação (Equação Geral)

$$\begin{aligned} \frac{\partial F_w}{\partial S_w} = & POTWM2 \frac{\partial T_{w_{m+1/2}}}{\partial S_w} + T_{w_{m+1/2}} \frac{\partial POTWM2}{\partial S_w} + \\ & - POTWM1 \frac{T_{w_{m-1/2}}}{\partial S_w} - T_{w_{m-1/2}} \frac{\partial POTWM1}{\partial S_w} + \\ & - \frac{\partial ACW}{\partial S_w} - \frac{\partial Q_w}{\partial S_w} \end{aligned} \quad (A.17)$$

2. Em Relação a $S_{w_{m-1}}$

$$\frac{\partial F_w}{\partial S_{w_{m-1}}} = -POTWM1 \frac{T_{w_{m-1/2}}}{\partial S_{w_{m-1}}} - T_{w_{m-1/2}} \frac{\partial POTWM1}{\partial S_{w_{m-1}}} \quad (A.18)$$

3. Em Relação a S_{w_m}

$$\begin{aligned} \frac{\partial F_w}{\partial S_{w_m}} = & POTWM2 \frac{\partial T_{w_{m+1/2}}}{\partial S_{w_m}} + T_{w_{m+1/2}} \frac{\partial POTWM2}{\partial S_{w_m}} + \\ & - POTWM1 \frac{T_{w_{m-1/2}}}{\partial S_{w_m}} - T_{w_{m-1/2}} \frac{\partial POTWM1}{\partial S_{w_m}} + \\ & - \frac{\partial ACW}{\partial S_{w_m}} + \frac{\partial Q_w}{\partial S_{w_m}} \end{aligned} \quad (A.19)$$

4. Em Relação a $S_{w_{m+1}}$

$$\frac{\partial F_w}{\partial S_{w_{m+1}}} = POTWM2 \frac{\partial T_{w_{m+1/2}}}{\partial S_{w_{m+1}}} + T_{w_{m+1/2}} \frac{\partial POTWM2}{\partial S_{w_{m+1}}} \quad (A.20)$$

Derivadas dos Potenciais em Relação a Pressão

• *POTOM1* ou *POTWM1*

$$\begin{aligned} POTOM1 = & (p_m - p_{m-1}) - [w\gamma o_m + (1-w)\gamma o_{m-1}] \\ & (D_{m-1} - D_m) \end{aligned} \quad (A.21)$$

$$\begin{aligned} POTWM1 = & (p_m - p_{m-1}) - (Pc_m - Pc_{m-1}) + \\ & -[w\gamma w_m + (1-w)\gamma w_{m-1}](D_{m-1} - D_m) \end{aligned} \quad (A.22)$$

$$\frac{\partial POTOM1}{\partial p_{m-1}} = -1 - (1-w) \frac{\partial \gamma l_{m-1}}{\partial p_{m-1}} (D_{m-1} - D_m) \quad (A.23)$$

$$\frac{\partial POTOM1}{\partial p_m} = 1 - w \frac{\partial \gamma l_m}{\partial p_m} (D_{m-1} - D_m) \quad (A.24)$$

$$\frac{\partial POTOM1}{\partial p_{m+1}} = 0 \quad (A.25)$$

• *POTOM2* ou *POTWM2*

$$\begin{aligned} POTOM2 = & (p_{m+1} - p_m) - [w\gamma o_{m+1} + (1-w)\gamma o_m] \\ & (D_m - D_{m+1}) \end{aligned} \quad (A.26)$$

$$\begin{aligned}
POTWM2 = & (p_{m+1} - p_m) - (Pc_{m+1} - Pc_m) + \\
& -[w\gamma w_{m+1} + (1-w)\gamma w_m]](D_m - D_{m+1})
\end{aligned} \tag{A.27}$$

$$\frac{\partial POTIM2}{\partial p_{m-1}} = 0 \tag{A.28}$$

$$\frac{\partial POTIM2}{\partial p_m} = -1 - (1-w)\frac{\partial \gamma l_m}{\partial p_m}(D_m - D_{m+1}) \tag{A.29}$$

$$\frac{\partial POTIM2}{\partial p_{m+1}} = 1 - w\frac{\partial \gamma l_{m+1}}{\partial p_{m+1}}(D_m - D_{m+1}) \tag{A.30}$$

Derivadas dos Potenciais em Relação a Saturação

- *POTOM1* ou *POTOM2*

$$\frac{\partial POTOM1,2}{\partial Sw_{m-1}} = \frac{\partial POTOM1,2}{\partial Sw_m} = \frac{\partial POTOM1,2}{\partial Sw_{m+1}} = 0 \quad (A.31)$$

- *POTWM1*

$$\frac{\partial POTWM1}{\partial Sw_{m-1}} = \frac{\partial Pc_{m-1}}{\partial Sw_{m-1}} \quad (A.32)$$

$$\frac{\partial POTWM1}{\partial Sw_m} = -\frac{\partial Pc_m}{\partial Sw_m} \quad (A.33)$$

$$\frac{\partial POTWM1}{\partial Sw_{m+1}} = 0 \quad (A.34)$$

- *POTWM2*

$$\frac{\partial POTWM2}{\partial Sw_{m-1}} = 0 \quad (A.35)$$

$$\frac{\partial POTWM2}{\partial Sw_m} = \frac{\partial Pc_m}{\partial Sw_m} \quad (A.36)$$

$$\frac{\partial POTWM2}{\partial Sw_{m+1}} = -\frac{\partial Pc_{m+1}}{\partial Sw_{m+1}} \quad (A.37)$$

Derivadas das Transmissibilidade

Convenção do sentido de fluxo para as duas malhas são:

coordenada m	sentido do fluxo	
	A	B
i	contrário ao eixo x	sentido do eixo x
j	sentido do eixo y	contrário ao eixo y
k	sentido do eixo z	contrário ao eixo z

Derivadas em relação a pressão

As derivadas em relação a pressão são iguais para todos os eixos e qualquer sentido de fluxo.

$$\frac{\partial Tl_{m+1/2}}{\partial p_{m-1}} = 0 \quad (\text{A.38})$$

$$\frac{\partial Tl_{m+1/2}}{\partial p_m} = (1-w)Tl_{m+1/2} \left[\frac{1}{bl_{m+1/2}} \frac{dbl_m}{dp_m} - \left[\frac{1}{\mu l_{m+1/2}} \frac{d\mu l_m}{dp_m} \right] \right] \quad (\text{A.39})$$

$$\frac{\partial Tl_{m+1/2}}{\partial p_{m+1}} = wTl_{m+1/2} \left[\frac{1}{bl_{m+1/2}} \frac{dbl_{m+1}}{dp_{m+1}} - \left[\frac{1}{\mu l_{m+1/2}} \frac{d\mu l_{m+1}}{dp_{m+1}} \right] \right] \quad (\text{A.40})$$

$$\frac{\partial Tl_{m-1/2}}{\partial p_{m-1}} = (1-w)Tl_{m-1/2} \left[\frac{1}{bl_{m-1/2}} \frac{dbl_{m-1}}{dp_{m-1}} - \left[\frac{1}{\mu l_{m-1/2}} \frac{d\mu l_{m-1}}{dp_{m-1}} \right] \right] \quad (\text{A.41})$$

$$\frac{\partial Tl_{m-1/2}}{\partial p_m} = (1-w)Tl_{m-1/2} \left[\frac{1}{bl_{m-1/2}} \frac{dbl_m}{dp_m} - \left[\frac{1}{\mu l_{m-1/2}} \frac{d\mu l_m}{dp_m} \right] \right] \quad (\text{A.42})$$

$$\frac{\partial Tl_{m-1/2}}{\partial p_{m+1}} = 0 \quad (\text{A.43})$$

Derivadas em relação a saturação

1. Direção x

(a) Sentido A

derivada	face $i - 1/2$	face $i + 1/2$
$\frac{\partial Tl}{\partial Sw_{i-1}}$	0	0
$\frac{\partial Tl}{\partial Sw_i}$	$FF_{i-1/2}(\frac{bl}{\mu l})_{i-1/2} \frac{dkrl_i}{dSw_i}$	0
$\frac{\partial Tl}{\partial Sw_{i+1}}$	0	$FF_{i-1/2}(\frac{bl}{\mu l})_{i-1/2} \frac{dkrl_{i+1}}{dSw_{i+1}}$

(b) Sentido B

derivada	face $i - 1/2$	face $i + 1/2$
$\frac{\partial Tl}{\partial Sw_{i-1}}$	$FF_{i-1/2}(\frac{bl}{\mu l})_{i-1/2} \frac{dkrl_{i-1}}{dSw_{i-1}}$	0
$\frac{\partial Tl}{\partial Sw_i}$	0	$FF_{i+1/2}(\frac{bl}{\mu l})_{i+1/2} \frac{dkrl_{i-1}}{dSw_{i-1}}$
$\frac{\partial Tl}{\partial Sw_{i+1}}$	0	0

2. Direção y e z

(a) Sentido A

derivada	face $m - 1/2$	face $m + 1/2$
$\frac{\partial Tl}{\partial Sw_{m-1}}$	$FF_{m-1/2}(\frac{bl}{\mu l})_{m-1/2} \frac{dkrl_{m-1}}{dSw_{m-1}}$	0
$\frac{\partial Tl}{\partial Sw_m}$	0	$FF_{m+1/2}(\frac{bl}{\mu l})_{m+1/2} \frac{dkrl_m}{dSw_m}$
$\frac{\partial Tl}{\partial Sw_{m+1}}$	0	0

(b) Sentido B

derivada	face $m - 1/2$	face $m + 1/2$
$\frac{\partial Tl}{\partial Sw_{m-1}}$	0	0
$\frac{\partial Tl}{\partial Sw_m}$	$FF_{m-1/2}(\frac{bl}{\mu l})_{m-1/2} \frac{dkrl_m}{dSw_m}$	0
$\frac{\partial Tl}{\partial Sw_{m+1}}$	0	$FF_{m+1/2}(\frac{bl}{\mu l})_{m+1/2} \frac{dkrl_{m+1}}{dSw_{m+1}}$

Derivadas dos Termos de acumulação

Óleo

$$\frac{\partial ACO}{\partial p_{ijk}} = \frac{V p_{ijk}}{\Delta t} ([1 + cr(p - p^n)](1 - Sw) \frac{\partial bo}{\partial p} + (1 - Sw) bocr) \quad (A.44)$$

$$\frac{\partial ACO}{\partial Sw_{ijk}} = -\frac{V p_{ijk}}{\Delta t} ([1 + cr(p - p^n)]bo) \quad (A.45)$$

Água

$$\frac{\partial ACW}{\partial p_{ijk}} = \frac{V p_{ijk}}{\Delta t} ([1 + cr(p - p^n)]Sw \frac{\partial bw}{\partial p} + Sw bwc r) \quad (A.46)$$

$$\frac{\partial ACW}{\partial Sw_{ijk}} = -\frac{V p_{ijk}}{\Delta t} ([1 + cr(p - p^n)]bw) \quad (A.47)$$

Derivadas das Vazões

A vazão de cada bloco comunicante com o poço é dada por:

$$Ql = Tl_{1/2,j,k} POTlX1 \quad (A.48)$$

Óleo

$$\frac{\partial Q_o}{\partial p_{ijk}} = POTlX1 \frac{\partial T_{o1/2jk}}{\partial p_{ijk}} + T_{o1/2jk} \frac{POTlX1}{\partial p_{ijk}} \quad (A.49)$$

ou,

$$\begin{aligned} \frac{\partial Q_o}{\partial p_{ijk}} = & POTlX1 [w T_{o1/2jk} (\frac{1}{b_{o1/2jk}} \frac{db_{o1jk}}{p_{ijk}} - \frac{1}{\mu_{o1/2jk}} \frac{\mu_{o1jk}}{p_{ijk}})] + \\ & T_{o1/2jk} [1 - w \frac{\partial \gamma_{o1jk}}{\partial p_{ijk}} (Dc - D_{1jk})] \end{aligned} \quad (A.50)$$

$$\frac{\partial Q_o}{\partial S_{w_{ijk}}} = POTlX1 [w T_{o1/2jk} (\frac{\partial T_{o1/2jk}}{\partial S_{w_{ijk}}}) \quad (A.51)$$

Para a Água

$$Q_w = \frac{\lambda w}{\lambda_o} Q_o = \underbrace{\frac{k r w b w \mu_o}{k r o b o \mu w}}_{RMWO} Q_o \quad (A.52)$$

$$\frac{\partial Q_w}{\partial p_{ijk}} = RMWO \frac{\partial Q_o}{\partial p_{ijk}} + Q_o \frac{RMWO}{\partial p_{ijk}} \quad (A.53)$$

onde o termo $\frac{\partial Q_o}{\partial p_{ijk}}$, já foi calculado.

$$\frac{RMWO}{\partial p_{1jk}} = \frac{krw}{kro} \left[\frac{\mu o}{bo\mu w} \frac{\partial bw}{\partial p} + \frac{bw}{bo\mu w} \frac{\mu o}{\partial p} - \frac{bw\mu o}{bo^2\mu w} \frac{bo}{\partial p} - \frac{bw\mu o}{bo\mu w^2} \frac{\mu w}{\partial p} \right] \quad (A.54)$$

esta equação pode ser escrita de outra forma:

$$\frac{RMWO}{\partial p_{1jk}} = \frac{krwbw\mu o}{krobo\mu w} \underbrace{\left[\frac{1}{bw} \frac{\partial bw}{\partial p} + \frac{1}{\mu o} \frac{\mu o}{\partial p} - \frac{1}{bo} \frac{bo}{\partial p} - \frac{1}{\mu w} \frac{\mu w}{\partial p} \right]}_{\frac{\partial RMWO}{\partial p}} \quad (A.55)$$

portanto mais simplificadamente, teremos:

$$\frac{\partial Q_w}{\partial p_{1jk}} = RMWO \left[\frac{\partial Q_o}{\partial p_{1jk}} + Q_o \frac{RMWO}{\partial p_{1jk}} \right] \quad (A.56)$$

Em relação a saturação teremos:

$$\frac{\partial Q_w}{\partial Sw_{1jk}} = RMWO \frac{\partial Q_o}{\partial Sw_{1jk}} + Q_o \frac{RMWO}{\partial So_{1jk}} \quad (A.57)$$

$$\frac{\partial RMWO}{\partial Sw_{1jk}} = \frac{bw\mu o}{krobo\mu o} \left[\frac{\partial krw}{\partial Sw_{1jk}} - \frac{krw}{kro} \frac{\partial kro}{\partial Sw_{1jk}} \right] \quad (A.58)$$

12 APÊNDICE B - Dados Utilizados na Simulação

Pressão MPa	B_o m^3/m^3 std	Viscosidade cp	Densidade kg/m^3	R_{so} m^3 std/ m^3 std
0.10	1.062	1.040	714.4	0.2
1.82	1.150	0.975	698.2	16.1
3.55	1.207	0.910	678.0	32.1
7.00	1.295	0.830	657.4	66.1
13.89	1.435	0.695	625.2	113.3
17.34	1.500	0.641	614.1	138.0
20.79	1.565	0.594	605.8	165.6
27.68	1.695	0.510	594.0	226.2
34.57	1.827	0.449	584.0	288.2
62.15	2.357	0.203	552.9	531.4

Tabela B.1: Propriedades PVT do óleo saturado

Pressão MPa	B_o m^3/m^3 std	Viscosidade cp	Densidade kg/m^3
27.68	1.695	0.510	594.0
62.15	1.579	0.740	637.6

Tabela B.2: Propriedades PVT do óleo sub-saturado

Pressão MPa	B_w m^3/m^3 std	Viscosidade cp	Densidade kg/m^3	R_{sw} m^3 std/ m^3 std
0.10	1.0410	0.310	997.0	0.0
1.82	1.0403	0.310	997.7	0.0
3.55	1.0395	0.310	998.4	0.0
7.00	1.0380	0.310	999.8	0.0
13.89	1.0350	0.310	1002.7	0.0
17.34	1.0335	0.310	1004.2	0.0
20.79	1.0320	0.310	1005.7	0.0
27.68	1.0290	0.310	1008.6	0.0
34.57	1.0258	0.310	1011.7	0.0
62.15	1.0130	0.310	1024.5	0.0

Tabela B.3: Propriedades PVT da água

Pressão MPa	B_g $\text{m}^3/\text{m}^3 \text{ std}$	Viscosidade cp	Densidade kg/m^3
0.10	0.93576	0.0080	1.0
1.82	0.06790	0.0096	14.3
3.55	0.03523	0.0112	27.5
7.00	0.01795	0.0140	54.0
13.89	0.00906	0.0189	107.0
17.34	0.00727	0.0208	133.5
20.79	0.00606	0.0228	159.9
27.68	0.00455	0.0268	213.0
34.57	0.00364	0.0309	266.1
62.15	0.00217	0.0470	447.7

Tabela B.4: Propriedades PVT do gás

S_w	K_{rw}	K_{ro}
0.00	0.0	1.0
0.12	0.0	1.000
0.20	0.075	0.997
0.25	0.125	0.980
0.30	0.190	0.900
0.40	0.410	0.500
0.45	0.60	0.100
0.50	0.72	0.050
0.60	0.87	0.0001
0.70	0.94	0.0
0.85	0.98	0.0
1.00	1.00	0.0
1.10	1.00	0.0

Tabela B.5: Permeabilidades relativas óleo - água

$S_l = S_w$	P_{cow} (Pa)
0.0	50000.
0.12	50000.
0.2	20000.
0.3	10000.
0.35385	5000.
0.40769	2500.
0.46154	1000.
0.51538	500.
0.56923	200.
0.62308	100.
0.67692	50.
0.73077	25.
0.83846	15.
0.89231	10.
0.94615	5.
1.00000	0.0
1.10000	0.0

Tabela B.6: Curva de pressão capilar óleo - água

13 APÊNDICE C - Resultados

13.1 Poço com 300m de comprimento

PROGRAMA ACTIOW.FOR
SIMULADOR DE RESERVATORIO COM POÇO HORIZONTAL

TOTALMENTE IMPLICITO
TRIDIMENSIONAL
BI-FASICO (OLEO/AGUA)

PRINCIPAIS DADOS UTILIZADOS

Parametros de Simulacao:

Regiao do reservatorio - GRID CARTESIANO -

N# de blocos na direcao X	(I3) NXC==>	3
N# de blocos na direcao Y	(I3) NYC==>	3
N# de blocos na direcao Z	(I3) NZC==>	1
Dim. dos blocos direcao X	(F10.4) DBDX==>	100.00
Dim. dos blocos direcao Y	(F10.4) DBDY==>	50.00
Dim. dos blocos direcao Z	(F10.4) DBDZ==>	300.00
N. de blocos produtores	(I3) NBP==>	1
Coord. i do bloco produtor	(I3) IP==>	2
Coord. j do bloco produtor	(I3) JP==>	2
Coord. k do primeiro bloco produtor	(I3) KP==>	1

Regiao do poço - GRID RADIAL -

N# de blocos na direcao X	(I3) NX==>	3
N# de blocos na direcao Y	(I3) NY==>	4
N# de blocos na direcao Z	(I3) NZ==>	1
Time step inicial (dias)	(F8.3) DT==>	.001
Tolerancia para pressao (Pa)	(F8.3) TOLP==>	3447.400
Tolerancia para saturacao (fr)	(F6.4) TOLS==>	.0010
Max var de sat num DT (fracao)	(F6.4) DSLIM==>	.0500
Max var de pres. num DT (KPa)	(F8.3) DPLIM==>	689.480
Vazao de oleo maxima (m3/dia)	(F8.3) QOMAX==>	300.000
Vazao de oleo minima (m3/dia)	(F8.3) QOMIN==>	15.000
Tempo final de simulacao (dias)	(F8.2) TSIM==>	5000.00
Pwf minima no poço (MPa)	(F8.3) PWMIN==>	30.000
Raio do poço (m)	(F8.4) RW==>	.1000
Int. tempo p/ impr.relatorio (dias)	(F5.1) TIMPR==>	30.0
Int. tempo p/ impr.grafico (dias)	(F5.1) TIMPG==>	10.0
Int. celulas de impressao	(I3) IRIMP==>	1
Incremento de tempo maximo (dias)	(F5.1) DTMAX==>	10.0
Incremento de tempo minimo (dias)	(F6.4) DTMIN==>	.0010
N# max de iteracoes no JACOBIANO	(I3) ITMAX==>	5
N# min de iteracoes no JACOBIANO	(I3) ITMIN==>	2
Fatia J base para o calculo de Pwf	(I3) JPROD==>	1

Dados do teste de formacao:

Simulacao de teste de formacao	(S ou N) OPT==>	N
Vazao de producao (m3/dia)	(F8.3) QOTF==>	50.000
Tempo de fluxo (horas)	(F8.3) TPTF==>	24.000
Tempo de estatica (horas)	(F8.3) TETF==>	72.000
Incr. de tempo no teste (horas)	(F8.3) DTTF==>	.500

Dados do reservatorio:

Saturacao inicial de agua (fracao) (F6.4) SWIN==> .1200
 Densidade do gas (adim) (F6.4) DGAS==> .8000
 Compressibilidade da rocha (1/MPa) (E9.3) CR==> .440E-03
 Pressao inicial do reservatorio(MPa) (F8.4) PRIN==> 35.0000
 Temperatura do reservatorio (°C) (F8.4) TEMP==> 90.0000
 Raio externo da regio do poço (m) (F8.4) RE==> 39.894
 Cota do poço (m) (F8.3) ZC==> -2560.00
 Cota do contato o/w (m) (F8.3) ZCOW==> -2610.00
 Porosidade igual a .3000 para todos os blocos
 Permeabilidade na direcao X igual a 50.0000 para todos blocos
 Permeabilidade na direcao Y igual a 50.0000 para todos blocos
 Permeabilidade na direcao Z igual a 50.0000 para todos blocos

RESUMO DO TIME STEP 1			
Tempo	(dia)	!	.00100
Dt	(dia)	!	.00100
Qo	(m3/dia)	!	.3000D+03
Qg	(m3/dia)	!	.6786D+05
Qw	(m3/dia)	!	.0000D+00
RGD	(m3/m3)	!	226.2000
RWD	(m3/m3)	!	.0000
Pwf	(MPa)	!	34.6510
Np	(m3)	!	.3000D+00
Gp	(m3)	!	.6786D+02
Wp	(m3)	!	.0000D+00
Balanco de Materiais			
Oleo	(%)	!	.1000D+01
Gas	(%)	!	.1000D+01
Agua	(%)	!	.1000D+01
Recup.	(%)	!	.0000
IP	(m3/d/kg/cm2)!		552.5163

Mapa de Pressao e Saturacao (Pressao, So e Sw)

GRID CARTESIANO			
Camada 1			
	01	02	03
3	34.4088 .88000 .12000	34.4088 .88000 .12000	34.4088 .88000 .12000
2	34.7043 .88000 .12000	.0000 .88000 .00000	34.7043 .88000 .12000
1	35.0000 .00000 1.00000	35.0006 .00000 1.00000	35.0000 .00000 1.00000

GRID RADIAL			
Camada 1			
	01	02	03
1	34.6208 .87999 .12001	34.6861 .88000 .12000	34.7042 .88000 .12000
2	34.6180 .87999 .12001	34.6659 .88000 .12000	34.5557 .88000 .12000
3	34.6208 .87999 .12001	34.6861 .88000 .12000	34.7042 .88000 .12000
4	34.6235 .87999 .12001	34.7062 .88000 .12000	34.8518 .88000 .12000

Camada 1			
----------	--	--	--

BLOCO	PRESSAO	Qoj,k	Qwj,k	RWD
I=1 J	(MPa)	(m3/D)	(m3/D)	(m3/m3)
1	34.62077	75.00000	.00000	.00000

2	34.61804	75.00000	.00000	.00000
3	34.62077	75.00000	.00000	.00000
4	34.62349	75.00000	.00000	.00000

RESUMO DO TIME STEP 24

Tempo	(dia)	!	32.44348	!
Dt	(dia)	!	10.00000	!
Qo	(m3/dia)	!	.3000D+03	!
Qg	(m3/dia)	!	.6786D+05	!
Qw	(m3/dia)	!	.1891D+00	!
RGD	(m3/m3)	!	226.2000	!
RWD	(m3/m3)	!	.0006	!
Pwf	(MPa)	!	32.2753	!
Np	(m3)	!	.9733D+04	!
Gp	(m3)	!	.2202D+07	!
Wp	(m3)	!	.3794D+01	!

Balanco de Materiais

Oleo	(%)	!	.1000D+01	!
Gas	(%)	!	.1000D+01	!
Agua	(%)	!	.1000D+01	!
Recup.	(%)	!	.6949	!
IP	(m3/d/kg/cm2)	!	127.8716	!

Mapa de Pressao e Saturacao (Pressao, So e Sw)

GRID CARTESIANO

Camada 1

	01	02	03
3	32.2221 .87978 .12022	32.2062 .87977 .12023	32.2221 .87978 .12022
2	32.5054 .87976 .12024	.0000 .88000 .00000	32.5054 .87976 .12024
1	32.8419 .00000 1.00000	32.8288 .00000 1.00000	32.8419 .00000 1.00000

GRID RADIAL

Camada 1

	01	02	03
1	32.3272 .87976 .12024	32.3949 .87976 .12024	32.4634 .87976 .12024
2	32.3245 .87972 .12028	32.3750 .87974 .12026	32.3154 .87979 .12021
3	32.3272 .87976 .12024	32.3949 .87976 .12024	32.4634 .87976 .12024
4	32.3298 .87983 .12017	32.4144 .87982 .12018	32.6074 .85597 .14403

Camada 1

BLOCO	PRESSAO	Qoj,k	Qwj,k	RWO
I=1 J	(MPa)	(m3/D)	(m3/D)	(m3/m3)
1	32.32716	74.99997	.04908	.00065
2	32.32449	74.99988	.05614	.00075
3	32.32716	74.99997	.04908	.00065
4	32.32976	75.00017	.03481	.00046

RESUMO DO TIME STEP 27

Tempo	(dia)	62.44348
Dt	(dia)	10.00000
Qo	(m3/dia)	.3000D+03
Qg	(m3/dia)	.6786D+05
Qw	(m3/dia)	.3419D+00
RGD	(m3/m3)	226.2000
RWD	(m3/m3)	.0011
Pwf	(MPa)	30.2398
Np	(m3)	.1873D+05
Gp	(m3)	.4237D+07
Wp	(m3)	.1256D+02

Balanco de Materiais

Oleo	(%)	.1000D+01
Gas	(%)	.1000D+01
Agua	(%)	.1000D+01
Recup.	(%)	1.3374
IP	(m3/d/kg/cm2)	130.5906

Mapa de Pressao e Saturacao (Pressao, So e Sw)

GRID CARTESIANO

Camada 1

	01	02	03
3	30.1828 .87960 .12040	30.1672 .87956 .12044	30.1828 .87960 .12040
2	30.4651 .87953 .12047	.0000 .88000 .00000	30.4651 .87953 .12047
1	30.8104 .00000 1.00000	30.7973 .00000 1.00000	30.8104 .00000 1.00000

GRID RADIAL

Camada 1

	01	02	03
1	30.2906 .87954 .12046	30.3569 .87955 .12045	30.4240 .87955 .12045
2	30.2879 .87951 .12049	30.3370 .87955 .12045	30.2766 .87964 .12036
3	30.2906 .87954 .12046	30.3569 .87955 .12045	30.4240 .87955 .12045
4	30.2931 .87966 .12034	30.3763 .87964 .12036	30.5674 .83326 .16674

Camada 1

BLOCO	PRESSAO	Qoj,k	Qwj,k	RWO
I=1 J	(MPa)	(m3/D)	(m3/D)	(m3/m3)
1	30.29055	74.99994	.08952	.00119
2	30.28789	74.99984	.09632	.00128
3	30.29055	74.99994	.08952	.00119
4	30.29315	75.00027	.06656	.00089

RESUMO DO TIME STEP 34

Tempo	(dia)	84.42590
Dt	(dia)	7.11914
Qo	(m3/dia)	.7956D+01
Qg	(m3/dia)	.1800D+04
Qw	(m3/dia)	.1249D-01
RGD	(m3/m3)	226.2000
RWD	(m3/m3)	.0016
Pwf	(MPa)	30.0060
Np	(m3)	.2070D+05
Gp	(m3)	.4682D+07
Wp	(m3)	.1496D+02

Balanço de Materiais

Oleo	(%)	.1000D+01
Gas	(%)	.1000D+01
Agua	(%)	.1000D+01
Recup.	(%)	1.4777
IP	(m3/d/kg/cm2)	132.0135

Mapa de Pressao e Saturacao (Pressao, So e Sw)

GRID CARTESIANO

Camada 1

	01	02	03
3	29.7195 .87959 .12041	29.7191 .87952 .12048	29.7195 .87959 .12041
2	30.0119 .87947 .12053	.0000 .88000 .00000	30.0119 .87947 .12053
1	30.3747 .00000 1.00000	30.3742 .00000 1.00000	30.3747 .00000 1.00000

GRID RADIAL

Camada 1

	01	02	03
1	30.0073 .87933 .12067	30.0091 .87950 .12050	30.0108 .87951 .12049
2	30.0046 .87914 .12086	29.9891 .87948 .12052	29.8637 .87966 .12034
3	30.0073 .87933 .12067	30.0091 .87950 .12050	30.0108 .87951 .12049

4	30.0100	30.0291	30.1579
	.87964	.87961	.82825
	.12036	.12039	.17175

Camada 1

BLOCO		PRESSAO	Qoj,k	Qwj,k	RWO
I=1	J	(MPa)	(m3/D)	(m3/D)	(m3/m3)
	1	30.00733	1.98917	.00325	.00163
	2	30.00462	1.98681	.00426	.00215
	3	30.00733	1.98917	.00325	.00163
	4	30.01005	1.99079	.00173	.00087

Tempo de execucao (seg) =27

13.2 Poço com 100m de comprimento

PROGRAMA ACTIOW.FOR
SIMULADOR DE RESERVATORIO COM POÇO HORIZONTAL
 TOTALMENTE IMPLICITO
 TRIDIMENSIONAL
 BI-FASICO (OLEO/AGUA)

PRINCIPAIS DADOS UTILIZADOS

Parametros de Simulacao:

Regiao do reservatorio - GRID CARTESIANO -

N# de blocos na direcao X	(I3) NXC==>	3
N# de blocos na direcao Y	(I3) NYC==>	3
N# de blocos na direcao Z	(I3) NZC==>	3
Dim. dos blocos direcao X	(F10.4) DBDX==>	100.00
Dim. dos blocos direcao Y	(F10.4) DBDY==>	50.00
Dim. dos blocos direcao Z	(F10.4) DBDZ==>	100.00
N. de blocos produtores	(I3) NBP==>	1
Coord. i do bloco produtor	(I3) IP==>	2
Coord. j do bloco produtor	(I3) JP==>	2
Coord. k do primeiro bloco produtor	(I3) KP==>	1

Regiao do poço - GRID RADIAL -

N# de blocos na direcao X	(I3) NX==>	3
N# de blocos na direcao Y	(I3) NY==>	4
N# de blocos na direcao Z	(I3) NZ==>	1
Time step inicial (dias)	(F8.3) DT==>	.001
Tolerancia para pressao (Pa)	(F8.3) TOLP==>	3447.400
Tolerancia para saturacao (fr)	(F6.4) TOLS==>	.0010
Max var de sat num DT (fracao)	(F6.4) DSLIM==>	.0500
Max var de pres. num DT (KPa)	(F8.3) DPLIM==>	689.480
Vazao de oleo maxima (m3/dia)	(F8.3) QOMAX==>	300.000
Vazao de oleo minima (m3/dia)	(F8.3) QOMIN==>	15.000
Tempo final de simulacao (dias)	(F8.2) TSIM==>	5000.00
Pwf minima no poço (MPa)	(F8.3) PWMIN==>	30.000
Raio do poço (m)	(F8.4) RW==>	.1000
Int. tempo p/ impr.relatorio (dias)	(F5.1) TIMPR==>	30.0
Int. tempo p/ impr.grafico (dias)	(F5.1) TIMPG==>	10.0
Int. celulas de impressao	(I3) IRIMP==>	1
Incremento de tempo maximo (dias)	(F5.1) DTMAX==>	10.0
Incremento de tempo minimo (dias)	(F6.4) DTMIN==>	.0010
N# max de iteracoes no JACOBIANO	(I3) ITMAX==>	5
N# min de iteracoes no JACOBIANO	(I3) ITMIN==>	2
Fatia J base para o calculo de Pwf	(I3) JPROD==>	1

Dados do teste de formacao:

Simulacao de teste de formacao	(S ou N) OPT==>	N
Vazao de producao (m3/dia)	(F8.3) QOTF==>	50.000
Tempo de fluxo (horas)	(F8.3) TPTF==>	24.000
Tempo de estatica (horas)	(F8.3) TETF==>	72.000
Incr. de tempo no teste (horas)	(F8.3) DTTF==>	.500

Dados do reservatorio:

Saturacao inicial de agua (fracao) (F6.4) SWIN==> .1200
 Densidade do gas (adim) (F6.4) DGAS==> .8000
 Compressibilidade da rocha (1/MPa) (E9.3) CR==> .440E-03
 Pressao inicial do reservatorio(MPa) (F8.4) PRIN==> 35.0000
 Temperatura do reservatorio (°C) (F8.4) TEMP==> 90.0000
 Raio externo da regio do poço (m) (F8.4) RE==> 39.894
 Cota do poço (m) (F8.3) ZC==> -2560.00
 Cota do contato o/w (m) (F8.3) ZCOW==> -2610.00
 Porosidade igual a .3000 para todos os blocos
 Permeabilidade na direcao X igual a 50.0000 para todos blocos
 Permeabilidade na direcao Y igual a 50.0000 para todos blocos
 Permeabilidade na direcao Z igual a 50.0000 para todos blocos

RESUMO DO TIME STEP 1			
Tempo	(dia)	!	.00100
Dt	(dia)	!	.00100
Qo	(m3/dia)	!	.3000D+03
Qg	(m3/dia)	!	.6786D+05
Qw	(m3/dia)	!	.0000D+00
RGO	(m3/m3)	!	226.2000
RWO	(m3/m3)	!	.0000
Pwf	(MPa)	!	34.5445
Np	(m3)	!	.3000D+00
Gp	(m3)	!	.6786D+02
Wp	(m3)	!	.0000D+00
Balanco de Materiais			
Oleo	(%)	!	.1000D+01
Gas	(%)	!	.1000D+01
Agua	(%)	!	.1000D+01
Recup.	(%)	!	.0000
IP	(m3/d/kg/cm2)	!	184.1721

Mapa de Pressao e Saturacao (Pressao, So e Sw)

GRID CARTESIANO

Camada 1

	01	02	03
3	34.4088 .88000 .12000	34.4088 .88000 .12000	34.4088 .88000 .12000
2	34.7043 .88000 .12000	.0000 .88000 .00000	34.7043 .88000 .12000
1	35.0000 .00000 1.00000	35.0006 .00000 1.00000	35.0000 .00000 1.00000

Camada 2

	01	02	03
3	34.4088 .88000 .12000	34.4088 .88000 .12000	34.4088 .88000 .12000
2	34.7043 .88000 .12000	34.7040 .88000 .12000	34.7043 .88000 .12000
1	35.0000 .00000 1.00000	35.0000 .00000 1.00000	35.0000 .00000 1.00000

Camada 3

	01	02	03
3	34.4088 .88000 .12000	34.4088 .88000 .12000	34.4088 .88000 .12000
2	34.7043 .88000 .12000	34.7043 .88000 .12000	34.7043 .88000 .12000

1	35.0000	35.0000	35.0000
	.00000	.00000	.00000
	1.00000	1.00000	1.00000

GRID RADIAL
Camada 1

	01	02	03
1	34.4538 .87997 .12003	34.6498 .87999 .12001	34.7039 .88000 .12000
2	34.4511 .87997 .12003	34.6296 .87999 .12001	34.5554 .88000 .12000
3	34.4538 .87997 .12003	34.6498 .87999 .12001	34.7039 .88000 .12000
4	34.4565 .87997 .12003	34.6698 .87999 .12001	34.8515 .88000 .12000

Camada 1

BLOCO	PRESSAO	Qo,j,k	Qw,j,k	RWD
I=1 J	(MPa)	(m3/D)	(m3/D)	(m3/m3)
1	34.45381	75.00000	.00000	.00000
2	34.45108	75.00000	.00000	.00000
3	34.45381	75.00000	.00000	.00000
4	34.45653	75.00000	.00000	.00000

RESUMO DO TIME STEP168

Tempo	(dia)	30.05590
Dt	(dia)	.19462
Qo	(m3/dia)	.3000D+03
Qg	(m3/dia)	.6786D+05
Qw	(m3/dia)	.1017D+02
RGD	(m3/m3)	226.2000
RWD	(m3/m3)	.0339
Pwf	(MPa)	31.9132
Np	(m3)	.9017D+04
Gp	(m3)	.2040D+07
Wp	(m3)	.1401D+03

Balanco de Materiais

Oleo	(%)	.1000D+01
Gas	(%)	.1000D+01
Agua	(%)	.1000D+01
Recup.	(%)	.6437
IP	(m3/d/kg/cm2)	43.9504

Mapa de Pressao e Saturacao (Pressao, So e Sw)

GRID CARTESIANO
Camada 1

	01	02	03
3	32.3138 .87979 .12021	32.2763 .87978 .12022	32.3138 .87979 .12021

2	32.5826 .87977 .12023	.0000 .88000 .00000	32.5826 .87977 .12023
---	-----------------------------	---------------------------	-----------------------------

1	32.9601 .00000 1.00000	32.9172 .00000 1.00000	32.9601 .00000 1.00000
---	------------------------------	------------------------------	------------------------------

Camada 2

	01	02	03
--	----	----	----

3	32.3923 .87980 .12020	32.3800 .87979 .12021	32.3923 .87980 .12020
---	-----------------------------	-----------------------------	-----------------------------

2	32.6833 .87978 .12022	32.6657 .87978 .12022	32.6833 .87978 .12022
---	-----------------------------	-----------------------------	-----------------------------

1	32.9898 .00002 .99998	32.9807 .00000 1.00000	32.9898 .00002 .99998
---	-----------------------------	------------------------------	-----------------------------

Camada 3

	01	02	03
--	----	----	----

3	32.4363 .87980 .12020	32.4331 .87980 .12020	32.4363 .87980 .12020
---	-----------------------------	-----------------------------	-----------------------------

2	32.7274 .87979 .12021	32.7239 .87979 .12021	32.7274 .87979 .12021
---	-----------------------------	-----------------------------	-----------------------------

1	33.0153 .00229 .99771	33.0126 .00206 .99794	33.0153 .00229 .99771
---	-----------------------------	-----------------------------	-----------------------------

GRID RADIAL

Camada 1

	01	02	03
--	----	----	----

1	32.0721 .87494 .12506	32.2745 .87856 .12144	32.4760 .87976 .12024
---	-----------------------------	-----------------------------	-----------------------------

2	32.0693 .87975 .12025	32.2540 .87976 .12024	32.3256 .87979 .12021
---	-----------------------------	-----------------------------	-----------------------------

3	32.0721 .87494 .12506	32.2745 .87856 .12144	32.4760 .87976 .12024
---	-----------------------------	-----------------------------	-----------------------------

4	32.0748 .83972 .16028	32.2947 .83030 .16970	32.6238 .79096 .20904
---	-----------------------------	-----------------------------	-----------------------------

Camada 1

BLOCO	PRESSAO	Qoj,k	Qwj,k	RWO
I=1 J	(MPa)	(m3/D)	(m3/D)	(m3/m3)
1	32.07208	75.02146	1.01680	.01355
2	32.06926	75.03504	.05114	.00068
3	32.07208	75.02146	1.01680	.01355
4	32.07484	74.92205	8.08654	.10793

RESUMO DO TIME STEP316

Tempo	(dia)	!	60.12462	!
Dt	(dia)	!	.43789	!

Qo	(m3/dia)	!	.2320D+03	!
Qg	(m3/dia)	!	.5248D+05	!
Qw	(m3/dia)	!	.1520D+02	!
RGD	(m3/m3)	!	226.2000	!
RWD	(m3/m3)	!	.0655	!
Pwf	(MPa)	!	30.0170	!
Np	(m3)	!	.1793D+05	!
Gp	(m3)	!	.4055D+07	!
Wp	(m3)	!	.5932D+03	!

Balanco de Materiais

Oleo	(%)	!	.1000D+01	!
Gas	(%)	!	.1000D+01	!
Agua	(%)	!	.1000D+01	!
Recup.	(%)	!	1.2800	!
IP	(m3/d/kg/cm2)!		44.5924	

Mapa de Pressao e Saturacao (Pressao, So e Sw)

GRID CARTESIANO

Camada 1

	01	02	03
3	30.2545 .87960 .12040	30.2246 .87957 .12043	30.2545 .87960 .12040
2	30.5272 .87953 .12047	.0000 .88000 .00000	30.5272 .87953 .12047
1	30.9030 .00000 1.00000	30.8712 .00000 1.00000	30.9030 .00000 1.00000

Camada 2

	01	02	03
3	30.3218 .87961 .12039	30.3113 .87961 .12039	30.3218 .87961 .12039
2	30.6122 .87955 .12045	30.5969 .87955 .12045	30.6122 .87955 .12045
1	30.9247 .00002 .99998	30.9179 .00000 1.00000	30.9247 .00002 .99998

Camada 3

	01	02	03
3	30.3619 .87961 .12039	30.3591 .87961 .12039	30.3619 .87961 .12039
2	30.6531 .87956 .12044	30.6501 .87956 .12044	30.6531 .87956 .12044
1	30.9426 .00427 .99573	30.9404 .00380 .99620	30.9426 .00427 .99573

GRID RADIAL

Camada 1

	01	02	03
1	30.1338 .86201 .13799	30.2868 .87088 .12912	30.4426 .87954 .12046
2	30.1313 .87803 .12197	30.2672 .87957 .12043	30.2940 .87964 .12036
3	30.1338 .86201 .13799	30.2868 .87088 .12912	30.4426 .87954 .12046
4	30.1361 .81679 .18321	30.3043 .79437 .20563	30.5876 .71263 .28737

Camada 1

BLOCO	PRESSAO	Qoj,k	Qwj,k	RWO
I=1 J	(MPa)	(m3/D)	(m3/D)	(m3/m3)
1	30.13384	58.05285	2.69940	.04650
2	30.13128	58.16289	.29044	.00499
3	30.13384	58.05285	2.69940	.04650
4	30.13610	57.72554	9.50680	.16469

RESUMO DO TIME STEP351

Tempo	(dia)	89.70982
Dt	(dia)	2.21684
Qo	(m3/dia)	.1480D+02
Qg	(m3/dia)	.3349D+04
Qw	(m3/dia)	.1005D+00
RGO	(m3/m3)	226.2000
RWO	(m3/m3)	.0068
Pwf	(MPa)	30.0170
Np	(m3)	.2019D+05
Gp	(m3)	.4568D+07
Wp	(m3)	.6893D+03

Balanço de Materiais

Oleo	(%)	.1000D+01
Gas	(%)	.1000D+01
Agua	(%)	.1000D+01
Recup.	(%)	1.4418
IP	(m3/d/kg/cm2)	39.1183

Mapa de Pressao e Saturacao (Pressao, So e Sw)

GRID CARTESIANO

Camada 1

	01	02	03
3	29.7628 .87959 .12041	29.7608 .87953 .12047	29.7628 .87959 .12041
2	30.0541 .87946 .12054	.0000 .88000 .00000	30.0541 .87946 .12054
1	30.4361 .00000	30.4344 .00000	30.4361 .00000

	1.00000	1.00000	1.00000
Camada 2			
	01	02	03
3	29.7679 .87959 .12041	29.7674 .87959 .12041	29.7679 .87959 .12041
2	30.0605 .87948 .12052	30.0599 .87947 .12053	30.0605 .87948 .12052
1	30.4371 .00002 .99998	30.4367 .00000 1.00000	30.4371 .00002 .99998

Camada 3			
	01	02	03
3	29.7711 .87959 .12041	29.7709 .87959 .12041	29.7711 .87959 .12041
2	30.0638 .87948 .12052	30.0637 .87948 .12052	30.0638 .87948 .12052
1	30.4377 .00436 .99564	30.4376 .00386 .99614	30.4377 .00436 .99564

GRID RADIAL
Camada 1

	01	02	03
1	30.0258 .87748 .12252	30.0370 .87730 .12270	30.0482 .87949 .12051
2	30.0232 .87926 .12074	30.0171 .87950 .12050	29.9010 .87968 .12032
3	30.0258 .87748 .12252	30.0370 .87730 .12270	30.0482 .87949 .12051
4	30.0285 .87550 .12450	30.0566 .86638 .13362	30.1952 .69598 .30402

Camada 1

BLOCO	PRESSAO	Qoj,k	Qwj,k	RWO
I=1 J	(MPa)	(m3/D)	(m3/D)	(m3/m3)
1	30.02585	3.70475	.02511	.00678
2	30.02316	3.71582	.00738	.00199
3	30.02585	3.70475	.02511	.00678
4	30.02851	3.67845	.04289	.01166

Tempo de execucao (seg) =2641

14 APÊNDICE D - Programas Computacionais

Neste anexo é apresentada a listagem do programa computacional, em linguagem FORTRAN, para o modelo totalmente acoplado. Este programa e os demais desacoplados, assim como os arquivos de exemplos de dados de entrada e de saídas, encontram-se arquivados no Departamento de Engenharia de Petróleo da Faculdade de Engenharia Mecânica da UNICAMP.

No total são três programas em linguagem FORTRAN:

1. Modelo totalmente acoplado
Programa ACTIOW.FOR
2. Modelo desacoplado com acoplamento na iteração
Programa ACOOWI.FOR
3. Modelo desacoplado com acoplamento no *time-step*
Programa ACOOWTS.FOR

UNIVERSIDADE ESTADUAL DE CAMPINAS - UNICAMP
MESTRADO EM ENGENHARIA DE PETROLEO - TURMA 1989

SIMULADOR DE RESERVATORIO COM POCO HORIZONTAL
=====

AUTOR: JOSE ADILSON TENORIO GOMES
=====

PROGRAMA --- ACTIOW.FOR (totalmente acoplado)

CARACTERISTICAS DO PROGRAMA

- * Linguagem FORTRAN;
- * Simula o comportamento de um reservatorio retangular produzindo
atraves de um poco horizontal, usando refinamento local, ou seja,
a regio do poco e representada por uma malha cilindrica e a regio
do reservatorio por uma malha cartesiana;
- * Regioes totalmente acoplados - resolvidas simultaneamente no mesmo
Jacobiano;
- * Tridimensional;
- * Bifasico - oleo/agua;
- * Totalmente implicito;
- * Fluidos inicialmente em equilibrio gravitacional;
- * A matriz jacobiana pode ser resolvida das seguintes formas:
 - metodo direto usando eliminacao de Gauss
 - metodo direto usando algoritmo otimizado (NSPIV)
 - metodo iterativo BSQR

Arquivos de entrada e saida :

Entrada

- 1 : (PVTW.DAT) - dados de analises PVT
- 2 : (PETROW.DAT) - dados de analises petrofisicas
- 3 : (ACTIOW.DAT) - dados gerais de reservatorio e de simulacao

SAIDA

- 4 : (ACTIOW.REL) - Relatorio geral, com resultados da producao,
balanco de materiais e mapas de pressao e
de saturacao
- 5 : (ACTIO1.DAT) - saida para grafico - (tempo, Qo, Qw, Qg, Pwf)
- 7 : (ACTIO2.DAT) - saida para grafico - (tempo, Np, Wp)
- 8 : (ACTIO3.DAT) - saida para grafico - (tempo, RW0, RG0)
- 9 : (ACTIO4.DAT) - saida para grafico - (perfil vert de saturacao)
- 11: (PHTEST.REL) - saida para relatorio de teste de formacao
- 12: (PHTEST.DAT) - saida para grafico de teste de formacao

\$system

IMPLICIT DOUBLE PRECISION(A-H,O-Z)

PARAMETER(NLAB=18,NXM=08,NYM=05,NZM=04,PI=3.141592654D0)

C Dados PVT e petrofisica

```

COMMON/LPVTG/PGL,BGL,VGL,DGL,NPG
COMMON/LPETR/SWL,KRWL,KROL,SLL,PCOWL,NKR,NPC
C Parametros de simulacao e de producao
COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*      NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
COMMON/SIM/DT,DTMAX,DTMIN,TOLP,TOLS,DSLIM,DPLIM,TSIM,
*      QOMAX,QOMIN,PWMIN,BSOR,JPROD,MSOLU,ITMAX,ITMIN
C Dados do reservatorio
COMMON/RES/DGAS,CR,PRIN,TEMP,ZC,RW,RE,SWIN,PHI,PERX,PERY,
*      PERZ,ZCOW,INDPOR,INDPER
C Dados de impressao
COMMON/IMP/TIMPR,TIMPG,IRIMP
C Dados do teste de formacao
COMMON/TFR/QOTF,TPTF,TETF,DTTF,DTI,TFLUX,BETA,OPT
C Variaveis da malha
COMMON/MALHA/ZB,RB,VPOR,DRO,WX,WY,WZ,FFX,FFY,FFZ,DTETA
C Variaveis do passo de tempo n
COMMON/TSA/POA,SOA,SGA,SWA
COMMON/PVTA/BOA,RSDA,BWA
C Variaveis do passo de tempo n+1
COMMON/TS/PO,SO,SG,SW
COMMON/PVT/BO,DBO,VO,DVO,DO,DDO,RSO,DRSO,BW,DBW,VW,DVW,DW,DDW,CW
C Variaveis das interfaces X e Y
COMMON/IFACEX/BOMX,VOMX,DOMX,BWMX,VWMX,DWMX
COMMON/IFACEY/BOMY,VOMY,DOMY,BWMY,VWMY,DWMY
COMMON/IFACEZ/BOMZ,VOMZ,DOMZ,BWMZ,VWMZ,DWMZ
C Valores de perm. relativas ,pressoes capilares e suas derivadas
COMMON/PETR/KRO,KRW,DKRO,DKRW,PC,DPC
C Valores de transmissibilidades e suas derivadas (faces X e Z)
COMMON/DTRANS/TOX,TWX,TOY,TWY,TOZ,TWZ,
*      TOX1SA,TOX1SI,TWX1SA,TWX1SI,
*      TOX2SI,TOX2SP,TWX2SI,TWX2SP,
*      TOY1SA,TOY1SI,TWY1SA,TWY1SI,
*      TOY2SI,TOY2SP,TWY2SI,TWY2SP,
*      TOZ1SA,TOZ1SI,TWZ1SA,TWZ1SI,
*      TOZ2SI,TOZ2SP,TWZ2SI,TWZ2SP
C Vazoes e suas derivadas, pressao de fluxo tempo
COMMON/VAZAO/QO,QOT,QG,QGT,QW,QWT,DQOP,DQOS,
*      DQWP,DQWS,RWO,RGO,PWF,TEMPO
C Volumes acumulados
COMMON/ACUM/QOAC,QGAC,QWAC,VOORR,VWORR,VOORC,VWORC,BNPR,BWPR,
*      BNPC,BWPC,PWFA
C Matriz jacobiana e vetores solucao e independente
COMMON/SEQS/XR,XC
C
DIMENSION POL(NLAB),BOL(NLAB),VOL(NLAB),DOL(NLAB),RSOL(NLAB),
*      POLS(NLAB),BOLS(NLAB),VOLS(NLAB),DOLS(NLAB)
DIMENSION PWL(NLAB),BWL(NLAB),VWL(NLAB),DWL(NLAB),RSWL(NLAB)
DIMENSION PGL(NLAB),BGL(NLAB),VGL(NLAB),DGL(NLAB)
DIMENSION SWL(NLAB),SLL(NLAB),PCOWL(NLAB)
REAL*8      KRWL(NLAB),KROL(NLAB)
C
DIMENSION PHI(NXM,NYM,NZM,2),PERX(NXM,NYM,NZM,2),
*      PERY(NXM,NYM,NZM,2),PERZ(NXM,NYM,NZM,2)
DIMENSION ZB(0:NXM+1,NYM,2),RB(NXM+1),VPOR(NXM,NYM,NZM,2),
*      FFX(NXM+1,NYM,NZM,2),FFY(NXM+1,NYM,NZM,2),
*      FFZ(NXM+1,NYM,NZM+1,2),DTETA(NYM+1)

```

```

DIMENSION POA(NXM,NYM,NZM,2), SOA(NXM,NYM,NZM,2),
*   SGA(NXM,NYM,NZM,2), SWA(NXM,NYM,NZM,2)
DIMENSION BOA(NXM,NYM,NZM,2), BWA(NXM,NYM,NZM,2)
DIMENSION PO(0:NXM+1,0:NYM+1,0:NZM+1,2), SO(NXM,NYM,NZM,2),
*   SG(NXM,NYM,NZM,2), SW(NXM,NYM,NZM,2)
DIMENSION BO(NXM,NYM,NZM,2), DBO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   VO(NXM,NYM,NZM,2), DVO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   DO(NXM,NYM,NZM,2), DDO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   BW(NXM,NYM,NZM,2), DBW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   VW(NXM,NYM,NZM,2), DVW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   DW(NXM,NYM,NZM,2), DDW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   CW(NXM,NYM,NZM,2)
DIMENSION BOMX(0:NXM+1,NYM,NZM,2), VOMX(0:NXM+1,NYM,NZM,2),
*   DOMX(0:NXM+1,NYM,NZM,2), BWMX(0:NXM+1,NYM,NZM,2),
*   VWMX(0:NXM+1,NYM,NZM,2), DWMX(0:NXM+1,NYM,NZM,2)
DIMENSION BOMY(NXM,NYM+1,NZM,2), VOMY(NXM,NYM+1,NZM,2),
*   DOMY(NXM,NYM+1,NZM,2), BWMY(NXM,NYM+1,NZM,2),
*   VWMY(NXM,NYM+1,NZM,2), DWMY(NXM,NYM+1,NZM,2)
DIMENSION BOMZ(NXM,NYM,NZM+1,2), VOMZ(NXM,NYM,NZM+1,2),
*   DOMZ(NXM,NYM,NZM+1,2), BWMZ(NXM,NYM,NZM+1,2),
*   VWMZ(NXM,NYM,NZM+1,2), DWMZ(NXM,NYM,NZM+1,2)
DIMENSION DKRO(NXM,NYM,NZM,2), DKRW(NXM,NYM,NZM,2),
*   PC(0:NXM+1,0:NYM+1,0:NZM+1,2), DPC(0:NXM+1,0:NYM+1,0:NZM+1,2)
REAL*8   KRO(NXM,NYM,NZM,2), KRW(NXM,NYM,NZM,2)
DIMENSION TOX(NXM+1,NYM,NZM,2), TWX(NXM+1,NYM,NZM,2),
*   TOX1SA(NXM,NYM,NZM,2), TOX1SI(NXM,NYM,NZM,2),
*   TWX1SA(NXM,NYM,NZM,2), TWX1SI(NXM,NYM,NZM,2),
*   TOX2SI(NXM,NYM,NZM,2), TOX2SP(NXM,NYM,NZM,2),
*   TWX2SI(NXM,NYM,NZM,2), TWX2SP(NXM,NYM,NZM,2)
DIMENSION TOY(NXM,NYM+1,NZM,2), TWY(NXM,NYM+1,NZM,2),
*   TOY1SA(NXM,NYM,NZM,2), TOY1SI(NXM,NYM,NZM,2),
*   TWY1SA(NXM,NYM,NZM,2), TWY1SI(NXM,NYM,NZM,2),
*   TOY2SI(NXM,NYM,NZM,2), TOY2SP(NXM,NYM,NZM,2),
*   TWY2SI(NXM,NYM,NZM,2), TWY2SP(NXM,NYM,NZM,2)
DIMENSION TOZ(NXM,NYM,NZM+1,2), TWZ(NXM,NYM,NZM+1,2),
*   TOZ1SA(NXM,NYM,NZM,2), TOZ1SI(NXM,NYM,NZM,2),
*   TWZ1SA(NXM,NYM,NZM,2), TWZ1SI(NXM,NYM,NZM,2),
*   TOZ2SI(NXM,NYM,NZM,2), TOZ2SP(NXM,NYM,NZM,2),
*   TWZ2SI(NXM,NYM,NZM,2), TWZ2SP(NXM,NYM,NZM,2)
DIMENSION XR(NXM*NYM*NZM*2), XC(NXM*NYM*NZM*2)
DIMENSION QO(NYM,NZM), QG(NYM,NZM), QW(NYM,NZM), DQOP(NYM,NZM),
*   DQOS(NYM,NZM), DQWP(NYM,NZM), DQWS(NYM,NZM)
integer*4 _STOPW,in
CHARACTER*1 OPT
LOGICAL TESTE,VPWF,CONV,VDS

```

C-----
C abertura dos arquivos

C-----
OPEN(UNIT=1,FILE='PVTW.DAT',STATUS='OLD')
OPEN(UNIT=2,FILE='PETROW.DAT',STATUS='OLD')
OPEN(UNIT=3,FILE='ACTIOW.DAT',STATUS='OLD')
OPEN(UNIT=4,FILE='ACTIOW.REL',STATUS='UNKNOWN')
OPEN(UNIT=5,FILE='ACTIO1.DAT',STATUS='UNKNOWN')
OPEN(UNIT=7,FILE='ACTIO2.DAT',STATUS='UNKNOWN')
OPEN(UNIT=8,FILE='ACTIO3.DAT',STATUS='UNKNOWN')
OPEN(UNIT=9,FILE='ACTIO4.DAT',STATUS='UNKNOWN')
OPEN(UNIT=11,FILE='PHTEST.REL',STATUS='UNKNOWN')

```
OPEN(UNIT=12,FILE='PHTEST.DAT',STATUS='UNKNOWN')
```

```
PROGRAMA PRINCIPAL
```

```
Leitura dos dados
```

```
in= STOPW(0)
CALL ENTRA
IF(MSOLU.EQ.3) THEN
  write(*,*)'Entre com bsor (1 a 2)'
  read(*,*)BSOR
ENDIF
CALL IMPDE
```

```
Conversao das unidades lidas para o sistema S.I.
```

```
CALL CONVUNI
```

```
Inicializacao de variaveis
```

```
CALL INICIA
```

```
Lancamento da malha e calculo das cotas dos pontos, fatores de forma e ponderadores
```

```
CALL GRID
```

```
Calculo das pressoes e saturacoes de agua iniciais
```

```
CALL PRESSO (PWF)
```

```
Inicializacao das saturacoes das fases nos blocos
```

```
Calculo do volume porose inicial dos blocos
```

```

NX=NXR
NY=NYR
NZ=NZR
DO 5 L=1,2
  IF(L.EQ.2) THEN
    NX=NXC
    NY=NYC
    NZ=NZC
  ENDIF
DO 5 I=1,NX
  DO 5 J=1,NY
    DO 5 K=1,NZ
      IF(L.EQ.2) THEN
        AUX=PHI(I,J,K,L)*(1.000-CR*POA(I,J,K,L))
        VPOR(I,J,K,L)=AUX*DBDX*DBDY*DBDZ
        IF(I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.K.LT.KP+NBP)
          * VPOR(I,J,K,L)=0.000
        ELSE
          AUX1=RW*RW*DBDZ/2.000
          AUX=PHI(I,J,K,L)*(1.000-CR*POA(I,J,K,L))
          VPOR(I,J,K,L)=(DEXP(2.000*I*DRO))-(DEXP(2.000*(I-1)*DRO))
          VPOR(I,J,K,L)=VPOR(I,J,K,L)*AUX*AUX1*DTETA(J)
        ENDIF
        SOA(I,J,K,L)=1.000-SWA(I,J,K,L)

        SGA(I,J,K,L)=0.000
        PO(I,J,K,L)=POA(I,J,K,L)
        SO(I,J,K,L)=SOA(I,J,K,L)
        SG(I,J,K,L)=SGA(I,J,K,L)
        SW(I,J,K,L)=SWA(I,J,K,L)
      
```

```

5      CONTINUE
C-----
C Calculo dos volumes originais de oleo e agua
C-----
C      CALL RESERV
C-----
C Impressao dos cabecalhos para os arquivos do graficos
C-----
C      WRITE(5,10)
C      WRITE(7,15)
C      WRITE(8,16)
10     FORMAT('TEMPO(dias)  QO(m3/dia)  QW(m3/dia)  Qg(m3/dia)  Pwf(
*MPa)')
15     FORMAT('TEMPO(dias)  NP(m3)  WP(m3)  GP(m3)')
16     FORMAT('TEMPO(dias)  RWD(m3/m3)  RGD(m3/m3)')
C-----
C Inic. de producoes de oleo, gas e agua acumuladas, do tempo e de pwf
C-----
C      QQAC=0.000
C      QGAC=0.000
C      QWAC=0.000
C      QOT=QOMAX
C      PWFA=PUF
C      TEMPO=0.000
C-----
C Inicializacao de variaveis de controle de impressao, contador
C de passos de tempo e de outras var. de controle do processo iterativo
C-----
C      ITR=0
C      ITG=0
C      NTS=0
C iniciar com vazao constante ----> vpwf=.true.
C      VPWF=.TRUE.
C-----
C Opcao para teste de formacao
C-----
C      IF((OPT.EQ.'S').AND.(TETP.NE.0.0).AND.(DTTF.NE.0.0)) THEN
C          OPT='I'
C          TESTE=.TRUE.
C          DTI=DT
C          TEMPTF=0.000
C          CALL CTESTE(TESTE,NTS,DT,QOMAX,TEMPTF,NITER)
C      ELSE
C          TESTE=.FALSE.
C      ENDIF
C-----
C Inicio de um time-step - Ponto de retorno apos cada DELTA-T
C-----
C      GRID RADIAL ----> L=1
C      GRID RETANGULAR ----> L=2
C-----
500    NTS=NTS+1
C      NITER=1
C-----
C Calculo das propriedades dos fluidos dependentes das pressoes na
C iteracao anterior (passo de tempo n)
C-----
C      CALL FPOA
C-----
C Ponto de retorno no processo iterativo dentro de uma iteracao no tempo
C-----
400    CONTINUE
C-----
C Calculo das propriedades dos fluidos dependentes das pressoes na
C iteracao atual e das pressoes nas interfaces
C-----

```

```

CALL FPPOM(PWF,WX,WY,WZ)
C-----
C Verificacao do sentido de fluxo entre os blocos e calculo das
C transmissibilidades das fases nas interfaces dos blocos pelo1AS PELO
C esquema "one point upsteam" e de suas derivadas em relacao a Sw
C-----
CALL TRANS
C-----
C Calculo das vazoes de oleo e agua e de suas derivadas
C-----
IF(VPWF) THEN
C-----
C Vazao de oleo constante -> calculo da pressao pwf no poço
C-----
      SUMT=0.0DO
      DO 90 J=1,NYR
        DO 90 K=1,NZR
          SUMT=SUMT+TOX(1,J,K,1)
90      CONTINUE
      DO 91 J=1,NYR
        DO 91 K=1,NZR
          QO(J,K)=QOT*TOX(1,J,K,1)/SUMT
91      CONTINUE
      PWF=PD(1,JPROD,1,1)-DOMX(1,JPROD,1,1)*(ZC-ZB(1,JPROD,1))-
      *      QO(JPROD,1)/TOX(1,JPROD,1,1)
      DO 80 J=1,NYR
        DO 80 K=1,NZR
          DQOP(J,K)=0.0DO
          DQOS(J,K)=0.0DO
80      CONTINUE
      IF((PWF.LT.1.001DO*PWFMIN).AND.(PWF.GT.0.999DO*PWFMIN))
      *      VPWF=.FALSE.
C para pwf "gravar" com pwfmin
      IF(PWF.LE.0.999DO*PWFMIN) THEN
        DT=DT/2.0DO
        CALL REPTS(NTS,PWFA,PWF)
        GOTO 500
      ENDIF
    ENDIF
C
IF(.NOT.VPWF) THEN
C-----
C Vazao de oleo variavel -> calculo de Qo e suas derivadas
C-----
      QOT=0.0DO
      DO 60 J=1,NYR
        DO 60 K=1,NZR
          POT=PD(1,J,K,1)-PWF-DOMX(1,J,K,1)*(ZC-ZB(1,J,1))
          QO(J,K)=TOX(1,J,K,1)*POT
          DQOP(J,K)=-WX*TOX(1,J,K,1)*(DVO(1,J,K,1)/VOMX(1,J,K,1)-
          *      DBO(1,J,K,1)/BOMX(1,J,K,1))*POT+(1.0DO-WX*
          *      DDO(1,J,K,1)*(ZC-ZB(1,J,1)))*TOX(1,J,K,1)
          DQOS(J,K)=POT*TOX1SI(1,J,K,1)
          QOT=QOT+QO(J,K)
60      CONTINUE
    ENDIF
C-----
C Calculo da vazao de agua e gas e suas derivadas
C Qw=Qo*(mobilidade agua), suas derivadas existem
C Qg=Qo*Rs ( considerando que nao havera gas livre no reservatorio )
C Razao de solubilidade na pressao de saturacao
C-----
CALL INTER(POL,RSOL,NPO,POLS(1),RSS,AUX)

```

```

QWT=0.0D0
QGT=0.0D0
DO 70 J=1,NYR
  DO 70 K=1,NZR
    RMWO=(KRW(1,J,K,1)*BW(1,J,K,1)*VO(1,J,K,1))/
      (KRO(1,J,K,1)*BO(1,J,K,1)*VW(1,J,K,1))
    *
    QW(J,K)=QO(J,K)*RMWO
    AUX=DVO(1,J,K,1)/VO(1,J,K,1)-DBO(1,J,K,1)/BO(1,J,K,1)-
    *
      DVW(1,J,K,1)/VW(1,J,K,1)+DBW(1,J,K,1)/BW(1,J,K,1)
    DQWP(J,K)=RMWO*(AUX*QO(J,K)+DQOP(J,K))
    AUX=DKRW(1,J,K,1)-KRW(1,J,K,1)*DKRO(1,J,K,1)/KRO(1,J,K,1)
    *
    AUX=AUX*BW(1,J,K,1)*VO(1,J,K,1)/(KRO(1,J,K,1)*
      BO(1,J,K,1)*VW(1,J,K,1))
    DQWS(J,K)=RMWO*DQOS(J,K)+AUX*QO(J,K)
    QWT=QWT+QW(J,K)
    QG(J,K)=QO(J,K)*RSS
    QGT=QGT+QG(J,K)
70  CONTINUE
C
  IF(QOT.NE.0.0D0) RWQ=QWT/QOT
  IF(QOT.NE.0.0D0) RGT=QGT/QOT
C-----
C Calculo das funcoes de residuos do oleo e da agua e de suas
C derivadas para a montagem da matriz Jacobiana
C-----
  CALL MONJAC
C-----
C Solucao do sistema linear de equacoes
C 1- por metodo direto usando Gauss
C 2- por metodo direto usando algoritmo otimizado (NSPIV)
C 3- por metodo iterativo (BSOR)
C-----
  IF(MSOLU.EQ.1) CALL SOLVE1
  IF(MSOLU.EQ.2) CALL SOLVE2
  IF(MSOLU.EQ.3) CALL SOLVE3
C-----
C Atualizacao dos valores de pressao e saturacao
C-----
  NC=0
  LC=0
  DO 20 I=1,NXC
    DO 20 J=1,NYC
      DO 20 K=1,NZC
        IF(I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.K.LT.KP+NBP) GOTO 20
        NC=NC+1
        LC=2*NC-1
        PO(I,J,K,2)=PO(I,J,K,2)+XC(LC)
        SW(I,J,K,2)=SW(I,J,K,2)+XC(LC+1)
        IF(SW(I,J,K,2).GT.1.0D0) SW(I,J,K,2)=1.0D0
        IF(SW(I,J,K,2).LT.SWIN) SW(I,J,K,2)=SWIN
        SO(I,J,K,2)=1.0D0-SW(I,J,K,2)
20  CONTINUE
  NC=0
  LC=0
  DO 21 I=1,NXR
    DO 21 J=1,NYR
      DO 21 K=1,NZR
        NC=NC+1
        LC=2*NC-1
        PO(I,J,K,1)=PO(I,J,K,1)+XR(LC)
        SW(I,J,K,1)=SW(I,J,K,1)+XR(LC+1)

```

```

                IF(SW(I,J,K,1).GT.1.0D0) SW(I,J,K,1)=1.0D0
                IF(SW(I,J,K,1).LT.SWIN) SW(I,J,K,1)=SWIN
                SO(I,J,K,1)=1.0D0-SW(I,J,K,1)
21  CONTINUE
C-----
C  Analise da convergencia
C  Verificacao das variacoes da saturacao e pressao
C-----
        CONV=.TRUE.
        NC=0
        LC=0
        DO 25 I=1,NXC
            DO 25 J=1,NYC
                DO 25 K=1,NZC
                    IF(I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.K.LT.KP+NBP) GOTO 25
                    NC=NC+1
                    LC=2*NC-1
                    IF(DABS(XC(LC)).GT.TOLP) CONV=.FALSE.
                    IF(DABS(XC(LC+1)).GT.TOLS) CONV=.FALSE.
25  CONTINUE
        NC=0
        LC=0
        DO 26 I=1,NXR
            DO 26 J=1,NYR
                DO 26 K=1,NZR
                    NC=NC+1
                    LC=2*LC-1
                    IF(DABS(XR(LC)).GT.TOLP) CONV=.FALSE.
                    IF(DABS(XR(LC+1)).GT.TOLS) CONV=.FALSE.
26  CONTINUE
        IF(CONV) THEN
            write(*,*)' dt=',dt/86400.0,' tempo=',tempo/86400.0
C-----
C  Determinacao das maximas variacoes de sat. e pr. no passo de tempo
C-----
        VDS=.TRUE.
        DS=0.0D0
        DP=0.0D0
        DO 30 I=1,NXC
            DO 30 J=1,NYC
                DO 30 K=1,NZC
                    IF(I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.K.LT.KP+NBP) GOTO 30
                    AUX=DABS(SO(I,J,K,2)-SOA(I,J,K,2))
                    IF(AUX.GT.DS) DS=AUX
                    AUX=DABS(SW(I,J,K,2)-SWA(I,J,K,2))
                    IF(AUX.GT.DS) DS=AUX
                    AUX=DABS(PO(I,J,K,2)-POA(I,J,K,2))
                    IF(AUX.GT.DP) DP=AUX
30  CONTINUE
        DO 31 I=1,NXR
            DO 31 J=1,NYR
                DO 31 K=1,NZR
                    AUX=DABS(SO(I,J,K,1)-SOA(I,J,K,1))
                    IF(AUX.GT.DS) DS=AUX
                    AUX=DABS(SW(I,J,K,1)-SWA(I,J,K,1))
                    IF(AUX.GT.DS) DS=AUX
                    AUX=DABS(PO(I,J,K,1)-POA(I,J,K,1))
                    IF(AUX.GT.DP) DP=AUX
31  CONTINUE
        IF(DS.GT.1.2D0*DSLIM) VDS=.FALSE.
        IF(DP.GT.1.2D0*DPLIM) VDS=.FALSE.
        IF(.NOT.VDS) THEN
            write(*,*)'-----> CONVERGIU mas nao passou na ds e dp'

```

```

C-----
C Nao verificaram-se os limit. de var. de sat.(DSLIM) e/ou de pr.(DPLIM)
C-----
      DS=DSLIM/DS
      DP=DPLIM/DP
      IF(DP.LE.DS) THEN
        DT=DT*DP
      ELSE
        DT=DT*DS
      ENDIF
      CALL REPTS(NTS,PWFA,PWF)
      GOTO 500
    ENDIF
C-----
C Se chegou-se ate esse ponto tem-se um passo de tempo bem sucedido
C-----
      DO 35 L=1,2
        NX=NXR
        NY=NYS
        NZ=NZR
        IF(L.EQ.2) THEN
          NX=NXC
          NY=NYC
          NZ=NZC
        ENDIF
        DO 35 I=1,NX
          DO 35 J=1,NY
            DO 35 K=1,NZ
              VPOR(I,J,K,L)=VPOR(I,J,K,L)*(1.0D0+CR*(PO(I,J,K,L)-
                POA(I,J,K,L)))
              POA(I,J,K,L)=PO(I,J,K,L)
              SOA(I,J,K,L)=SO(I,J,K,L)
              SWA(I,J,K,L)=SW(I,J,K,L)
              SGA(I,J,K,L)=SG(I,J,K,L)
              IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.K.LT.KP+NBK)
                VPOR(I,J,K,L)=0.0D0
            *
          *
        CONTINUE
      35
C-----
C Calculo das producoes de oleo, gas e agua acumulados
C-----
      PWFA=PWF
      QOAC=QOAC+QOT*DT
      QGAC=QGAC+QGT*DT
      QWAC=QWAC+QWT*DT
      TEMPO=TEMPO+DT
C-----
C Opcao para teste de formacao
C-----
      IF(TESTE) THEN
        CALL CTESTE(TESTE,NTS,DT,QOMAX,TEMPTF,NITER)
        GOTO 500
      ENDIF
C-----
C Controle de impressao de resultados para relatorio e graficos
C-----
      IF(((TSIM.NE.0.0D0).AND.(TEMPO.GE.TSIM)).OR.(QOT.LE.QOMIN)) THEN
        CALL BALM
        CALL IMPRER(DT,NTS)
        CALL IMPREG
        write(*,40)tempo/86400.,pwf/1.0D+6,rwo,dt/86400.,sw(1,1,1,1)
        GOTO 50
      ENDIF
      IF(TEMPO.GE.ITR*TIMPR) THEN
        ITR=ITR+1
        CALL BALM

```

```

      CALL IMPRER(DT,NTS)
      ENDIF
      IF(TEMPO.GE.ITG*TIMPG) THEN
        write(*,40)tempo/86400.,pwf/1.0D+6,rwo,dt/86400.,sw(1,1,1,1)
40    format(' tempo=',f8.3,' pwf=',f6.3,' rwo=',e10.4,
      *' dt=',f7.4,' sw=',f6.4)
        ITG=ITG+1
        CALL BALM
        CALL IMPREG
      ENDIF
C-----
C Controle automatico de DT ao final de um passo de tempo bem sucedido
C Quando Pwf se aproxima da pressao de sat. no poco reduz-se DT
C-----
      IF((PWF.GE.0.95D0*POLS(1)).AND.(PWF.LE.1.05D0*POLS(1)))
      *   DT=DT/2.0D0
        IF(NITER.LE.ITMIN) DT=1.5D0*DT
        IF(DT.GT.DTMAX) DT=DTMAX
        IF(DT.LT.DTMIN) DT=DTMIN
        GOTO 500
      ENDIF
C=====
C nao se obteve convergencia,assim uma nova iteracao sera feita
C=====
      write(*,*)'-----> nao convergiu dt=',dt/86400.
      NITER=NITER+1
C-----
C n# de iteracoes em um passo de tempo superou o maximo,logo reduz-se DT
C e reinicia-se o processo do inicio.Chama-se portanto rotina REPTS.
C-----
      IF(NITER.GT.ITMAX) THEN
        DT=DT/2.0D0
        CALL REPTS(NTS,PWFA,PWF)
        GOTO 500
      ENDIF
      GOTO 400
C
50  in=_STOPW(1)
      WRITE(4,*)'      Tempo de execucao (seg) =',in/1000
      CLOSE(UNIT=4,STATUS='KEEP')
      CLOSE(UNIT=6,STATUS='KEEP')
      CLOSE(UNIT=7,STATUS='KEEP')
C
      STOP
      END
C-----
C SUBROTINA DE LEITURA DOS DADOS DE ENTRADA
C-----
      SUBROUTINE ENTRA
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      PARAMETER(NLAB=18,NXM=08,NYM=05,NZM=04)
C
      COMMON/LPVT0/POL,BOL,VOL,DOL,RSOL,POLS,BOLS,VOLS,DOLS,NPO,NPOS
      COMMON/LPVTW/PWL,BWL,VWL,DWL,RSWL,NPW
      COMMON/LPVTG/PGL,BGL,VGL,DGL,NPG
      COMMON/LPETR/SWL,KRWL,KROL,SLL,PCOWL,NKR,NPC
      COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
      *   NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
      COMMON/SIM/DT,DTMAX,DTMIN,TOLP,TOLS,DSLIM,DPLIM,TSIM,

```

```

*          QOMAX,QOMIN,PWMIN,BSOR,JPROD,MSOLU,ITMAX,ITMIN
COMMON/RES/DGAS,CR,PRIN,TEMP,ZC,RW,RE,SWIN,PHI,PERX,PERY,
*          PERZ,ZCOW,INDPOR,INDPER
COMMON/IMP/TIMPR,TIMPG,IRIMP
COMMON/TFR/QOTF,TPTF,TETF,DTTF,DTI,TFLUX,BETA,OPT
C
  DIMENSION POL(NLAB),BOL(NLAB),VOL(NLAB),DOL(NLAB),RSOL(NLAB),
*          POLS(NLAB),BOLS(NLAB),VOLS(NLAB),DOLS(NLAB)

  DIMENSION PWL(NLAB),BWL(NLAB),VWL(NLAB),DWL(NLAB),RSWL(NLAB)
  DIMENSION PGL(NLAB),BGL(NLAB),VGL(NLAB),DGL(NLAB)
  DIMENSION SWL(NLAB),SLL(NLAB),PCOWL(NLAB)
  REAL*8    KRWL(NLAB),KROL(NLAB)
  DIMENSION PHI(NXM,NYM,NZM,2),PERX(NXM,NYM,NZM,2),
*          PERY(NXM,NYM,NZM,2),PERZ(NXM,NYM,NZM,2)
  LOGICAL SINAL,SINAL1
  CHARACTER*1 OPT,FLAG
-----
C
C Leitura dos dados PVT para oleo,agua,gas
C
  READ(1,*,ERR=200)
  READ(1,*,ERR=200)
  READ(1,*,ERR=200)

C
  I=0
  5   I=I+1
      READ(1,100,ERR=200)POL(I),BOL(I),VOL(I),DOL(I),RSOL(I),FLAG
      IF(FLAG.EQ.' ') GOTO 5
      NPO=I

C
      READ(1,*,ERR=200)
      READ(1,*,ERR=200)

C
  I=0
  10  I=I+1
      READ(1,110,ERR=200)POLS(I),BOLS(I),VOLS(I),DOLS(I),FLAG
      IF(FLAG.EQ.' ') GOTO 10
      NPOS=I

C
      READ(1,*,ERR=200)
      READ(1,*,ERR=200)

C
  I=0
  15  I=I+1
      READ(1,100,ERR=200)PWL(I),BWL(I),VWL(I),DWL(I),RSWL(I),FLAG
      IF(FLAG.EQ.' ') GOTO 15
      NPW=I

C
      READ(1,*,ERR=200)
      READ(1,*,ERR=200)

C
  I=0
  20  I=I+1
      READ(1,115,ERR=200)PGL(I),BGL(I),VGL(I),DGL(I),FLAG
      IF(FLAG.EQ.' ') GOTO 20
      NPG=I

C
  100  FORMAT(5F10.4,A1)
  110  FORMAT(4F10.4,10X,A1)
  115  FORMAT(F10.4,2F10.6,F10.4,10X,A1)
      CLOSE(UNIT=1,STATUS='KEEP')
-----
C
C Leitura de dados de pressao capilar e de permeabilidade relativa

```

```

C-----
  READ(2,*,ERR=200)
  READ(2,*,ERR=200)
  READ(2,*,ERR=200)
C
  I=0
25  I=I+1
    READ(2,120,ERR=200)SWL(I),KRWL(I),KROL(I),FLAG
    IF(FLAG.EQ.' ') GOTO 25
    NKR=I
C
  READ(2,*,ERR=200)
  READ(2,*,ERR=200)
C
  I=0
30  I=I+1
    READ(2,125,ERR=200)SLL(I),PCOWL(I),FLAG
    IF(FLAG.EQ.' ') GOTO 30
    NPC=I
C
120  FORMAT(3F10.6,A1)
125  FORMAT(F10.6,F10.3,10X,A1)
    CLOSE(UNIT=2,STATUS='KEEP')

```

```

C-----
C Leitura dos parametros de simulacao
C-----

```

```

  READ(3,*,ERR=200)
  READ(3,*,ERR=200)
  READ(3,130,ERR=200)NXC
  READ(3,130,ERR=200)NYC
  READ(3,130,ERR=200)NZC
  READ(3,155,ERR=200)DBDX
  READ(3,155,ERR=200)DBDY
  READ(3,155,ERR=200)DBDZ
  READ(3,130,ERR=200)NBP
  READ(3,130,ERR=200)IP
  READ(3,130,ERR=200)JP
  READ(3,130,ERR=200)KP
  READ(3,130,ERR=200)NXR
  READ(3,130,ERR=200)NQL
  READ(3,130,ERR=200)NQS
  READ(3,130,ERR=200)NQI
C Determinacao do numero de blocos na direcao y
C
  NZR=NBP
  NYR=2*NQL+NQS+NQI
C
  READ(3,130,ERR=200)MSOLU
  READ(3,130,ERR=200)ITMAX
  READ(3,130,ERR=200)ITMIN
  READ(3,140,ERR=200)DT
  READ(3,135,ERR=200)DTMAX
  READ(3,145,ERR=200)DTMIN
  READ(3,140,ERR=200)TOLP
  READ(3,145,ERR=200)TOLS
  READ(3,145,ERR=200)DSLIM
  READ(3,140,ERR=200)DPLIM
  READ(3,140,ERR=200)QOMAX
  READ(3,140,ERR=200)QOMIN
  READ(3,155,ERR=200)TSIM

```

```

READ(3,140,ERR=200)PWMIN
READ(3,150,ERR=200)RW
READ(3,135,ERR=200)TIMPR
READ(3,135,ERR=200)TIMPG
READ(3,130,ERR=200)IRIMP
READ(3,130,ERR=200)JPROD
IF(IRIMP.EQ.0) IRIMP=1
IF(TIMPR.EQ.0.0D0) TIMPR=182.5D0
IF(TIMPG.EQ.0.0D0) TIMPG=TIMPR

```

```

C
  READ(3,*,ERR=200)

```

```

C-----
C Leitura dos dados do teste de formacao
C-----

```

```

  READ(3,165,ERR=200)OPT
  READ(3,155,ERR=200)QOTF
  READ(3,155,ERR=200)TPTF
  READ(3,155,ERR=200)TETF
  READ(3,155,ERR=200)DTTF
  READ(3,155,ERR=200)BETA

```

```

C
  READ(3,*,ERR=200)

```

```

C-----
C Leitura dos dados do reservatorio
C-----

```

```

  READ(3,145,ERR=200)SWIN
  READ(3,145,ERR=200)DGAS
  READ(3,160,ERR=200)CR
  READ(3,150,ERR=200)PRIN
  READ(3,150,ERR=200)TEMP
  READ(3,155,ERR=200)ZC
  READ(3,155,ERR=200)ZCOW
  RE=DSQRT(DBDX*DBDY/3.141592654D0)

```

```

C
130  FORMAT(35X,I3)
135  FORMAT(35X,F5.1)
140  FORMAT(35X,F8.3)
145  FORMAT(35X,F6.4)
150  FORMAT(35X,F8.4)
155  FORMAT(35X,F10.4)
160  FORMAT(35X,D9.3)
165  FORMAT(35X,A1)
166  FORMAT(35X,3(I2,1X))

```

```

C
  READ(3,*,ERR=200)

```

```

C-----
C Bloco de leitura do vetor porosidade a Patm ,um valor por camada,
C associado a um numero NPHI de camadas sobrepostas (de baixo para cima)
C-----

```

```

  READ(3,170,ERR=200)INDPOR
170  FORMAT(I1)
      GOTO(180,190),INDPOR
180  READ(3,210,ERR=200)AUXP
210  FORMAT(F8.2)
      DO 220 I=1,NXR
        DO 220 J=1,NYR
          DO 220 K=1,NZR
            PHI(I,J,K,1)=AUXP
220  CONTINUE
      DO 221 I=1,NXC

```

```

      DO 221 J=1,NYC
        DO 221 K=1,NZC
          PHI(I,J,K,2)=AUXP
221    CONTINUE
      GOTO 230
190    WRITE(*,*)'NAO DEFINIDA OPCAO PARA POROSIDADE POR BLOCOS'
C
230  READ(3,*,ERR=200)
C-----
C Bloco de leitura da permeabilidades na direcao x
C-----
      READ(3,170,ERR=200)INDPER
      GOTO(240,250),INDPER
240    READ(3,210,ERR=200)AUXP
      DO 260 I=1,NXR
        DO 260 J=1,NYR
          DO 260 K=1,NZR
            PERX(I,J,K,1)=AUXP
260    CONTINUE
      DO 261 I=1,NXC
        DO 261 J=1,NYC
          DO 261 K=1,NZC
            PERX(I,J,K,2)=AUXP
261    CONTINUE
      GOTO 270
250    WRITE(*,*)'NAO DEFINIDA OPCAO PARA PERMEABILIDADE POR BLOCOS'
C
270  READ(3,*,ERR=200)
C-----
C Bloco de leitura da permeabilidades na direcao y
C-----
      READ(3,170,ERR=200)INDPER
      GOTO(280,290),INDPER
280    READ(3,210,ERR=200)AUXP
      DO 300 I=1,NXR
        DO 300 J=1,NYR
          DO 300 K=1,NZR
            PERY(I,J,K,1)=AUXP
300    CONTINUE
      DO 301 I=1,NXC
        DO 301 J=1,NYC
          DO 301 K=1,NZC
            PERY(I,J,K,2)=AUXP
301    CONTINUE
      GOTO 310
290    WRITE(*,*)'NAO DEFINIDA OPCAO PARA PERMEABILIDADE POR BLOCOS'
310  READ(3,*,ERR=200)
C-----
C Bloco de leitura da permeabilidades na direcao y
C-----
      READ(3,170,ERR=200)INDPER
      GOTO(320,330),INDPER
320    READ(3,210,ERR=200)AUXP
      DO 340 I=1,NXR
        DO 340 J=1,NYR
          DO 340 K=1,NZR
            PERZ(I,J,K,1)=AUXP
340    CONTINUE
      DO 341 I=1,NXC
        DO 341 J=1,NYC
          DO 341 K=1,NZC
            PERZ(I,J,K,2)=AUXP

```

```

341      CONTINUE
      GOTO 350
330      WRITE(*,*) 'NAO DEFINIDA OPCAO PARA PERMEABILIDADE POR BLOCOS'
350      RETURN
C-----
C Caso ocorra erro na leitura de dados de permeabilidade
C-----
      CLOSE(UNIT=3,STATUS='KEEP')
200      PRINT*, 'ERRO NA LEITURA DE ARQUIVO DE DADOS'
C
      RETURN
      END
C-----
C SUBROTINA DE IMPRESSAO DOS DADOS LIDOS PARA SUA VERIFICACAO
C-----
      SUBROUTINE IMPDE
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      PARAMETER(NLAB=18,NXM=08,NYM=05,NZM=04)
C
      COMMON/LPVT0/POL,BOL,VOL,DOL,RSOL,POLS,BOLS,VOLS,DOLS,NPO,NPOS
      COMMON/LPVTW/PWL,BWL,VWL,DWL,RSWL,NPW
      COMMON/LPVTG/PGL,BGL,VGL,DGL,NPG
      COMMON/LPETR/SWL,KRWL,KROL,SLL,PCOWL,NKR,NPC
      COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*          NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
      COMMON/SIM/DT,DTMAX,DTMIN,TOLP,TOLS,DSLIM,DPLIM,TSIM,
*          QOMAX,QOMIN,PWMIN,BSOR,JPROD,MSOLU,ITMAX,ITMIN
      COMMON/RES/DGAS,CR,PRIN,TEMP,ZC,RW,RE,SWIN,PHI,PERX,PERY,
*          PERZ,ZCOW,INDPOR,INDPER
      COMMON/IMP/TIMPR,TIMPG,IRIMP
      COMMON/TFR/QOTF,TPTF,TETF,DTTF,DTI,TFLUX,BETA,OPT
C
      DIMENSION POL(NLAB),BOL(NLAB),VOL(NLAB),DOL(NLAB),RSOL(NLAB),
*          POLS(NLAB),BOLS(NLAB),VOLS(NLAB),DOLS(NLAB)
      DIMENSION PWL(NLAB),BWL(NLAB),VWL(NLAB),DWL(NLAB),RSWL(NLAB)
      DIMENSION PGL(NLAB),BGL(NLAB),VGL(NLAB),DGL(NLAB)
      DIMENSION SWL(NLAB),SLL(NLAB),PCOWL(NLAB)
      REAL*8    KRWL(NLAB),KROL(NLAB)
      DIMENSION PHI(NXM,NYM,NZM,2),PERX(NXM,NYM,NZM,2),
*          PERY(NXM,NYM,NZM,2),PERZ(NXM,NYM,NZM,2)
      CHARACTER*1 OPT
C
      Impressao do cabecalho
      WRITE(04,4)
4      FORMAT(//,30X,'PROGRAMA ACTIOW4.FOR',
*//,20X,'SIMULADOR DE RESERVATORIO COM POÇO HORIZONTAL',/,
*20X,45('='),//,30X,'TOTALMENTE IMPLICITO',/,30X,'TRIDIMENSIONAL',/
*,30X,'BI-FASICO ( OLEO/AGUA )',//)
C
      GRAVACAO DOS PARAMETROS DE SIMULACAO,DADOS DO RESEV. E DO TESTE
C
      WRITE(04,3)NXC,NYC,NZC,DBDX,DBDY,DBDZ,NBP,IP,JP,KP
3      FORMAT(3X,'PRINCIPAIS DADOS UTILIZADOS',/,3X,25('='),//,
*3X,'Parametros de Simulacao :',//,
*5X,'Regiao do reservatorio - GRID CARTESIANO -',/,
* '      N# de blocos na direcao X          (I3) NXC==>',I3,/,
* '      N# de blocos na direcao Y          (I3) NYC==>',I3,/,
* '      N# de blocos na direcao Z          (I3) NZC==>',I3,/,
* '      Dim. dos blocos direcao X          (F10.4) DBDX==>',F9.2,/,

```

```

*'      Dim. dos blocos direcao Y          (F10.4) DBDY===>',F9.2,/,
*'      Dim. dos blocos direcao Z          (F10.4) DBDZ===>',F9.2,/,
*'      N. de blocos produtores            (I3)  NBP===>',I3,/,
*'      Coord. i do bloco produtor          (I3)  IP===>',I3,/,
*'      Coord. j do bloco produtor          (I3)  JP===>',I3,/,
*'      Coord. k do primeiro bloco produtor (I3)  KP===>',I3)

C
  WRITE(04,5)NXR,NYR,NZR,DT,TOLP,TOLS,DSLIM,DPLIM,QOMAX,QOMIN,TSIM,
  *PWMIN,RW,TIMPR,TIMPG
5  FORMAT(4X,'Regiao do poço - GRID RADIAL -',/,
*'      N# de blocos na direcao X          (I3) NX===>',I3,/,
*'      N# de blocos na direcao Y          (I3) NY===>',I3,/,
*'      N# de blocos na direcao Z          (I3) NZ===>',I3,/,
*'      Time step inicial (dias)            (F8.3) DT===>',F8.3,/,
*'      Tolerancia para pressao (Pa)        (F8.3) TOLP===>',F8.3,/,
*'      Tolerancia para saturacao (fr)      (F6.4) TOLS===>',F6.4,/,
*'      Max var de sat num DT (fracao)      (F6.4) DSLIM===>',F6.4,/,
*'      Max var de pres. num DT (KPa)       (F8.3) DPLIM===>',F8.3,/,
*'      Vazao de óleo maxima (m3/dia)       (F8.3) QOMAX===>',F8.3,/,
*'      Vazao de óleo minima (m3/dia)       (F8.3) QOMIN===>',F8.3,/,
*'      Tempo final de simulacao (dias)     (F8.2) TSIM===>',F8.2,/,
*'      Pwf minima no poço (MPa)            (F8.3) PWMIN===>',F8.3,/,
*'      Raio do poço (m)                    (F8.4) RW===>',F8.4,/,
*'      Int. tempo p/ impr.relatorio (dias) (F5.1) TIMPR===>',F5.1,/,
*'      Int. tempo p/ impr.grafico (dias)   (F5.1) TIMPG===>',F5.1)
  WRITE(04,6)IRIMP,DTMAX,DTMIN,ITMAX,ITMIN,JPROD
6  FORMAT(5X,'Int. celulas de impressao    (I3) IRIMP===>',I3,/,
*'      Incremento de tempo maximo (dias)   (F5.1) DTMAX===>',F5.1,/,
*'      Incremento de tempo minimo (dias)   (F6.4) DTMIN===>',F6.4,/,
*'      N# max de iteracoes no JACOBIANO    (I3) ITMAX===>',I3,/,
*'      N# min de iteracoes no JACOBIANO    (I3) ITMIN===>',I3,/,
*'      Fatia J base para o calculo de Pwf  (I3) JPROD===>',I3,/)

C
  WRITE(04,10)OPT,QOTF,TPTF,TETF,DTTF
10 FORMAT(3X,'Dados do teste de formacao :',/,
*'      Simulacao de teste de formacao      (S ou N) OPT===>',A1,/,
*'      Vazao de producao (m3/dia)          (F8.3) QOTF===>',F8.3,/,
*'      Tempo de fluxo (horas)              (F8.3) TPTF===>',F8.3,/,
*'      Tempo de estatica (horas)           (F8.3) TETF===>',F8.3,/,
*'      Incr. de tempo no teste (horas)     (F8.3) DTF===>',F8.3,/)

C
  WRITE(04,15)SWIN,DGAS,CR,PRIN,TEMP,RE,ZC,ZCOW
15 FORMAT(3X,'Dados do reservatorio :',/,
*'      Saturacao inicial de agua (fracao)  (F6.4) SWIN===>',F6.4,/,
*'      Densidade do gas (adim)              (F6.4) DGAS===>',F6.4,/,
*'      Compressibilidade da rocha (1/MPa)   (E9.3) CR===>',E9.3,/,
*'      Pressao inicial do reservatorio(MPa) (F8.4) PRIN===>',F8.4,/,
*'      Temperatura do reservatorio (°C)     (F8.4) TEMP===>',F8.4,/,
*'      Raio externo da regiao do poço (m)  (F8.4) RE===>',F8.3,/,
*'      Cota do poço (m)                     (F8.3) ZC===>',F8.2,/,
*'      Cota do contato o/w (m)              (F8.3) ZCOW===>',F8.2)
  WRITE(04,25)PHI(1,1,1,1)
25 FORMAT(5X,'Porosidade igual a ',F10.4,' para todos os blocos')
  WRITE(04,30)PERX(1,1,1,1)
  WRITE(04,31)PERY(1,1,1,1)

```

```

WRITE(04,32)PERZ(1,1,1,1)
30  FORMAT(5X,'Permeabilidade na direcao X igual a',F10.4,' para todos
    * blocos')
31  FORMAT(5X,'Permeabilidade na direcao Y igual a',F10.4,' para todos
    * blocos')
32  FORMAT(5X,'Permeabilidade na direcao Z igual a',F10.4,' para todos
    * blocos')
C
C
C  GRAVACAO DOS DADOS DE PVT
WRITE(04,50)NPO
50  FORMAT(/,3X,'NPO=',I3,' DADOS PVT DO OLEO SATURADO',//,3X,' Pol(MP
    *a) Bol(m3/m3) Vol(cp) Dol(kg/m3) RSol(m3/m3)',/)
    DO 55 I=1,NPO
        WRITE(04,60)POL(I),BOL(I),VOL(I),DOL(I),RSOL(I)
60  FORMAT(3X,5(F8.3,2X))
55  CONTINUE
WRITE(04,65)NPOS
65  FORMAT(/,3X,'NPOS=',I3,' DADOS PVT DO OLEO SUBSATURADO',//,3X,' Po
    *ls(MPa) Bols(m3/m3) Vols(cp) Dols(kg/m3)',/)
    DO 70 I=1,NPOS
        WRITE(04,75)POL(I),BOLS(I),VOLS(I),DOLS(I)
75  FORMAT(3X,3(F8.3,2X),4X,F8.3)
70  CONTINUE
WRITE(04,80)NPW
80  FORMAT(/,3X,'NPW=',I3,' DADOS PVT DA AGUA',//,3X,' Pwl(MPa) Bwl(m
    *3/m3) Vwl(cp) Dw1(kg/m3) RSw1(m3/m3)',/)
    DO 85 I=1,NPW
        WRITE(04,90)PWL(I),BWL(I),VWL(I),DWL(I),RSWL(I)
90  FORMAT(3X,5(F9.4,2X))
85  CONTINUE
WRITE(04,95)NPG
95  FORMAT(/,3X,'NPG=',I3,' DADOS PVT DO GAS ',//,3X,' Pgl(MPa) Bgl(m
    *3/m3) Vgl(cp) Dgl(kg/m3)',/)
    DO 100 I=1,NPG
        WRITE(04,105)PGL(I),BGL(I),VGL(I),DGL(I)
105  FORMAT(3X,F8.3,2X,F8.6,2X,F8.5,2X,F8.3)
100 CONTINUE
C
C
C  GRAVACAO DOS DADOS DE PETROFISICA
WRITE(04,110)NKR
110 FORMAT(/,3X,'NKR=',I3,' DADOS DE PERMEABILIDADE RELATIVA OLEO/AGUA
    *,//,3X,' Swl      Krwl      Krol',/)
    DO 115 I=1,NKR
        WRITE(04,120)SWL(I),KRWL(I),KROL(I)
120  FORMAT(3X,3(F8.5,2X))
115 CONTINUE
WRITE(04,125)NPC
125 FORMAT(/,3X,'NPC=',I3,' DADOS DE PRESSAO CAPILAR OLEO/AGUA',//,3X,
    *' Sll      Pcow(Pa)',/)
    DO 130 I=1,NPC
        WRITE(04,135)SLL(I),PCOWL(I)
135  FORMAT(3X,2(F12.5,2X))
130 CONTINUE
C
C  RETURN
END
C
C

```

C SUBR. DE CONVERSAO DE UNIDADES PARA O S.I. E CALC. DOS PESOS ESP. N/M3

C-----
C SUBROUTINE CONVUNI

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
PARAMETER(NLAB=18,NXM=08,NYM=05,NZM=04)

COMMON/LPVTQ/POL,BOL,VOL,DOL,RSOL,POLS,BOLS,VOLS,DOLS,NPO,NPOS
COMMON/LPVTW/PWL,BWL,VWL,DWL,RSWL,NPW
COMMON/LPVTG/PGL,BGL,VGL,DGL,NPG
COMMON/LPETR/SWL,KRWL,KROL,SLL,PCOWL,NKR,NPC
COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
* NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
COMMON/SIM/DT,DTMAX,DTMIN,TOLP,TOLS,DSLIM,DPLIM,TSIM,
* QOMAX,QOMIN,PWMIN,BSOR,JPROD,MSOLU,ITMAX,ITMIN
COMMON/RES/DGAS,CR,PRIN,TEMP,ZC,RW,RE,SWIN,PHI,PERX,PERY,
* PERZ,ZCOW,INDPOR,INDPER
COMMON/IMP/TIMPR,TIMPG,IRIMP
COMMON/TFR/QOTF,TPTF,TETF,DTTF,DTI,TFLUX,BETA,OPT

C
* DIMENSION POL(NLAB),BOL(NLAB),VOL(NLAB),DOL(NLAB),RSOL(NLAB),
* POLS(NLAB),BOLS(NLAB),VOLS(NLAB),DOLS(NLAB)
DIMENSION PWL(NLAB),BWL(NLAB),VWL(NLAB),DWL(NLAB),RSWL(NLAB)
DIMENSION PGL(NLAB),BGL(NLAB),VGL(NLAB),DGL(NLAB)
DIMENSION SWL(NLAB),SLL(NLAB),PCOWL(NLAB)
REAL*8 KRWL(NLAB),KROL(NLAB)
DIMENSION PHI(NXM,NYM,NZM,2),PERX(NXM,NYM,NZM,2),
* PERY(NXM,NYM,NZM,2),PERZ(NXM,NYM,NZM,2)
CHARACTER*1 OPT

C
C Transforma-se MPa para Pa , cp para kg/s*m , massa esp. para peso esp.

DO 5 I=1,NPO
POL(I)=1.0D+06*POL(I)
VOL(I)=1.0D-03*VOL(I)
DOL(I)=9.8067D0*DOL(I)
BOL(I)=1.0D0/BOL(I)

5 CONTINUE
DO 10 I=1,NPOS
POLS(I)=1.0D+06*POLS(I)
VOLS(I)=1.0D-03*VOLS(I)
DOLS(I)=9.8067D0*DOLS(I)
BOLS(I)=1.0D0/BOLS(I)

10 CONTINUE
DO 15 I=1,NPW
PWL(I)=1.0D+06*PWL(I)
VWL(I)=1.0D-03*VWL(I)
DWL(I)=9.8067D0*DWL(I)
BWL(I)=1.0D0/BWL(I)

15 CONTINUE
DO 20 I=1,NPG
PGL(I)=1.0D+06*PGL(I)
VGL(I)=1.0D-03*VGL(I)
DGL(I)=9.8067D0*DGL(I)
BGL(I)=1.0D0/BGL(I)

20 CONTINUE
PRIN=1.0D+06*PRIN
PWMIN=1.0D+06*PWMIN
DPLIM=1.0D+03*DPLIM
CR=1.0D-06*CR

C
C Transforma-se md para m2 onde 1md=9.869E-16 m2

```

DO 25 I=1,NXR
  DO 25 J=1,NYR
    DO 25 K=1,NZR
      PERX(I,J,K,1)=9.869D-16*PERX(I,J,K,1)
      PERY(I,J,K,1)=9.869D-16*PERY(I,J,K,1)
      PERZ(I,J,K,1)=9.869D-16*PERZ(I,J,K,1)
25  CONTINUE
  DO 26 I=1,NXC
    DO 26 J=1,NYC
      DO 26 K=1,NZC
        PERX(I,J,K,2)=9.869D-16*PERX(I,J,K,2)
        PERY(I,J,K,2)=9.869D-16*PERY(I,J,K,2)
        PERZ(I,J,K,2)=9.869D-16*PERZ(I,J,K,2)
26  CONTINUE
C
C Vazoes de m3/dia para m3/s, tempos de dia ou hora para s
  QOMAX=QOMAX/86400.0D0
  QOMIN=QOMIN/86400.0D0
  QOTF=QOTF/86400.0D0
  DT=DT*86400.0D0
  DTF=DTF*3600.0D0
  TPTF=TPTF*3600.0D0
  TETF=TETF*3600.0D0
  DTMAX=DTMAX*86400.0D0
  DTMIN=DTMIN*86400.0D0
  TSIM=TSIM*86400.0D0
  TIMPR=TIMPR*86400.0D0
  TIMPG=TIMPG*86400.0D0
C
  RETURN
END
-----
C
C INICIALIZACAO DE VARIAVEIS DA MALHA (pressoes,prop. PVT,transmissib.)
C
C -----
C
C SUBROUTINE INICIA
  IMPLICIT DOUBLE PRECISION(A-H,O-Z)
  PARAMETER(NXM=08,NYM=05,NZM=04)
  COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*          NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
  COMMON/TS/PO,SO,SG,SW
  COMMON/PVT/BO,DBO,VO,DVO,DO,DDO,RSO,DRSO,BW,DBW,VW,DVW,DW,DDW,CW
  COMMON/IFACEX/BOMX,VOMX,DOMX,BWMX,VWMX,DWMX
  COMMON/IFACEY/BOMY,VOMY,DOMY,BWMY,VWMY,DWMY
  COMMON/IFACEZ/BOMZ,VOMZ,DOMZ,BWMZ,VWMZ,DWMZ
  COMMON/PETR/KRO,KRW,DKRO,DKRW,PC,DPC
  COMMON/DTRANS/TOX,TWX,TOY,TWY,TOZ,TWZ,
*          TOX1SA,TOX1SI,TWX1SA,TWX1SI,
*          TOX2SI,TOX2SP,TWX2SI,TWX2SP,
*          TOY1SA,TOY1SI,TWY1SA,TWY1SI,
*          TOY2SI,TOY2SP,TWY2SI,TWY2SP,
*          TOZ1SA,TOZ1SI,TWZ1SA,TWZ1SI,
*          TOZ2SI,TOZ2SP,TWZ2SI,TWZ2SP
C
  DIMENSION PO(0:NXM+1,0:NYM+1,0:NZM+1,2),SO(NXM,NYM,NZM,2),
*          SG(NXM,NYM,NZM,2),SW(NXM,NYM,NZM,2)
  DIMENSION BO(NXM,NYM,NZM,2),DBO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          VO(NXM,NYM,NZM,2),DVO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          DO(NXM,NYM,NZM,2),DDO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          BW(NXM,NYM,NZM,2),DBW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          VW(NXM,NYM,NZM,2),DVW(0:NXM+1,0:NYM+1,0:NZM+1,2),

```

```

*          DW(NXM,NYM,NZM,2),DDW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          CW(NXM,NYM,NZM,2)
  DIMENSION BOMX(0:NXM+1,NYM,NZM,2),VOMX(0:NXM+1,NYM,NZM,2),
*          DOMX(0:NXM+1,NYM,NZM,2),BWMX(0:NXM+1,NYM,NZM,2),
*          VWMX(0:NXM+1,NYM,NZM,2),DWMX(0:NXM+1,NYM,NZM,2)
  DIMENSION BOMY(NXM,NYM+1,NZM,2),VOMY(NXM,NYM+1,NZM,2),
*          DOMY(NXM,NYM+1,NZM,2),BWMY(NXM,NYM+1,NZM,2),
*          VWMY(NXM,NYM+1,NZM,2),DWMY(NXM,NYM+1,NZM,2)
  DIMENSION BOMZ(NXM,NYM,NZM+1,2),VOMZ(NXM,NYM,NZM+1,2),
*          DOMZ(NXM,NYM,NZM+1,2),BWMZ(NXM,NYM,NZM+1,2),
*          VWMZ(NXM,NYM,NZM+1,2),DWMZ(NXM,NYM,NZM+1,2)
  DIMENSION DKRO(NXM,NYM,NZM,2),DKRW(NXM,NYM,NZM,2),
*          PC(0:NXM+1,0:NYM+1,0:NZM+1,2),DPC(0:NXM+1,0:NYM+1,0:NZM+1,2)
  REAL*8   KRO(NXM,NYM,NZM,2),KRW(NXM,NYM,NZM,2)
  DIMENSION TOX(NXM+1,NYM,NZM,2),TWX(NXM+1,NYM,NZM,2),
*          TOX1SA(NXM,NYM,NZM,2),TOX1SI(NXM,NYM,NZM,2),
*          TWX1SA(NXM,NYM,NZM,2),TWX1SI(NXM,NYM,NZM,2),
*          TOX2SI(NXM,NYM,NZM,2),TOX2SP(NXM,NYM,NZM,2),
*          TWX2SI(NXM,NYM,NZM,2),TWX2SP(NXM,NYM,NZM,2)
  DIMENSION TOY(NXM,NYM+1,NZM,2),TWY(NXM,NYM+1,NZM,2),
*          TOY1SA(NXM,NYM,NZM,2),TOY1SI(NXM,NYM,NZM,2),
*          TWY1SA(NXM,NYM,NZM,2),TWY1SI(NXM,NYM,NZM,2),
*          TOY2SI(NXM,NYM,NZM,2),TOY2SP(NXM,NYM,NZM,2),
*          TWY2SI(NXM,NYM,NZM,2),TWY2SP(NXM,NYM,NZM,2)
  DIMENSION TOZ(NXM,NYM,NZM+1,2),TWZ(NXM,NYM,NZM+1,2),
*          TOZ1SA(NXM,NYM,NZM,2),TOZ1SI(NXM,NYM,NZM,2),
*          TWZ1SA(NXM,NYM,NZM,2),TWZ1SI(NXM,NYM,NZM,2),
*          TOZ2SI(NXM,NYM,NZM,2),TOZ2SP(NXM,NYM,NZM,2),
*          TWZ2SI(NXM,NYM,NZM,2),TWZ2SP(NXM,NYM,NZM,2)

```

C

```

DO 50 L=1,2
  LX=NXR
  LY=NYR
  LZ=NZR
  IF(L.EQ.2) THEN
    LX=NXC
    LY=NYC
    LZ=NZC
  ENDIF
DO 5 I=1,LX+1
  DO 5 J=1,LY
    DO 5 K=1,LZ
      TOX(I,J,K,L)=0.0D0
      TWX(I,J,K,L)=0.0D0
5    CONTINUE
DO 6 I=1,LX
  DO 6 J=1,LY+1
    DO 6 K=1,LZ
      TOY(I,J,K,L)=0.0D0
      TWY(I,J,K,L)=0.0D0
6    CONTINUE
DO 10 I=1,LX
  DO 10 J=1,LY
    DO 10 K=1,LZ+1
      TOZ(I,J,K,L)=0.0D0
      TWZ(I,J,K,L)=0.0D0
10  CONTINUE
DO 15 I=0,LX+1
  DO 15 J=0,LY+1
    DO 15 K=0,LZ+1

```

```

      PG(I,J,K,L)=0.0D0
      PC(I,J,K,L)=0.0D0
      DPC(I,J,K,L)=0.0D0
      DBO(I,J,K,L)=0.0D0
      DVO(I,J,K,L)=0.0D0
      DDO(I,J,K,L)=0.0D0
      DBW(I,J,K,L)=0.0D0
      DVW(I,J,K,L)=0.0D0
      DDW(I,J,K,L)=0.0D0
15  CONTINUE
      DO 20 I=0,LX+1
        DO 20 J=1,LY
          DO 20 K=1,LZ
            BOMX(I,J,K,L)=1.0D0
            VOMX(I,J,K,L)=1.0D0
            DOMX(I,J,K,L)=0.0D0
            BWMX(I,J,K,L)=1.0D0
            VWMX(I,J,K,L)=1.0D0
            DWMX(I,J,K,L)=0.0D0
20  CONTINUE
      DO 25 I=1,LX
        DO 25 J=1,LY+1
          DO 25 K=1,LZ
            BOMY(I,J,K,L)=1.0D0
            VOMY(I,J,K,L)=1.0D0
            DOMY(I,J,K,L)=0.0D0
            BWMY(I,J,K,L)=1.0D0
            VWMY(I,J,K,L)=1.0D0
            DWMY(I,J,K,L)=0.0D0
25  CONTINUE
      DO 26 I=1,LX
        DO 26 J=1,LY
          DO 26 K=1,LZ+1
            BOMZ(I,J,K,L)=1.0D0
            VOMZ(I,J,K,L)=1.0D0
            DOMZ(I,J,K,L)=0.0D0
            BWMZ(I,J,K,L)=1.0D0
            VWMZ(I,J,K,L)=1.0D0
            DWMZ(I,J,K,L)=0.0D0
26  CONTINUE
      DO 30 I=1,LX
        DO 30 J=1,LY
          DO 30 K=1,LZ
            TOX1SA(I,J,K,L)=0.0D0
            TWX1SA(I,J,K,L)=0.0D0
            TOX2SP(I,J,K,L)=0.0D0
            TWX2SP(I,J,K,L)=0.0D0
            TOY1SA(I,J,K,L)=0.0D0
            TWY1SA(I,J,K,L)=0.0D0
            TOY2SP(I,J,K,L)=0.0D0
            TWY2SP(I,J,K,L)=0.0D0
            TOZ1SA(I,J,K,L)=0.0D0
            TWZ1SA(I,J,K,L)=0.0D0
            TOZ2SP(I,J,K,L)=0.0D0
            TWZ2SP(I,J,K,L)=0.0D0
30  CONTINUE
50  CONTINUE
C   RETURN
    END

```

```

C-----
C
C SUBROTINA DE LANÇAMENTO DA MALHA, CÁLCULO DOS PONTOS DE MALHA E DAS
C SUAS COTAS, CÁLCULO DOS FATORES DE FORMA E DO PONDERADOR WX E WZ
C-----
C      SUBROUTINE GRID
C
C      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
C      PARAMETER(NXM=08,NYM=05,NZM=04,PI=3.141592654D0)
C      COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
C      *      NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
C      COMMON/RES/DGAS,CR,PRIN,TEMP,ZC,RW,RE,SWIN,PHI,PERX,PERY,
C      *      PERZ,ZCOW,INDPOR,INDPER
C      COMMON/TFR/QOTF,TPTF,TETF,DTTF,DTI,TFLUX,BETA,OPT
C      COMMON/MALHA/ZB,RB,VPOR,DRO,WX,WY,WZ,FFX,FFY,FFZ,DTETA
C      DIMENSION PHI(NXM,NYM,NZM,2),PERX(NXM,NYM,NZM,2),
C      *      PERY(NXM,NYM,NZM,2),PERZ(NXM,NYM,NZM,2)
C      DIMENSION ZB(0:NXM+1,NYM,2),RB(NXM+1),VPOR(NXM,NYM,NZM,2),
C      *      FFX(NXM+1,NYM,NZM,2),FFY(NXM+1,NYM,NZM,2),
C      *      FFZ(NXM+1,NYM,NZM+1,2),DTETA(NYM+1)
C      CHARACTER*1 OPT
C-----
C      === GRID RADIAL === calculo das cotas e fatores de forma
C-----
C      Calculo dos raios RB dos grids points ( o raio e o mesmo para as
C      camadas NZR e fatias NYR do mesmo I)
C-----
C      DRO=DLOG(RE/RW)/NXR
C      ALFA=DEXP(DRO)
C      ALFA=DEXP((ALFA*ALFA*DRO)/(ALFA*ALFA-1.0D0)-0.5D0)
C      DO 10 I=1,NXR
C          AUX=RW*DEXP((I-1)*DRO)
C          RB(I)=AUX*ALFA
C 10      CONTINUE
C-----
C      Formacao do grid
C      Determinacao dos angulos
C      lado direito
C-----
C      AA=PI/2.0D0
C      DO 20 J=1,NQL
C          DTETA(J)=AA/FLOAT(NQL)
C 20      CONTINUE
C-----
C      lado superior
C-----
C      DO 30 J=(NQL+1),(NQL+NQS)
C          DTETA(J)=AA/FLOAT(NQS)
C 30      CONTINUE
C-----
C      lado esquerdo
C-----
C      DO 40 J=(NQL+NQS+1),(2*NQL+NQS)
C          DTETA(J)=AA/FLOAT(NQL)
C 40      CONTINUE
C-----
C      lado inferior
C-----
C      DO 50 J=(2*NQL+NQS+1),NYR
C          DTETA(J)=AA/FLOAT(NQI)
C 50      CONTINUE
C-----
C      Calculo das cotas de cada grid point

```

```

C-----
DO 100 I=1,NXR
SUMT=(DTETA(1)/2.0D0)-(PI/4.0D0)
DO 100 J=1,NYR
  ZB(I,J,1)=ZC+RB(I)*DSIN(SUMT)
  SUMT=SUMT+(DTETA(J)+DTETA(J+1))/2.0D0
  IF(OPT.EQ.'S') ZB(I,J,1)=ZC
100 CONTINUE
C-----
C Ponderadores WX, WY e WZ nas direcoes X, Y e Z
C-----
  IF(NXR.GT.2) THEN
    WX=((NXR-2)*DRO-DLOG(RB(NXR-2)/RW))/DLOG(RB(NXR-1)/RB(NXR-2))
  ELSE
    WX=0.5D0
  ENDIF
  WY=0.5D0
  WZ=0.5D0
  WRC=(RE-RB(NXR))/(RE+DBDX/2.0D0-RB(NXR))
C-----
C Calculo dos fatores geometricos FFX para a 1# coluna i=1 (poco)
C-----
C   para a 1# coluna i=1 (poco)
C-----
  DO 120 K=1,NZR
    DO 120 J=1,NYR
      FFX(1,J,K,1)=DTETA(J)*DBDZ*PERX(1,J,K,1)/DLOG(RB(1)/RW)
120 CONTINUE
C-----
C   para i=2,NXR ;j=1,NYR ;k=1,NZR
C-----
  DO 130 I=2,NXR
    AUX=RW*DEXP((I-1)*DRO)
    DO 130 J=1,NYR
      DO 130 K=1,NZR
        AUX1=DTETA(J)*DBDZ*PERX(I-1,J,K,1)*PERX(I,J,K,1)
        FFX(I,J,K,1)=AUX1/(PERX(I,J,K,1)*
          * DLOG(AUX/RB(I-1))+PERX(I-1,J,K,1)*DLOG(RB(I)/AUX))
130 CONTINUE
C-----
C   para i=NXR+1 ;j=1,NYR
C-----
  DO 131 J=1,NYR
    DO 131 K=1,NZR
      IF(J.LE.NQL) THEN
        IF(IP.EQ.NXC) GO TO 131
        IX=IP+1
        JX=JP
        AUXK=PERX(IX,JX,KP,2)
      ENDIF
      IF(J.GT.NQL.AND.J.LE.NQL+NQS) THEN
        IF(JP.EQ.NYC) GO TO 131
        IX=IP
        JX=JP+1
        AUXK=PERY(IX,JX,KP,2)
      ENDIF
      IF(J.GT.NQL+NQS.AND.J.LE.NQS+2*NQL) THEN
        IF(IP.EQ.1) GO TO 131
        IX=IP-1
        JX=JP
        AUXK=PERX(IX,JX,KP,2)
      ENDIF
      IF(J.GT.NQS+2*NQL) THEN
        IF(JP.EQ.1) GO TO 131

```

```

IX=IP
JX=JP-1
AUXK=PERY(IX,JX,KP,2)
ENDIF
AUX1=DTETA(J)*DBDZ
FFX(NXR+1,J,K,1)=AUX1/(DLOG(RE/RB(NXR)))/PERX(NXR,J,K,1)+
*      DLOG((RE+DBDX/2.0D0)/RE)/AUXK)
C      FFX(NXR+1,J,K,1)=0.0D0
131 CONTINUE
-----
C Calculo dos fatores geometricos FFZ
C-----
DO 140 K=2,NZR
DO 140 I=1,NXR
DO 140 J=1,NYR
AUX=DTETA(J)*RW*RW*(DEXP(2.0D0*I*DRO)-
*      DEXP(2.0D0*(I-1)*DRO))
FFZ(I,J,K,1)=AUX*PERZ(I,J,K-1,1)*PERZ(I,J,K,1)/
*      (DBDZ*(PERZ(I,J,K-1,1)+PERZ(I,J,K,1)))
140 CONTINUE
C-----
C Calculo dos fatores geometricos FFZ, para a prim. e ultima face
C-----
IF(KP.NE.1) THEN
DO 141 I=1,NXR
DO 141 J=1,NYR
AUX=DTETA(J)*RW*RW*(DEXP(2.0D0*I*DRO)-
*      DEXP(2.0D0*(I-1)*DRO))
FFZ(I,J,1,1)=AUX*PERZ(IP,JP,KP-1,2)*PERZ(I,J,1,1)/
*      (DBDZ*(PERZ(IP,JP,KP-1,2)+PERZ(I,J,1,1)))
141 CONTINUE
ENDIF
C
IF(KP+NBP-1.LT.NZC) THEN
DO 142 I=1,NXR
DO 142 J=1,NYR
AUX=DTETA(J)*RW*RW*(DEXP(2.0D0*I*DRO)-
*      DEXP(2.0D0*(I-1)*DRO))
FFZ(I,J,NZR,1)=AUX*PERZ(I,J,NZR,1)*PERZ(IP,JP,KP+NBP,2)/
*      (DBDZ*(PERZ(I,J,NZR,1)+PERZ(IP,JP,KP+NBP,2)))
142 CONTINUE
ENDIF
C-----
C Calculo dos fatores geometricos FFY
C-----
AUX=DBDZ*DRO*2.0D0
DO 150 I=1,NXR
DO 150 J=1,NYR
DO 150 K=1,NZR
JMA=1
IF(J.NE.NYR) JMA=J+1
AUX2=AUX/DTETA(J)
FFY(I,J,K,1)=AUX2/(1.0D0/PERY(I,J,K,1)+1.0D0/
*      PERY(I,JMA,K,1))
150 CONTINUE
C-----
C Impressao para verificacao
C-----
WRITE(04,170)
SAREA=0.0D0
DO 160 J=1,NYR
DO 160 I=1,NXR

```

```

        AUX1=RW*DEXP((I-1)*DRO)
        AUX2=RW*DEXP(I*DRO)
        AUX3=DTETA(J)*(AUX2*AUX2-AUX1*AUX1)/2.0D0
        GTETA=DTETA(J)*360.0D0/(2.0D0*PI)
        WRITE(04,180)I,J,AUX1,RB(I),AUX2,ZB(I,J,1),
*           GTETA,AUX3
        SAREA=SAREA+AUX3
160 CONTINUE
170 FORMAT(//,'          Verificacao dos resultados da Sub GRID',/,
* '    GRID RADIAL',//,
* /,' BLOCO',T13,'Ri-1/2   Ri   Ri+1/2   Cota   Angulo   area',
* /,' I   J   K',T16,'m',T25,'m',T34,'m',T41,'m   graus   m2')
180 FORMAT(2(I2,1X),6(F8.2,1X))
    WRITE(04,181)SAREA
181 FORMAT(T32,'Area total =',F8.2)
    WRITE(04,190)
190 FORMAT(//,' BLOCO           FFX           FFY           FFZ')
    DO 200 K=1,NZR+1
        DO 200 I=1,NXR+1
            DO 200 J=1,NYR
                WRITE(04,210)I,J,K,FFX(I,J,K,1),FFY(I,J,K,1),FFZ(I,J,K,1)
200 CONTINUE
210 FORMAT(3(I2,1X),3(D14.4,1X))
C-----
C
C === GRID RETANGULAR === calculo das cotas e fatores de forma
C-----
C Calculo das cotas
C-----
        IF(OPT.EQ.'S') THEN
            DO 221 I=1,NXC
                DO 221 J=1,NYC
                    ZB(I,J,2)=ZC
221 CONTINUE
        ELSE
            DO 220 I=1,NXC
                ZB(I,JP,2)=ZC
C-----
C Calculo das cotas dos blocos abaixo do poco
C-----
                SUM=DBDY
                DO 230 J=(JP-1),1,-1
                    ZB(I,J,2)=ZC-SUM
                    SUM=SUM+DBDY
230 CONTINUE
C-----
C Calculo das cotas dos blocos acima do poco
C-----
                SUM=DBDY
                DO 240 J=(JP+1),NYC
                    ZB(I,J,2)=ZC+SUM
                    SUM=SUM+DBDY
240 CONTINUE
220 CONTINUE
        ENDIF
C-----
C Calculo dos fatores geometricos FFX para i=2,NX ; j=1,NY
C-----
        DO 270 I=2,NXC
            DO 270 J=1,NYC
                DO 270 K=1,NZC
                    IF(K.GE.KP.AND.K.LT.KP+NBP)THEN
                        IF(I.EQ.IP.AND.J.EQ.JP.OR.I.EQ.IP+1.AND.J.EQ.JP)GO TO 270
                    ENDIF

```

```

      AUX=2.0D0*DBDY*DBDZ*PERX(I-1,J,K,2)*PERX(I,J,K,2)/DBDX
      FFX(I,J,K,2)=AUX/(PERX(I-1,J,K,2)+PERX(I,J,K,2))

```

```

270  CONTINUE

```

```

C-----
C  Calculo dos fatores geometricos FFZ para k=2,NX ;j=1,NY
C-----

```

```

      DO 280 I=1,NXC
        DO 280 J=1,NYC
          DO 280 K=2,NZC
            IF(I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.K.LE.KP+NBP)GO TO 280
            AUX=2.0D0*DBDX*DBDY*PERZ(I,J,K-1,2)*PERZ(I,J,K,2)/DBDZ
            FFZ(I,J,K,2)=AUX/(PERZ(I,J,K-1,2)+PERZ(I,J,K,2))

```

```

280  CONTINUE

```

```

C-----
C  Calculo dos fatores geometricos FFY
C-----

```

```

      DO 290 J=2,NYC
        DO 290 I=1,NXC
          DO 290 K=1,NZC
            IF(K.GE.KP.AND.K.LT.KP+NBP)THEN
              IF(I.EQ.IP.AND.J.EQ.JP.OR.I.EQ.IP.AND.J.EQ.JP+1)GO TO 290
            ENDIF
            AUX=2.0D0*DBDX*DBDZ*PERY(I,J-1,K,2)*PERY(I,J,K,2)/DBDY
            FFY(I,J,K,2)=AUX/(PERY(I,J-1,K,2)+PERY(I,J,K,2))

```

```

290  CONTINUE

```

```

C-----
C  Impressao para verificacao
C-----

```

```

      WRITE(04,300)
300  FORMAT(//,' GRID RETANGULAR',
*//,' BLOCO      FFX              FFY              FFZ')
      DO 310 K=1,NZC
        DO 310 I=1,NXC
          DO 310 J=1,NYC
            WRITE(04,320)I,J,K,FFX(I,J,K,2),FFY(I,J,K,2),FFZ(I,J,K,2)
310  CONTINUE
320  FORMAT(3(I2,1X),3(D14.4,1X))
      DO 250 I=1,NXC
        DO 250 J=1,NYC
          WRITE(4,260)I,J,ZB(I,J,2)
260  FORMAT('I=',I2,'J=',I2,' COTA=',F20.12)
250  CONTINUE
C-----
      RETURN
      END

```

```

C-----
C  SUBROT. PARA CALC. DAS PRESSOES E SATURACOES DE AGUA NOS GRIDS POINTS
C-----

```

```

      SUBROUTINE PRESS0(PWF)

```

```

C-----
C  IMPLICIT DOUBLE PRECISION(A-H,O-Z)
C  PARAMETER(NLAB=18,NXM=08,NYM=05,NZM=04)
C  COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*      NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
C  COMMON/RES/DGAS,CR,PRIN,TEMP,ZC,RW,RE,SWIN,PHI,PERX,PERY,
*      PERZ,ZCOW,INDPOR,INDPER
C  COMMON/MALHA/ZB,RB,VPOR,DRO,WX,WY,WZ,FFX,FFY,FFZ,DTETA
C  COMMON/TSA/POA,SOA,SGA,SWA
C  COMMON/LPETR/SWL,KRWL,KROL,SLL,PCOWL,NKR,NPC
C  DIMENSION PHI(NXM,NYM,NZM,2),PERX(NXM,NYM,NZM,2),
*      PERY(NXM,NYM,NZM,2),PERZ(NXM,NYM,NZM,2)

```

```

      DIMENSION ZB(0:NXM+1,NYM,2),RB(NXM+1),VPOR(NXM,NYM,NZM,2),
*          FFX(NXM+1,NYM,NZM,2),FFY(NXM+1,NYM,NZM,2),
*          FFZ(NXM+1,NYM,NZM+1,2),DTETA(NYM+1)
      DIMENSION POA(NXM,NYM,NZM,2),SOA(NXM,NYM,NZM,2),
*          SGA(NXM,NYM,NZM,2),SWA(NXM,NYM,NZM,2)
      DIMENSION SWL(NLAB),SLL(NLAB),PCOWL(NLAB)
      REAL*8      KRWL(NLAB),KROL(NLAB)
C-----
C Pressao de referencia no contato oleo/agua
C-----
      ZREF=ZCOW
      PREF=PRIN
C-----
C Det. das pressoes iniciais nos blocos I, J. Para as camadas K's se
C repetem
C-----
      DO 60 L=1,2
        LX=NXR
        LY=NYR
        LZ=NZR
        IF(L.EQ.2) THEN
          LX=NXC
          LY=NYC
          LZ=NZC
        ENDIF
      DO 10 I=1,LX
        DO 10 J=1,LY
          DO 10 K=1,LZ
            POA(I,J,K,L)=PR(ZB(I,J,L),PREF,ZREF,1)
            PWA=PR(ZB(I,J,L),PREF,ZREF,2)
            IF(ZB(I,J,L).LT.ZREF) THEN
              POA(I,J,K,L)=PWA
              DPCA=0.000
            ELSE
              DPCA=POA(I,J,K,L)-PWA
            ENDIF
C Calculo das saturacoes iniciais
            IF(DPCA.LE.0.000) SWA(I,J,K,L)=1.000
            IF(DPCA.GT.PCOWL(1)) SWA(I,J,K,L)=SWIN
            IF(DPCA.GT.0.000.AND.DPCA.LE.PCOWL(1))
*              CALL INTERPC(PCOWL,SLL,NPC,DPCA,SWA(I,J,K,L),AUX)
          10 CONTINUE
          PWF=POA(1,1,1,1)
C-----
C Impressao para verificacao
C-----
            IF(L.EQ.1) WRITE(04,70)
            IF(L.EQ.2) WRITE(04,80)
          70 FORMAT(//,' GRID RADIAL - regiao do poco')
          80 FORMAT(//,' GRID RETANGULAR - regiao do reservatorio')
            WRITE(04,30)
          30 FORMAT(//,' Impressao para verificacao da sub PRESSO',//,
*          ' Bloco Pressao (MPa) Saturacao de Agua')
            DO 40 K=1,LZ
              DO 40 J=1,LY
                DO 40 I=1,LX
                  WRITE(04,50)I,J,K,POA(I,J,K,L)/1.0D+6,SWA(I,J,K,L)
          50 FORMAT(1X,3(I2,1X),3X,F10.4,3X,F10.4)
          40 CONTINUE
          60 CONTINUE
            RETURN
            END

```

```

C-----
DOUBLE PRECISION FUNCTION PR(ZAUX,PREF,ZREF,IZ)
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
PARAMETER(NLAB=18)
COMMON/LPVT0/POL,BOL,VOL,DOL,RSOL,POLS,BOLS,VOLS,DOLS,NPO,NPOS
COMMON/LPVTW/PWL,BWL,VWL,DWL,RSWL,NPW
DIMENSION POL(NLAB),BOL(NLAB),VOL(NLAB),DOL(NLAB),RSOL(NLAB),
*       POLS(NLAB),BOLS(NLAB),VOLS(NLAB),DOLS(NLAB)
DIMENSION PWL(NLAB),BWL(NLAB),VWL(NLAB),DWL(NLAB),RSWL(NLAB)
C
IF(IZ.EQ.1) THEN
  IF(PREF.GE.POLS(1)) THEN
    CALL INTER(POLS,DOLS,NPOS,PREF,DORF,YDORF)
  ELSE
    CALL INTER(POL,DOL,NPO,PREF,DORF,YDORF)
  ENDIF
ELSE
  CALL INTER(PWL,DWL,NPW,PREF,DORF,YDORF)
ENDIF
PB=PREF+DORF*(ZREF-ZAUX)
PMED=(PB+PREF)/2.0D0
5 IF(IZ.EQ.1) THEN
  IF(PMED.GE.POLS(1)) THEN
    CALL INTER(POLS,DOLS,NPOS,PMED,DORF,YDORF)
  ELSE
    CALL INTER(POL,DOL,NPO,PMED,DORF,YDORF)
  ENDIF
ELSE
  CALL INTER(PWL,DWL,NPW,PREF,DORF,YDORF)
ENDIF
FR=2.0D0*(PMED-PREF)-DORF*(ZREF-ZAUX)
DFR=2.0D0-(ZREF-ZAUX)*YDORF
DPMED=-FR/DFR
PMED=PMED+DPMED
IF(DPMED.GT.1.0D-6) THEN
  GOTO 5
ELSE
  PR=2.0D0*PMED-PREF
ENDIF
C
RETURN
END

```

```

C-----
C
C CALCULO DOS VOLUMES ORIGINAIS DE OLEO, GAS E AGUA NO RESERVATORIO
C-----

```

```

C
SUBROUTINE RESERV
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
PARAMETER(NXM=08,NYM=05,NZM=04)
COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*       NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
COMMON/ACUM/QOAC,QGAC,QWAC,VOORR,VWORR,VOORC,VWORC,BNPR,BWPR,
*       BNPC,BWPC,PWFA
COMMON/MALHA/ZB,RB,VPOR,DRO,WX,WY,WZ,FFX,FFY,FFZ,DTETA
COMMON/TSA/POA,SOA,SGA,SWA
COMMON/PVTA/BOA,RSOA,BWA
DIMENSION POA(NXM,NYM,NZM,2),SOA(NXM,NYM,NZM,2),
*       SGA(NXM,NYM,NZM,2),SWA(NXM,NYM,NZM,2)
DIMENSION BOA(NXM,NYM,NZM,2),BWA(NXM,NYM,NZM,2)
DIMENSION ZB(0:NXM+1,NYM,2),RB(NXM+1),VPOR(NXM,NYM,NZM,2),
*       FFX(NXM+1,NYM,NZM,2),FFY(NXM+1,NYM,NZM,2),

```

```

*          FFZ(NXM+1,NYM,NZM+1,2),DTETA(NYM+1)
C
  VOORR=0.0D0
  VGORR=0.0D0
  VWORR=0.0D0
  VOORC=0.0D0
  VGORC=0.0D0
  VWORC=0.0D0
  CALL FPOA
  DO 40 L=1,2
    IF(L.EQ.1) THEN
C-----
C      GRID RADIAL
C-----
      WRITE(04,50)
50      FORMAT(/,'  GRID RADIAL',/, ' bloco          Vp')
      DO 6 I=1,NXR
        DO 6 J=1,NYR
          DO 6 K=1,NZR
            VOORR=VOORR+VPOR(I,J,K,L)*SOA(I,J,K,L)*BOA(I,J,K,L)
            VWORR=VWORR+VPOR(I,J,K,L)*SWA(I,J,K,L)*BWA(I,J,K,L)
            WRITE(04,30)I,J,K,VPOR(I,J,K,L)
30          FORMAT(3(I2,1X),1X,F14.2)
6          CONTINUE
        ELSE
C-----
C      GRID CARTESIANO
C-----
      WRITE(04,60)
60      FORMAT(/,'  GRID RETANGULAR',/, ' bloco          Vp')
      DO 5 I=1,NXC
        DO 5 J=1,NYC
          DO 5 K=1,NZC
            IF(I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.K.LT.KP+NBP)GO TO 5
            VOORC=VOORC+VPOR(I,J,K,L)*SOA(I,J,K,L)*BOA(I,J,K,L)
            VWORC=VWORC+VPOR(I,J,K,L)*SWA(I,J,K,L)*BWA(I,J,K,L)
            WRITE(04,30)I,J,K,VPOR(I,J,K,L)
5          CONTINUE
        ENDIF
40      CONTINUE
C-----
C Impressao para verificacao
C-----
      WRITE(04,10)VOORR,VWORR,VOORC,VWORC,(VOORR+VOORC),(VWORR+VWORC)
10      FORMAT(//,' Verificacao dos volumes originais',/,
*          Radial          Cartesiano          Total',/,
*          OIP(m3)  WIP(m3)  OIP(m3)  WIP(m3)  OIP(m3)  WIP(m3)',/,
*          *3X,6(D9.3,1X))
      RETURN
      END
C-----
C SUBROTINA DE CONTROLE DA EXECUCAO DE TESTE DE FORMACAO
C-----
C SUBROUTINE CTESTE(TESTE,NTS,DT,QOMAX,TEMPTF,NITER)
C
  IMPLICIT DOUBLE PRECISION(A-H,O-Z)
  PARAMETER(NLAB=18,NXM=08,NYM=05,NZM=04,PI=3.141592654D0)
  COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*          NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
  COMMON/TFR/QOTF,TPTF,TETF,DTTF,DTI,TFLUX,BETA,OPT
  COMMON/VAZAO/QO,QOT,QG,QGT,QW,QWT,DQOP,DQOS,
*          DQWP,DQWS,RWO,RGO,PWF,TEMPO

```

```

COMMON/PVT/BO,DBO,VO,DVO,DO,DDO,RSO,DRSO,BW,DBW,VW,DVW,DW,DDW,CW
COMMON/RES/DGAS,CR,PRIN,TEMP,ZC,RW,RE,SWIN,PHI,PERX,PERY,
*   PERZ,ZCOW,INDPOR,INDPER
DIMENSION QO(NYM,NZM),QG(NYM,NZM),QW(NYM,NZM),DQOP(NYM,NZM),
*   DQOS(NYM,NZM),DQWP(NYM,NZM),DQWS(NYM,NZM)
DIMENSION BO(NXM,NYM,NZM,2),DBO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   VO(NXM,NYM,NZM,2),DVO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   DO(NXM,NYM,NZM,2),DDO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   BW(NXM,NYM,NZM,2),DBW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   VW(NXM,NYM,NZM,2),DVW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   DW(NXM,NYM,NZM,2),DDW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*   CW(NXM,NYM,NZM,2)
DIMENSION PHI(NXM,NYM,NZM,2),PERX(NXM,NYM,NZM,2),
*   PERY(NXM,NYM,NZM,2),PERZ(NXM,NYM,NZM,2)
LOGICAL TESTE
CHARACTER*1 OPT
C
  C1=3600.0DO
  C2=1.0D+6
  C3=1.0D+3
C Início do processo, passa-se a usar vazao e DT do teste.
  IF(OPT.EQ.'I') THEN
    QOT=QOTF

    DT=DTTF
    OPT='F'
    WRITE(12,5)
5    FORMAT(2X,'Tempo(h)  dsqrt(t)    pwf(MPa)  pi-pwf(KPa)',/,2X,
  *   'Tempo(h) (t+dt)/dt  pws(MPa)  pi-pws(KPa)')
    RETURN
  ENDIF
C Fase de fluxo, ao seu final vazao de oleo eh nula.
  IF(OPT.EQ.'F') THEN
    IF(TEMPO/C1.LT.0.0001) DTITF=0.00001
    IF(TEMPO/C1.GE.0.0001.AND.TEMPO/C1.LT.0.001) DTITF=0.0001
    IF(TEMPO/C1.GE.0.001.AND.TEMPO/C1.LT.0.01) DTITF=0.001
    IF(TEMPO/C1.GE.0.01.AND.TEMPO/C1.LT.0.1) DTITF=0.01
    IF(TEMPO/C1.GE.0.1.AND.TEMPO/C1.LT.1.0) DTITF=0.1
    IF(TEMPO/C1.GE.1.0.AND.TEMPO/C1.LT.10.0) DTITF=1.0
    IF(TEMPO/C1.GE.10.0.AND.TEMPO/C1.LT.100.0) DTITF=10.0
    IF(TEMPO/C1.GE.100.0.AND.TEMPO/C1.LT.1000.0) DTITF=100.0
    IF(TEMPO/C1.GE.1000.0.AND.TEMPO/C1.LT.10000.0) DTITF=1000.0
    DTITF=DTITF*C1
CC   IF(TEMPO.GE.TEMPTF) THEN
      DELTP=PRIN-PWF
      WRITE(12,10)TEMPO/C1,DSQRT(TEMPO/C1),PWF/C2,DELTP/C3
10   FORMAT(4(E10.5,2X))
      TEMPTF=TEMPTF+DTITF
CC   ENDIF
CC   IF(NITER.LE.2) DT=1.5DO*DT
CC   IF(DT.GE.DTITF) DT=DTITF
C
  IF(TEMPO.GE.TPTF) THEN
    CALL IMPRET(DT,OPT,L)
    QOT=0.0DO
    RWO=0.0DO
    TFLUX=TEMPO
    DT=DT/12.0DO
    OPT='E'
    RETURN
  ENDIF
ENDIF
C Fase de estatica, ao seu final verifica-se se a simul. vai prosseguir.

```

```

IF(OPT.EQ.'E') THEN
  HORN=TEMPO/(TEMPO-TFLUX)
  DELTP=PRIN-PWF
  WRITE(12,11)TEMPO/C1,HORN,PWF/C2,DELTP/C3
11  FORMAT(4(E12.6,1X))
  IF(TEMPO.GE.(TPTF+24.0D0*DT)) DT=DTTF
  IF(TEMPO.GE.(TETF+TPTF)) THEN
    CALL IMPRET(DT,OPT,L)
    PRINT*,'FINAL DO TESTE DE FORMACAO'
    PRINT*,'DESEJA PROSSEGUIR COM A SIMULACAO S ou N ?'
    READ(*,15)OPT
15  FORMAT(A1)
    IF(OPT.NE.'S') THEN
      PRINT*,'FINAL DO PROGRAMA, APENAS REALIZADO TFR'
      STOP
    ELSE
      TEMPO=0.0D0
      NTS=0
      DT=DTI
      QOT=QOMAX
      TESTE=.FALSE.
    ENDIF
  ENDIF
ENDIF
C
RETURN
END
C-----
C
C CALCULO DAS PROPRIEDADES DOS FLUIDOS DEPENDENTES DAS PRESSOES NA
C ITERACAO ANTERIOR (passo de tempo n)
C
C a partir desta sub nao existe po(ip,jp,kp,1) e sw(ip,jp,kp,1)
C-----
C
SUBROUTINE FPOA
C
  IMPLICIT DOUBLE PRECISION(A-H,O-Z)
  PARAMETER(NLAB=18,NXM=08,NYM=05,NZM=04)
  COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
  *      NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
  COMMON/LPVT0/POL,BOL,VOL,DOL,RSOL,POLS,BOLS,VOLS,DOLS,NPO,NPOS
  COMMON/LPVTW/PWL,BWL,VWL,DWL,RSWL,NPW
  COMMON/TSA/POA,SOA,SGA,SWA
  COMMON/PVTA/BOA,RSOA,BWA
  DIMENSION POL(NLAB),BOL(NLAB),VOL(NLAB),DOL(NLAB),RSOL(NLAB),
  *      POLS(NLAB),BOLS(NLAB),VOLS(NLAB),DOLS(NLAB)
  DIMENSION PWL(NLAB),BWL(NLAB),VWL(NLAB),DWL(NLAB),RSWL(NLAB)
  DIMENSION POA(NXM,NYM,NZM,2),SOA(NXM,NYM,NZM,2),
  *      SGA(NXM,NYM,NZM,2),SWA(NXM,NYM,NZM,2)
  DIMENSION BOA(NXM,NYM,NZM,2),BWA(NXM,NYM,NZM,2)
C-----
C Funcoes de PDA
C-----
  LX=NXR
  LY=NYR
  LZ=NZR
  DO 10 L=1,2
  IF(L.EQ.2) THEN
    LX=NXC
    LY=NYC
    LZ=NZC
  ENDIF
  DO 5 I=1,LX
    DO 5 J=1,LY
      DO 5 K=1,LZ

```

```

      IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP.AND.
*      K.GE.KP.AND.K.LT.KP+NBP) GO TO 5
      IF(POA(I,J,K,L).GE.POLS(1)) THEN
        CALL INTER(POLS,BOLS,NPOS,POA(I,J,K,L),BOA(I,J,K,L),AUX)
      ELSE
        CALL INTER(POL,BOL,NPO,POA(I,J,K,L),BOA(I,J,K,L),AUX)
      ENDIF
      CALL INTER(PWL,BWL,NPW,POA(I,J,K,L),BWA(I,J,K,L),AUX)
5    CONTINUE
10   CONTINUE
C    RETURN
    END
-----
C
C  CALCULO DAS PROPRIEDADES DOS FLUIDOS DEPENDENTES DAS PRESSOES NA
C  ITERACAO PRESENTE E DAS PRESSOES NA INTERFACES
C
C  -----
C  SUBROUTINE FPPOM(PWF,WX,WY,WZ)
C
C  IMPLICIT DOUBLE PRECISION(A-H,O-Z)
C  PARAMETER(NLAB=18,NXM=08,NYM=05,NZM=04)
C  COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*  NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
C  COMMON/LPVT0/POL,BOL,VOL,DOL,RSOL,POLS,BOLS,VOLS,DOLS,NPO,NPOS
C  COMMON/LPVTW/PWL,BWL,VWL,DWL,RSWL,NPW
C  COMMON/LPVTG/PGL,BGL,VGL,DGL,NPG
C  COMMON/TS/PO,SO,SG,SW
C  COMMON/PVT/BO,DBO,VO,DVO,DO,DDO,RSO,DRSO,BW,DBW,VW,DVW,DW,DDW,CW
C  COMMON/IFACEX/BOMX,VOMX,DOMX,BWMX,VWMX,DWMX
C  COMMON/IFACEY/BOMY,VOMY,DOMY,BWMY,VWMY,DWMY
C  COMMON/IFACEZ/BOMZ,VOMZ,DOMZ,BWMZ,VWMZ,DWMZ
C  DIMENSION POL(NLAB),BOL(NLAB),VOL(NLAB),DOL(NLAB),RSOL(NLAB),
*  POLS(NLAB),BOLS(NLAB),VOLS(NLAB),DOLS(NLAB)
C  DIMENSION PWL(NLAB),BWL(NLAB),VWL(NLAB),DWL(NLAB),RSWL(NLAB)
C  DIMENSION PGL(NLAB),BGL(NLAB),VGL(NLAB),DGL(NLAB)
C  DIMENSION PO(0:NXM+1,0:NYM+1,0:NZM+1,2),SO(NXM,NYM,NZM,2),
*  SG(NXM,NYM,NZM,2),SW(NXM,NYM,NZM,2)
C  DIMENSION BO(NXM,NYM,NZM,2),DBO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*  VO(NXM,NYM,NZM,2),DVO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*  DO(NXM,NYM,NZM,2),DDO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*  BW(NXM,NYM,NZM,2),DBW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*  VW(NXM,NYM,NZM,2),DVW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*  DW(NXM,NYM,NZM,2),DDW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*  CW(NXM,NYM,NZM,2)
C  DIMENSION BOMX(0:NXM+1,NYM,NZM,2),VOMX(0:NXM+1,NYM,NZM,2),
*  DOMX(0:NXM+1,NYM,NZM,2),BWMX(0:NXM+1,NYM,NZM,2),
*  VWMX(0:NXM+1,NYM,NZM,2),DWMX(0:NXM+1,NYM,NZM,2)
C  DIMENSION BOMY(NXM,NYM+1,NZM,2),VOMY(NXM,NYM+1,NZM,2),
*  DOMY(NXM,NYM+1,NZM,2),BWMY(NXM,NYM+1,NZM,2),
*  VWMY(NXM,NYM+1,NZM,2),DWMY(NXM,NYM+1,NZM,2)
C  DIMENSION BOMZ(NXM,NYM,NZM+1,2),VOMZ(NXM,NYM,NZM+1,2),
*  DOMZ(NXM,NYM,NZM+1,2),BWMZ(NXM,NYM,NZM+1,2),
*  VWMZ(NXM,NYM,NZM+1,2),DWMZ(NXM,NYM,NZM+1,2)
C
C  -----
C  Funcoes de PO (pressao no passo de tempo n+1)
C
C  -----
      NX=NXR
      NY=NYR
      NZ=NZR

```

```

DO 4 L=1,2
  IF(L.EQ.2) THEN
    NX=NXC
    NY=NYC
    NZ=NZC
  ENDIF
DO 5 I=1,NX
DO 5 J=1,NY
DO 5 K=1,NZ
  IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP.AND.
*      K.GE.KP.AND.K.LT.KP+NBP) GO TO 5
  IF(PO(I,J,K,L).GT.POLS(1)) THEN
    CALL INTER(POLS,BOLS,NPOS,PO(I,J,K,L),BO(I,J,K,L),DBO(I,J,K,L))
    CALL INTER(POLS,VOLS,NPOS,PO(I,J,K,L),VO(I,J,K,L),DVO(I,J,K,L))
    CALL INTER(POLS,DOLS,NPOS,PO(I,J,K,L),DO(I,J,K,L),DDO(I,J,K,L))
  ELSE
    CALL INTER(POL,BOL,NPO,PO(I,J,K,L),BO(I,J,K,L),DBO(I,J,K,L))
    CALL INTER(POL,VOL,NPO,PO(I,J,K,L),VO(I,J,K,L),DVO(I,J,K,L))
    CALL INTER(POL,DOL,NPO,PO(I,J,K,L),DO(I,J,K,L),DDO(I,J,K,L))
  ENDIF
  CALL INTER(PWL,BWL,NPW,PO(I,J,K,L),BW(I,J,K,L),DBW(I,J,K,L))
  CALL INTER(PWL,VWL,NPW,PO(I,J,K,L),VW(I,J,K,L),DVW(I,J,K,L))
  CALL INTER(PWL,DWL,NPW,PO(I,J,K,L),DW(I,J,K,L),DDW(I,J,K,L))
  CW(I,J,K,L)=DDW(I,J,K,L)/DW(I,J,K,L)
5  CONTINUE
C-----
C Funcoes PVT das pr. nas interfaces X,Y e Z internas (usam sub. saux.)
C Interface (1,j,k) - face do poco GRID RADIAL
C-----
  IF(L.EQ.1) THEN
    DO 10 J=1,NY
      DO 10 K=1,NZ
        CALL SAUX(PWF,BOMX(1,J,K,L),VOMX(1,J,K,L),DOMX(1,J,K,L),
*              BWMX(1,J,K,L),VWMX(1,J,K,L),DWMX(1,J,K,L))
10  CONTINUE
  ENDIF
C-----
C Interfaces X (apenas as interfaces internas da malha)
C Na face externa,I=NX+1, e I=0 as funcoes sao iguais a ZERO
C-----
C cartesiano
C
  IF(L.EQ.2) THEN
    W=0.5DO
    DO 20 I=2,NX
      DO 20 J=1,NY
        DO 20 K=1,NZ
          IF(K.GE.KP.AND.K.LT.KP+NBP) THEN
            IF(I.EQ.IP.AND.J.EQ.JP.OR.I.EQ.IP+1.AND.J.EQ.JP) GO TO 20
          ENDIF
          POM=(1.ODO-W)*PO(I-1,J,K,L)+W*PO(I,J,K,L)
          CALL SAUX(POM,BOMX(I,J,K,L),VOMX(I,J,K,L),DOMX(I,J,K,L),
*                BWMX(I,J,K,L),VWMX(I,J,K,L),DWMX(I,J,K,L))
20  CONTINUE
  ENDIF
C
C radial
C
  IF(L.EQ.1) THEN
    DO 21 I=2,NX
      DO 21 J=1,NY
        DO 21 K=1,NZ

```

```

IF(I.EQ.NX) THEN
  IF(J.LE.NQL) THEN
    IF(IP.EQ.NXC) GO TO 21
    IX=IP+1
    JX=JP
  ENDIF
  IF(J.GT.NQL.AND.J.LE.NQL+NQS) THEN
    IF(JP.EQ.NYC) GO TO 21
    IX=IP
    JX=JP+1
  ENDIF
  IF(J.GT.NQL+NQS.AND.J.LE.NQS+2*NQL) THEN
    IF(IP.EQ.1) GO TO 21
    IX=IP-1
    JX=JP
  ENDIF
  IF(J.GT.NQS+2*NQL) THEN
    IF(JP.EQ.1) GO TO 21
    IX=IP
    JX=JP-1
  ENDIF
  POM=(1.0DO-WRC)*PO(NX,J,K,1)+WRC*PO(IX,JX,KP+K-1,2)
  CALL SAUX(POM,BOMX(I+1,J,K,1),VOMX(I+1,J,K,1),
    *          DOMX(I+1,J,K,1),BWMX(I+1,J,K,1),
    *          VWMX(I+1,J,K,1),DWMX(I+1,J,K,1))
  ENDIF
  POM=(1.0DO-WX)*PO(I-1,J,K,1)+WX*PO(I,J,K,L)
  CALL SAUX(POM,BOMX(I,J,K,L),VOMX(I,J,K,L),DOMX(I,J,K,L),
    *          BWMX(I,J,K,L),VWMX(I,J,K,L),DWMX(I,J,K,L))
21  * CONTINUE
    ENDIF
C-----
C Interfaces Z (apenas as interfaces internas da malha)
C-----
    DO 40 I=1,NX
      DO 40 J=1,NY
        DO 40 K=2,NZ
          IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP) THEN
            IF(K.GE.KP.AND.K.LT.KP+NBP) GO TO 40
          ENDIF
          POM=(1.0DO-WZ)*PO(I,J,K-1,L)+WZ*PO(I,J,K,L)
          CALL SAUX(POM,BOMZ(I,J,K,L),VOMZ(I,J,K,L),DOMZ(I,J,K,L),
            *          BWMZ(I,J,K,L),VWMZ(I,J,K,L),DWMZ(I,J,K,L))
40  * CONTINUE
C-----
C Interfaces Z (primeira e ultima face para o grid radial)
C-----
    IF(L.EQ.1) THEN
      IF(KP.NE.1) THEN
        DO 41 I=1,NX
          DO 41 J=1,NY
            POM=(1.0DO-WZ)*PO(IP,JP,KP-1,2)+WZ*PO(I,J,1,1)
            CALL SAUX(POM,BOMZ(I,J,1,1),VOMZ(I,J,1,1),DOMZ(I,J,1,1),
              *          BWMZ(I,J,1,1),VWMZ(I,J,1,1),DWMZ(I,J,1,1))
41  * CONTINUE
        ENDIF
C
        IF(KP+NBP-1.LT.NZC) THEN
          DO 42 I=1,NX
            DO 42 J=1,NY
              POM=(1.0DO-WZ)*PO(I,J,NZR,1)+WZ*PO(IP,JP,KP+NBP,2)
              CALL SAUX(POM,BOMZ(I,J,NZR+1,1),VOMZ(I,J,NZR+1,1),
                *          DOMZ(I,J,NZR+1,1),BWMZ(I,J,NZR+1,1),

```

```

*          VWMZ(I,J,NZR+1,1),DWMZ(I,J,NZR+1,1))
42  CONTINUE
    ENDIF
    ENDIF
C-----
C Interfaces Y (apenas as interfaces internas da malha)
C-----
    LINF=1
    LSUP=NY
    IF(L.EQ.2) THEN
        LINF=2
    ENDIF
    DO 50 I=1,NX
        DO 50 J=LINF,LSUP
            DO 50 K=1,NZ
                IF(L.EQ.2.AND.K.GE.KP.AND.K.LT.KP+NBP) THEN
                    IF(I.EQ.IP.AND.J.EQ.JP.OR.
*          I.EQ.IP.AND.J.EQ.JP+1) GO TO 50
                ENDIF
                AUXP=PO(I,J-1,K,L)
                IF(L.EQ.1.AND.J.EQ.1) AUXP=PO(I,NY,K,L)
                POM=(1.ODO-WY)*AUXP+WY*PO(I,J,K,L)
                CALL SAUX(POM,BOMY(I,J,K,L),VOMY(I,J,K,L),DOMY(I,J,K,L),
*          BWMY(I,J,K,L),VWMY(I,J,K,L),DWMY(I,J,K,L))
50  CONTINUE
4  CONTINUE
C
    RETURN
    END
C-----
C
C Subrot. auxiliar para calculo das funcoes das pressoes nas interfaces
C-----
    SUBROUTINE SAUX(POM,BOM,VOM,DOM,BWM,VWM,DWM)
    IMPLICIT DOUBLE PRECISION(A-H,O-Z)
    PARAMETER(NLAB=18)
    COMMON/LPVT0/POL,BOL,VOL,DOL,RSOL,POLS,BOLS,VOLS,DOLS,NPO,NPOS
    COMMON/LPVTW/PWL,BWL,VWL,DWL,RSWL,NPW
    DIMENSION POL(NLAB),BOL(NLAB),VOL(NLAB),DOL(NLAB),RSOL(NLAB),
*          POLS(NLAB),BOLS(NLAB),VOLS(NLAB),DOLS(NLAB)
    DIMENSION PWL(NLAB),BWL(NLAB),VWL(NLAB),DWL(NLAB),RSWL(NLAB)
C
    IF(POM.GT.POLS(1)) THEN
        CALL INTER(POLS,BOLS,NPOS,POM,BOM,AUX)
        CALL INTER(POLS,VOLS,NPOS,POM,VOM,AUX)
        CALL INTER(POLS,DOLS,NPOS,POM,DOM,AUX)
    ELSE
        CALL INTER(POL,BOL,NPO,POM,BOM,AUX)
        CALL INTER(POL,VOL,NPO,POM,VOM,AUX)
        CALL INTER(POL,DOL,NPO,POM,DOM,AUX)
    ENDIF
    CALL INTER(PWL,BWL,NPW,POM,BWM,AUX)
    CALL INTER(PWL,VWL,NPW,POM,VWM,AUX)
    CALL INTER(PWL,DWL,NPW,POM,DWM,AUX)
C
    RETURN
    END
C-----
C
C SUBROTINA DE VERIFICACAO DO SENTIDO DE FLUXO ENTRE AS CELULAS,
C CALCULO DAS TRANSMISSIBILIDADES AO OLEO E AO GAS NAS INTERFACES DAS
C CELULAS PELO ESQUEMA "UM PONTO A MONTANTE" E DE SUAS DERIVADAS dt/dsw
C

```

SUBROUTINE TRANS

IMPLICIT DOUBLE PRECISION(A-H,O-Z)

PARAMETER(NLAB=18,NXM=08,NYM=05,NZM=04)

COMMON/SIM/DT,DTMAX,DTMIN,TOLP,TOLS,DSLIM,DPLIM,TSIM,
* QOMAX,QOMIN,PWMIN,BSOR,JPROD,MSOLU,ITMAX,ITMIN

COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
* NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP

COMMON/RES/DGAS,CR,PRIN,TEMP,ZC,RW,RE,SWIN,PHI,PERX,PERY,
* PERZ,ZCOW,INDPOR,INDPER

COMMON/LPETR/SWL,KRWL,KROL,SLL,PCOWL,NKR,NPC

COMMON/MALHA/ZB,RB,VPOR,DRO,WX,WY,WZ,FFX,FFY,FFZ,DTETA

COMMON/TS/PO,SO,SG,SW

COMMON/IFACEX/BOMX,VOMX,DOMX,BWMX,VWMX,DWMX

COMMON/IFACEY/BOMY,VOMY,DOMY,BWY,VWY,DWY

COMMON/IFACEZ/BOMZ,VOMZ,DOMZ,BWMZ,VWMZ,DWMZ

COMMON/PETR/KRO,KRW,DKRO,DKRW,PC,DPC

COMMON/DTRANS/TOX,TWX,TOY,TWY,TOZ,TWZ,

* TOX1SA,TOX1SI,TWX1SA,TWX1SI,

* TOX2SI,TOX2SP,TWX2SI,TWX2SP,

* TOY1SA,TOY1SI,TWY1SA,TWY1SI,

* TOY2SI,TOY2SP,TWY2SI,TWY2SP,

* TOZ1SA,TOZ1SI,TWZ1SA,TWZ1SI,

* TOZ2SI,TOZ2SP,TWZ2SI,TWZ2SP

COMMON/TF/TFLUXO,TFLUXW,SUMTO,SUMTW,AFXO,AFXW

DIMENSION SWL(NLAB),SLL(NLAB),PCOWL(NLAB)

REAL*8 KRWL(NLAB),KROL(NLAB)

DIMENSION ZB(0:NXM+1,NYM,2),RB(NXM+1),VPOR(NXM,NYM,NZM,2),

* FFX(NXM+1,NYM,NZM,2),FFY(NXM+1,NYM,NZM,2),

* FFZ(NXM+1,NYM,NZM+1,2),DTETA(NYM+1)

DIMENSION PQ(0:NXM+1,0:NYM+1,0:NZM+1,2),SO(NXM,NYM,NZM,2),

* SG(NXM,NYM,NZM,2),SW(NXM,NYM,NZM,2)

DIMENSION PHI(NXM,NYM,NZM,2),PERX(NXM,NYM,NZM,2),

* PERY(NXM,NYM,NZM,2),PERZ(NXM,NYM,NZM,2)

DIMENSION BOMX(0:NXM+1,NYM,NZM,2),VOMX(0:NXM+1,NYM,NZM,2),

* DOMX(0:NXM+1,NYM,NZM,2),BWMX(0:NXM+1,NYM,NZM,2),

* VWMX(0:NXM+1,NYM,NZM,2),DWMX(0:NXM+1,NYM,NZM,2)

DIMENSION BOMY(NXM,NYM+1,NZM,2),VOMY(NXM,NYM+1,NZM,2),

* DOMY(NXM,NYM+1,NZM,2),BWY(NXM,NYM+1,NZM,2),

* VWY(NXM,NYM+1,NZM,2),DWY(NXM,NYM+1,NZM,2)

DIMENSION BOMZ(NXM,NYM,NZM+1,2),VOMZ(NXM,NYM,NZM+1,2),

* DOMZ(NXM,NYM,NZM+1,2),BWMZ(NXM,NYM,NZM+1,2),

* VWMZ(NXM,NYM,NZM+1,2),DWMZ(NXM,NYM,NZM+1,2)

DIMENSION DKRO(NXM,NYM,NZM,2),DKRW(NXM,NYM,NZM,2),

* PC(0:NXM+1,0:NYM+1,0:NZM+1,2),DPC(0:NXM+1,0:NYM+1,0:NZM+1,2)

REAL*8 KRO(NXM,NYM,NZM,2),KRW(NXM,NYM,NZM,2)

DIMENSION TOX(NXM+1,NYM,NZM,2),TWX(NXM+1,NYM,NZM,2),

* TOX1SA(NXM,NYM,NZM,2),TOX1SI(NXM,NYM,NZM,2),

* TWX1SA(NXM,NYM,NZM,2),TWX1SI(NXM,NYM,NZM,2),

* TOX2SI(NXM,NYM,NZM,2),TOX2SP(NXM,NYM,NZM,2),

* TWX2SI(NXM,NYM,NZM,2),TWX2SP(NXM,NYM,NZM,2)

DIMENSION TOY(NXM,NYM+1,NZM,2),TWY(NXM,NYM+1,NZM,2),

* TOY1SA(NXM,NYM,NZM,2),TOY1SI(NXM,NYM,NZM,2),

* TWY1SA(NXM,NYM,NZM,2),TWY1SI(NXM,NYM,NZM,2),

* TOY2SI(NXM,NYM,NZM,2),TOY2SP(NXM,NYM,NZM,2),

* TWY2SI(NXM,NYM,NZM,2),TWY2SP(NXM,NYM,NZM,2)

```

      DIMENSION TOZ(NXM,NYM,NZM+1,2),TWZ(NXM,NYM,NZM+1,2),
*          TOZ1SA(NXM,NYM,NZM,2),TOZ1SI(NXM,NYM,NZM,2),
*          TWZ1SA(NXM,NYM,NZM,2),TWZ1SI(NXM,NYM,NZM,2),
*          TOZ2SI(NXM,NYM,NZM,2),TOZ2SP(NXM,NYM,NZM,2),
*          TWZ2SI(NXM,NYM,NZM,2),TWZ2SP(NXM,NYM,NZM,2)
      INTEGER DELTO,DELTW,DTO(NYM,NYM),DTW(NYM,NYM)
C-----
C Det. de perm. relat. ao oleo e a agua,pressoes capilares e derivadas
C-----
      DO 1 I=1,NXR
        DO 1 J=1,NYR
          DO 1 K=1,NZR
            CALL INTER(SWL,KROL,NKR,SW(I,J,K,1),KRO(I,J,K,1),DKRO(I,J,K,1))
            CALL INTER(SWL,KRWL,NKR,SW(I,J,K,1),KRW(I,J,K,1),DKRW(I,J,K,1))
            CALL INTER(SLL,PCOWL,NPC,SW(I,J,K,1),PC(I,J,K,1),DPC(I,J,K,1))
1      CONTINUE
C
      DO 2 I=1,NXC
        DO 2 J=1,NYC
          DO 2 K=1,NZC
            IF(I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.K.LT.KP+NBP) GO TO 2
            CALL INTER(SWL,KROL,NKR,SW(I,J,K,2),KRO(I,J,K,2),DKRO(I,J,K,2))
            CALL INTER(SWL,KRWL,NKR,SW(I,J,K,2),KRW(I,J,K,2),DKRW(I,J,K,2))
            CALL INTER(SLL,PCOWL,NPC,SW(I,J,K,2),PC(I,J,K,2),DPC(I,J,K,2))
2      CONTINUE
C-----
      NX=NXR
      NY=NYR
      NZ=NZR
      DO 1000 L=1,2
        IF(L.EQ.2) THEN
          NX=NXC
          NY=NYC
          NZ=NZC
        ENDIF
C-----
C Verificacao dos sentidos de fluxo para as fases oleo e agua
C para "head" positivo :delto e deltw =1 (fluxo para o poco)
C para "head" negativo :delto e deltw =0 (fluxo contra o poco)
C-----
C Blocos do poco - GRID RADIAL
C Transmissibilidades ao oleo e a agua no poco ,derivadas dT/dSw
C-----
      IF(L.EQ.1) THEN
        DO 40 J=1,NY
          DO 40 K=1,NZ
            TOX(1,J,K,L)=FFX(1,J,K,L)*BOMX(1,J,K,L)*KRO(1,J,K,L)/VOMX(1,J,K,L)
            TWX(1,J,K,L)=FFX(1,J,K,L)*BWMX(1,J,K,L)*KRW(1,J,K,L)/VWMX(1,J,K,L)
C
            TOX1SI(1,J,K,L)=FFX(1,J,K,L)*BOMX(1,J,K,L)*DKRO(1,J,K,L)/
            * VOMX(1,J,K,L)
            TWX1SI(1,J,K,L)=FFX(1,J,K,L)*BWMX(1,J,K,L)*DKRW(1,J,K,L)/
            * VWMX(1,J,K,L)
40      CONTINUE
      ENDIF
C-----
C Verificacao dos sentidos de fluxo para as fases oleo e agua
C para "head" positivo :delto e deltg =1 (fluxo para o poco) ou
C (da direita para a esquerda)
C para "head" negativo :delto e deltg =0 (fluxo contra o poco)
C (da esquerda para a direita)

```

C Blocos (1,j,k)

```

C-----
DO 50 J=1,NY
DO 50 K=1,NZ
  IF(L.EQ.2.AND.K.GE.KP.AND.K.LT.KP+NBP) THEN
    IF(IP.EQ.1.AND.J.EQ.JP.OR.
      * IP.EQ.2.AND.J.EQ.JP) GO TO 50
    ENDIF
    HEAD=PO(2,J,K,L)-PO(1,J,K,L)+DOMX(2,J,K,L)*(ZB(2,J,L)-ZB(1,J,L))
    DELTO=1
    IF(HEAD.LT.0.0D0) DELTO=0
    HEAD=PO(2,J,K,L)-PO(1,J,K,L)-PC(2,J,K,L)+PC(1,J,K,L)+
      * DWMX(2,J,K,L)*(ZB(2,J,L)-ZB(1,J,L))
    DELTW=1
    IF(HEAD.LT.0.0D0) DELTW=0
    TOX2SI(1,J,K,L)=(1-DELTO)*FFX(2,J,K,L)*BOMX(2,J,K,L)*
      * DKRO(1,J,K,L)/VOMX(2,J,K,L)
    TOX2SP(1,J,K,L)=DELTO*FFX(2,J,K,L)*BOMX(2,J,K,L)*
      * DKRO(2,J,K,L)/VOMX(2,J,K,L)
    TWX2SI(1,J,K,L)=(1-DELTW)*FFX(2,J,K,L)*BWMX(2,J,K,L)*
      * DKRW(1,J,K,L)/VWMX(2,J,K,L)
    TWX2SP(1,J,K,L)=DELTW*FFX(2,J,K,L)*BWMX(2,J,K,L)*
      * DKRW(2,J,K,L)/VWMX(2,J,K,L)
    DTO(J,K)=DELTO
    DTW(J,K)=DELTW
50 CONTINUE

```

C INTERFACES X:Transmissib. ao oleo e a agua e suas derivadas dT/dSw

```

C-----
DO 10 K=1,NZ
DO 10 J=1,NY
DO 10 I=2,NX-1
  IF(I.EQ.2) THEN
    DELTO=DTO(J,K)
    DELTW=DTW(J,K)
  ENDIF
  IF(L.EQ.2.AND.K.GE.KP.AND.K.LT.KP+NBP) THEN
    IF(I.EQ.IP.AND.J.EQ.JP) GO TO 10
    IF(I.EQ.IP+1.AND.J.EQ.JP)GO TO 11
  ENDIF
  TOX(I,J,K,L)=DELTO*FFX(I,J,K,L)*BOMX(I,J,K,L)*KRO(I,J,K,L)/
    * VOMX(I,J,K,L)+(1-DELTO)*FFX(I,J,K,L)*
    * BOMX(I,J,K,L)*KRO(I-1,J,K,L)/VOMX(I,J,K,L)
  TWX(I,J,K,L)=DELTW*FFX(I,J,K,L)*BWMX(I,J,K,L)*KRW(I,J,K,L)/
    * VWMX(I,J,K,L)+(1-DELTW)*FFX(I,J,K,L)*
    * BWMX(I,J,K,L)*KRW(I-1,J,K,L)/VWMX(I,J,K,L)
C
  TOX1SA(I,J,K,L)=(1-DELTO)*FFX(I,J,K,L)*BOMX(I,J,K,L)*
    * DKRO(I-1,J,K,L)/VOMX(I,J,K,L)
  TOX1SI(I,J,K,L)=DELTO*FFX(I,J,K,L)*BOMX(I,J,K,L)*DKRO(I,J,K,L)/
    * VOMX(I,J,K,L)
  TWX1SA(I,J,K,L)=(1-DELTW)*FFX(I,J,K,L)*BWMX(I,J,K,L)*
    * DKRW(I-1,J,K,L)/VWMX(I,J,K,L)
  TWX1SI(I,J,K,L)=DELTW*FFX(I,J,K,L)*BWMX(I,J,K,L)*
    * DKRW(I,J,K,L)/VWMX(I,J,K,L)
C Repete-se a verificacao do sentido de fluxo para a proxima interface X
  IF(L.EQ.2.AND.I.EQ.IP-1.AND.J.EQ.JP)GO TO 10
11 HEAD=PO(I+1,J,K,L)-PO(I,J,K,L)+DOMX(I+1,J,K,L)*
  * (ZB(I+1,J,L)-ZB(I,J,L))

```

```

      DELTO=1
      IF(HEAD.LT.0.0D0) DELTO=0
      HEAD=PO(I+1,J,K,L)-PO(I,J,K,L)-PC(I+1,J,K,L)+PC(I,J,K,L)+
*      DWMX(I+1,J,K,L)*(ZB(I+1,J,L)-ZB(I,J,L))
      DELTW=1
      IF(HEAD.LT.0.0D0) DELTW=0
      TOX2SI(I,J,K,L)=(1-DELTO)*FFX(I+1,J,K,L)*BOMX(I+1,J,K,L)*
*      DKRO(I,J,K,L)/VOMX(I+1,J,K,L)
      TOX2SP(I,J,K,L)=DELTO*FFX(I+1,J,K,L)*BOMX(I+1,J,K,L)*
*      DKRO(I+1,J,K,L)/VOMX(I+1,J,K,L)
      TWX2SI(I,J,K,L)=(1-DELTW)*FFX(I+1,J,K,L)*BWMX(I+1,J,K,L)*
*      DKRW(I,J,K,L)/VWMX(I+1,J,K,L)
      TWX2SP(I,J,K,L)=DELTW*FFX(I+1,J,K,L)*BWMX(I+1,J,K,L)*
*      DKRW(I+1,J,K,L)/VWMX(I+1,J,K,L)
C
C 10      CONTINUE
C-----
C INTERF.X:Transmissib. ao oleo e a agua e suas derivadas dT/dSw na
C      ULTIMA COLUMNA de celulas (no limite do reserv. To e Tw = 0)
C Repete-se a verificacao do sentido de fluxo nas interfaces X
C-----
      DO 20 J=1,NY
      DO 20 K=1,NZ
      IF(L.EQ.2.AND.K.GE.KP.AND.K.LT.KP+NBP) THEN
      IF(IP.EQ.NX-1.AND.J.EQ.JP.OR.
*      IP.EQ.NX.AND.J.EQ.JP) GO TO 20
      ENDIF
      HEAD=PO(NX,J,K,L)-PO(NX-1,J,K,L)+DOMX(NX,J,K,L)*
*      (ZB(NX,J,L)-ZB(NX-1,J,L))
      DELTO=1
      IF(HEAD.LT.0.0D0) DELTO=0
      HEAD=PO(NX,J,K,L)-PO(NX-1,J,K,L)-PC(NX,J,K,L)+PC(NX-1,J,K,L)+
*      DWMX(NX,J,K,L)*(ZB(NX,J,L)-ZB(NX-1,J,L))
      DELTW=1
      IF(HEAD.LT.0.0D0) DELTW=0
      TOX(NX,J,K,L)=DELTO*FFX(NX,J,K,L)*BOMX(NX,J,K,L)*KRO(NX,J,K,L)/
*      VOMX(NX,J,K,L)+
*      (1-DELTO)*FFX(NX,J,K,L)*BOMX(NX,J,K,L)*
*      KRO(NX-1,J,K,L)/VOMX(NX,J,K,L)
      TWX(NX,J,K,L)=DELTW*FFX(NX,J,K,L)*BWMX(NX,J,K,L)*KRW(NX,J,K,L)/
*      VWMX(NX,J,K,L)+
*      (1-DELTW)*FFX(NX,J,K,L)*BWMX(NX,J,K,L)*
*      KRW(NX-1,J,K,L)/VWMX(NX,J,K,L)
C
      TOX1SA(NX,J,K,L)=(1-DELTO)*FFX(NX,J,K,L)*BOMX(NX,J,K,L)*
*      DKRO(NX-1,J,K,L)/VOMX(NX,J,K,L)
      TOX1SI(NX,J,K,L)=DELTO*FFX(NX,J,K,L)*BOMX(NX,J,K,L)*
*      DKRO(NX,J,K,L)/VOMX(NX,J,K,L)
      TWX1SA(NX,J,K,L)=(1-DELTW)*FFX(NX,J,K,L)*BWMX(NX,J,K,L)*
*      DKRW(NX-1,J,K,L)/VWMX(NX,J,K,L)
      TWX1SI(NX,J,K,L)=DELTW*FFX(NX,J,K,L)*BWMX(NX,J,K,L)*
*      DKRW(NX,J,K,L)/VWMX(NX,J,K,L)
C transmissibilidade e derivadas na face externa
      IF(L.EQ.1) THEN
      IF(J.LE.NQL) THEN
      IF(IP.EQ.NXC) GO TO 20
      IX=IP+1
      JX=JP
      ENDIF
      IF(J.GT.NQL.AND.J.LE.NQL+NQS) THEN

```

```

      IF(JP.EQ.NYC) GO TO 20
      IX=IP

      JX=JP+1
    ENDIF
    IF(J.GT.NQL+NQS.AND.J.LE.NQS+2*NQL) THEN
      IF(IP.EQ.1) GO TO 20
      IX=IP-1
      JX=JP
    ENDIF
    IF(J.GT.NQS+2*NQL) THEN
      IF(JP.EQ.1) GO TO 20
      IX=IP
      JX=JP-1
    ENDIF
    AKROC=KRO(IX,JX,KP+K-1,2)
    AKRWC=KRW(IX,JX,KP+K-1,2)
    AUXBO=BOMX(NX+1,J,K,1)
    AUXBW=BWMX(NX+1,J,K,1)
    AUXDO=DOMX(NX+1,J,K,1)
    AUXDW=DWMX(NX+1,J,K,1)
    AUXVO=VOMX(NX+1,J,K,1)
    AUXVW=VWMX(NX+1,J,K,1)
    HEAD=PO(IX,JX,KP+K-1,2)-PO(NX,J,K,1)+AUXDO*
  *   (ZB(NX,J,1)-ZB(IX,JX,2))
    DELTO=1
    IF(HEAD.LT.0.0D0) DELTO=0
    HEAD=PO(IX,JX,KP+K-1,2)-PO(NX,J,K,1)-PC(IX,JX,KP+K-1,2)+
  *   PC(NX,J,K,1)+AUXDW*(ZB(NX,J,1)-ZB(IX,JX,2))
    DELTW=1
    IF(HEAD.LT.0.0D0) DELTW=0
    TOX(NX+1,J,K,L)=DELTO*FFX(NX+1,J,K,L)*AUXBO*AKROC/AUXVO+
  *   (1-DELTO)*FFX(NX+1,J,K,L)*AUXBO*KRO(NX,J,K,L)/AUXVO
    TWX(NX+1,J,K,L)=DELTW*FFX(NX+1,J,K,L)*AUXBW*AKRWC/AUXVW+
  *   (1-DELTW)*FFX(NX+1,J,K,L)*AUXBW*KRW(NX,J,K,L)/AUXVW
C
    TOX2SI(NX,J,K,L)=(1-DELTO)*FFX(NX+1,J,K,L)*AUXBO*
  *   DKRO(NX,J,K,L)/AUXVO
    TOX2SP(NX,J,K,L)=DELTO*FFX(NX+1,J,K,L)*AUXBO*
  *   DKRO(IX,JX,KP+K-1,2)/AUXVO
    TWX2SI(NX,J,K,L)=(1-DELTW)*FFX(NX+1,J,K,L)*AUXBW*
  *   DKRW(NX,J,K,L)/AUXVW
    TWX2SP(NX,J,K,L)=DELTW*FFX(NX+1,J,K,L)*AUXBW*
  *   DKRW(IX,JX,KP+K-1,2)/AUXVW
  *
    ENDIF
C
20  CONTINUE
-----
C INTERFACES Z:Transmissib. ao oleo e a agua e suas derivadas dT/dSw
C
C Verificacao do sentido de fluxo nas interfaces Z
C para "head" positivo :delto e deltg =1 (fluxo para dentro "do papel")
C para "head" negativo :delto e deltg =0 (fluxo para fora "do papel")
C Res. horizontal a diferenca entre ZB's e ZERO.
C Sentido de fluxo da 1 e 2 camada
-----
      DO 60 I=1,NX
        DO 60 J=1,NY
          IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP) THEN
            IF(KP.EQ.1.OR.KP.EQ.2) GO TO 60
          ENDIF
          IF(L.EQ.1.AND.KP.NE.1) THEN

```

C quando tem fluxo na primeira interface cilindro/cartesiano

```

HEAD=PO(IP,JP,KP-1,2)-PO(I,J,1,1)-DOMZ(I,J,1,1)*
*      (ZB(I,J,1)-ZB(IP,JP,2))
DELTO=1
IF(HEAD.LT.0.0DO) DELTO=0
HEAD=PO(IP,JP,KP-1,2)-PO(I,J,1,1)-PC(IP,JP,KP-1,2)+PC(I,J,1,1)-
*      DWMZ(I,J,1,1)*(ZB(I,J,1)-ZB(IP,JP,2))
DELTW=1
IF(HEAD.LT.0.0DO) DELTW=0
TOZ(I,J,1,1)=DELTO*FFZ(I,J,1,1)*BOMZ(I,J,1,1)*
*      KRO(IP,JP,KP-1,2)/VOMZ(I,J,1,1)+
*      (1-DELTO)*FFZ(I,J,1,1)*BOMZ(I,J,1,1)*
*      KRO(I,J,1,1)/VOMZ(I,J,1,1)
TWZ(I,J,1,1)=DELTW*FFZ(I,J,1,1)*BWMZ(I,J,1,1)*
*      KRW(IP,JP,KP-1,2)/VWMZ(I,J,1,1)+
*      (1-DELTW)*FFZ(I,J,1,1)*BWMZ(I,J,1,1)*
*      KRW(I,J,1,1)/VWMZ(I,J,1,1)

```

```

C      TOZ1SA(I,J,1,1)=DELTO*FFZ(I,J,1,1)*BOMZ(I,J,1,1)*
*      DKRO(IP,JP,KP-1,2)/VOMZ(I,J,1,1)
TOZ1SI(I,J,1,1)=(1-DELTO)*FFZ(I,J,1,1)*BOMZ(I,J,1,1)*
*      DKRO(I,J,1,1)/VOMZ(I,J,1,1)
TWZ1SA(I,J,1,1)=DELTW*FFZ(I,J,1,1)*BWMZ(I,J,1,1)*
*      DKRW(IP,JP,KP-1,2)/VWMZ(I,J,1,1)
TWZ1SI(I,J,1,1)=(1-DELTW)*FFZ(I,J,1,1)*BWMZ(I,J,1,1)*
*      DKRW(I,J,1,1)/VWMZ(I,J,1,1)

```

ENDIF

```

C      HEAD=PO(I,J,1,L)-PO(I,J,2,L)
DELTO=1
IF(HEAD.LT.0.0DO) DELTO=0
HEAD=PO(I,J,1,L)-PO(I,J,2,L)-PC(I,J,1,L)+PC(I,J,2,L)
DELTW=1
IF(HEAD.LT.0.0DO) DELTW=0
TOZ2SI(I,J,1,L)=DELTO*FFZ(I,J,2,L)*BOMZ(I,J,2,L)*
*      DKRO(I,J,1,L)/VOMZ(I,J,2,L)
TOZ2SP(I,J,1,L)=(1-DELTO)*FFZ(I,J,2,L)*BOMZ(I,J,2,L)*
*      DKRO(I,J,2,L)/VOMZ(I,J,2,L)
TWZ2SI(I,J,1,L)=DELTW*FFZ(I,J,2,L)*BWMZ(I,J,2,L)*
*      DKRW(I,J,1,L)/VWMZ(I,J,2,L)
TWZ2SP(I,J,1,L)=(1-DELTW)*FFZ(I,J,2,L)*BWMZ(I,J,2,L)*
*      DKRW(I,J,2,L)/VWMZ(I,J,2,L)

```

DTO(I,J)=DELTO

DTW(I,J)=DELTW

60 CONTINUE

```

C      DO 30 I=1,NX
DO 30 J=1,NY
DO 30 K=2,NZ-1
IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP) THEN
IF(K.GE.KP.AND.K.LT.KP+NBP) GO TO 30
ENDIF
IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP.AND.K.EQ.KP+NBP)GO TO 61
IF(K.EQ.2) THEN
DELTO=DTO(I,J)
DELTW=DTW(I,J)
ENDIF
TOZ(I,J,K,L)=DELTO*FFZ(I,J,K,L)*BOMZ(I,J,K,L)*
*      KRO(I,J,K-1,L)/VOMZ(I,J,K,L)+
*      (1-DELTO)*FFZ(I,J,K,L)*BOMZ(I,J,K,L)*

```

```

*          KRO(I,J,K,L)/VOMZ(I,J,K,L)
* TWZ(I,J,K,L)=DELTW*FFZ(I,J,K,L)*BWMZ(I,J,K,L)*
*          KRW(I,J,K-1,L)/VWMZ(I,J,K,L)+
*          (1-DELTW)*FFZ(I,J,K,L)*BWMZ(I,J,K,L)*
*          KRW(I,J,K,L)/VWMZ(I,J,K,L)
C
*          TOZ1SA(I,J,K,L)=DELTO*FFZ(I,J,K,L)*BOMZ(I,J,K,L)*
*          DKRO(I,J,K-1,L)/VOMZ(I,J,K,L)
*          TOZ1SI(I,J,K,L)=(1-DELTO)*FFZ(I,J,K,L)*BOMZ(I,J,K,L)*
*          DKRO(I,J,K,L)/VOMZ(I,J,K,L)
*          TWZ1SA(I,J,K,L)=DELTW*FFZ(I,J,K,L)*BWMZ(I,J,K,L)*
*          DKRW(I,J,K-1,L)/VWMZ(I,J,K,L)
*          TWZ1SI(I,J,K,L)=(1-DELTW)*FFZ(I,J,K,L)*BWMZ(I,J,K,L)*
*          DKRW(I,J,K,L)/VWMZ(I,J,K,L)
C Repete-se a verificacao do sentido de fluxo para a proxima interface Z
C o res. e horizontal, dai deltaz=0
61      IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP.AND.K.EQ.KP-1)GOTO 30
      HEAD=PO(I,J,K,L)-PO(I,J,K+1,L)
      DELTO=1
      IF(HEAD.LT.0.0DO) DELTO=0
      HEAD=PO(I,J,K,L)-PO(I,J,K+1,L)-PC(I,J,K,L)+PC(I,J,K+1,L)
      DELTW=1
      IF(HEAD.LT.0.0DO) DELTW=0
      TOZ2SI(I,J,K,L)=DELTO*FFZ(I,J,K+1,L)*BOMZ(I,J,K+1,L)*
*          DKRO(I,J,K,L)/VOMZ(I,J,K+1,L)
*          TOZ2SP(I,J,K,L)=(1-DELTO)*FFZ(I,J,K+1,L)*BOMZ(I,J,K+1,L)*
*          DKRO(I,J,K+1,L)/VOMZ(I,J,K+1,L)
*          TWZ2SI(I,J,K,L)=DELTW*FFZ(I,J,K+1,L)*BWMZ(I,J,K+1,L)*
*          DKRW(I,J,K,L)/VWMZ(I,J,K+1,L)
*          TWZ2SP(I,J,K,L)=(1-DELTW)*FFZ(I,J,K+1,L)*BWMZ(I,J,K+1,L)*
*          DKRW(I,J,K+1,L)/VWMZ(I,J,K+1,L)
30      CONTINUE
C-----
C INTERF.Z:Transmissib. ao oleo e a agua e suas derivadas dT/dSw na
C      ultima camada de celulas (no limite do reserv. To e Tw = 0)
C-----
DO 35 I=1,NX
DO 35 J=1,NY
IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP.AND.KP+NB-1.GE.NZC-1)GOTO 35
HEAD=PO(I,J,NZ-1,L)-PO(I,J,NZ,L)
DELTO=1
IF(HEAD.LT.0.0DO) DELTO=0
HEAD=PO(I,J,NZ-1,L)-PO(I,J,NZ,L)-PC(I,J,NZ-1,L)+PC(I,J,NZ,L)
DELTW=1
IF(HEAD.LT.0.0DO) DELTW=0
TOZ(I,J,NZ,L)=DELTO*FFZ(I,J,NZ,L)*BOMZ(I,J,NZ,L)*
*          KRO(I,J,NZ-1,L)/VOMZ(I,J,NZ,L)+
*          (1-DELTO)*FFZ(I,J,NZ,L)*BOMZ(I,J,NZ,L)*
*          KRO(I,J,NZ,L)/VOMZ(I,J,NZ,L)
*          TWZ(I,J,NZ,L)=DELTW*FFZ(I,J,NZ,L)*BWMZ(I,J,NZ,L)*
*          KRW(I,J,NZ-1,L)/VWMZ(I,J,NZ,L)+
*          (1-DELTW)*FFZ(I,J,NZ,L)*BWMZ(I,J,NZ,L)*
*          KRW(I,J,NZ,L)/VWMZ(I,J,NZ,L)
C
*          TOZ1SA(I,J,NZ,L)=DELTO*FFZ(I,J,NZ,L)*BOMZ(I,J,NZ,L)*
*          DKRO(I,J,NZ-1,L)/VOMZ(I,J,NZ,L)
*          TOZ1SI(I,J,NZ,L)=(1-DELTO)*FFZ(I,J,NZ,L)*BOMZ(I,J,NZ,L)*
*          DKRO(I,J,NZ,L)/VOMZ(I,J,NZ,L)
C

```

```

      TWZ1SA(I,J,NZ,L)=DELTW*FFZ(I,J,NZ,L)*BWMZ(I,J,NZ,L)*
      *      DKRW(I,J,NZ-1,L)/VWMZ(I,J,NZ,L)
      TWZ1SI(I,J,NZ,L)=(1-DELTW)*FFZ(I,J,NZ,L)*BWMZ(I,J,NZ,L)*
      *      DKRW(I,J,NZ,L)/VWMZ(I,J,NZ,L)
C  quando tem fluxo na ultima interface cilindro/cartesiano
      IF(L.EQ.1.AND.KP+NBP-1.LT.NZC) THEN
C
      HEAD=PO(I,J,NZR,1)-PO(IP,JP,KP+NBP,2)
      HEAD=PO(I,J,NZR,1)-PO(IP,JP,KP+NBP,2)-DOMZ(I,J,NZR+1,1)*
      *      (ZB(IP,JP,2)-ZB(I,J,1))
      DELTO=1
      IF(HEAD.LT.0.0D0) DELTO=0
      HEAD=PO(I,J,NZR,1)-PO(IP,JP,KP+NBP,2)-
      *      PC(I,J,NZR,1)+PC(IP,JP,KP+NBP,2)-DWMZ(I,J,NZR+1,1)*
      *      (ZB(IP,JP,2)-ZB(I,J,1))
      DELTW=1
      IF(HEAD.LT.0.0D0) DELTW=0
      TOZ(I,J,NZR+1,1)=DELTO*FFZ(I,J,NZR+1,1)*BOMZ(I,J,NZR+1,1)*
      *      KRO(I,J,NZR,1)/VOMZ(I,J,NZR+1,1)+
      *      (1-DELTO)*FFZ(I,J,NZR+1,1)*BOMZ(I,J,NZR+1,1)*
      *      KRO(IP,JP,KP+NBP,2)/VOMZ(I,J,NZR+1,1)

      TWZ(I,J,NZR+1,1)=DELTW*FFZ(I,J,NZR+1,1)*BWMZ(I,J,NZR+1,1)*
      *      KRW(I,J,NZR,1)/VWMZ(I,J,NZR+1,1)+
      *      (1-DELTW)*FFZ(I,J,NZR+1,1)*BWMZ(I,J,NZR+1,1)*
      *      KRW(IP,JP,KP+NBP,2)/VWMZ(I,J,NZR+1,1)
      TOZ2SI(I,J,NZR,1)=DELTO*FFZ(I,J,NZR+1,1)*
      *      BOMZ(I,J,NZR+1,1)*DKRO(I,J,NZR,1)/VOMZ(I,J,NZR+1,1)
      TOZ2SP(I,J,NZR,1)=(1-DELTO)*FFZ(I,J,NZR+1,1)*
      *      BOMZ(I,J,NZR+1,1)*DKRO(IP,JP,KP+NBP,2)/VOMZ(I,J,NZR+1,1)
      TWZ2SI(I,J,NZR,1)=DELTW*FFZ(I,J,NZR+1,1)*
      *      BWMZ(I,J,NZR+1,1)*DKRW(I,J,NZR,1)/VWMZ(I,J,NZR+1,1)
      TWZ2SP(I,J,NZR,1)=(1-DELTW)*FFZ(I,J,NZR+1,1)*
      *      BWMZ(I,J,NZR+1,1)*DKRW(IP,JP,KP+NBP,2)/VWMZ(I,J,NZR+1,1)
      ENDIF
35  CONTINUE
C-----
C INTERF.Y:Transmissib. ao oleo e a agua e suas derivadas dT/dSw na
C Verificacao do sentido de fluxo
C para "head" positivo :delto e deltw =1 (fluxo no sentido anti horario)
C                                     ou (fluxo para cima)
C para "head" negativo :delto e deltw =0 (fluxo no sentido horario)
C                                     ou (fluxo para baixo)
C-----
      J1=1
      J2=2
      IF(L.EQ.1) THEN
        J1=NY
        J2=1
      ENDIF
      DO 80 I=1,NX
      DO 80 K=1,NZ
        IF(L.EQ.2.AND.K.GE.KP.AND.K.LT.KP+NBP) THEN
          IF(I.EQ.IP.AND.JP.EQ.1.OR.
      *      I.EQ.IP.AND.JP.EQ.2) GO TO 80
          ENDIF
          HEAD=PO(I,J1,K,L)-PO(I,J2,K,L)+DOMY(I,J2,K,L)*
      *      (ZB(I,J1,L)-ZB(I,J2,L))
          DELTO=1
          IF(HEAD.LT.0.0D0) DELTO=0
          HEAD=PO(I,J1,K,L)-PO(I,J2,K,L)-PC(I,J1,K,L)+PC(I,J2,K,L)+

```

```

*          DWMY(I,J2,K,L)*(ZB(I,J1,L)-ZB(I,J2,L))
DELTW=1
IF(HEAD.LT.0.0D0) DELTW=0
IF(L.EQ.2) THEN
TOY2SI(I,1,K,L)=DELTO*FFY(I,2,K,L)*BOMY(I,2,K,L)*
*          DKRO(I,1,K,L)/VOMY(I,2,K,L)
TOY2SP(I,1,K,L)=(1-DELTO)*FFY(I,2,K,L)*BOMY(I,2,K,L)*
*          DKRO(I,2,K,L)/VOMY(I,2,K,L)
TWY2SI(I,1,K,L)=DELTW*FFY(I,2,K,L)*BWMY(I,2,K,L)*
*          DKRW(I,1,K,L)/VWMY(I,2,K,L)
TWY2SP(I,1,K,L)=(1-DELTW)*FFY(I,2,K,L)*BWMY(I,2,K,L)*
*          DKRW(I,2,K,L)/VWMY(I,2,K,L)
ENDIF
DTO(I,K)=DELTO
DTW(I,K)=DELTW
80  CONTINUE
    IPO=1
    IPW=1
C
C JME eh jmenos ou j-1
C JMA eh jmais ou j+1
    JLIM=NY
    IF(L.EQ.2) JLIM=NY-1
    DO 70 K=1,NZ
        DO 70 I=1,NX
            DO 70 J=L,JLIM
                IF(L.EQ.2.AND.K.GE.KP.AND.K.LT.KP+NBP) THEN
                    IF(I.EQ.IP.AND.J.EQ.JP) GO TO 70
                    IF(I.EQ.IP.AND.J.EQ.JP+1) GO TO 71
                ENDIF
                IF(J.EQ.L) THEN
                    DELTO=DTO(I,K)
                    DELTW=DTW(I,K)
                ENDIF
                JME=J-1
                IF(L.EQ.1.AND.J.EQ.1) JME=NY
                TOY(I,J,K,L)=IPO*DELTO*FFY(I,J,K,L)*BOMY(I,J,K,L)*
*          KRO(I,JME,K,L)/VOMY(I,J,K,L)+
*          IPO*(1-DELTO)*FFY(I,J,K,L)*BOMY(I,J,K,L)*
*          KRO(I,J,K,L)/VOMY(I,J,K,L)
                TWY(I,J,K,L)=IPW*DELTW*FFY(I,J,K,L)*BWMY(I,J,K,L)*
*          KRW(I,JME,K,L)/VWMY(I,J,K,L)+
*          IPW*(1-DELTW)*FFY(I,J,K,L)*BWMY(I,J,K,L)*
*          KRW(I,J,K,L)/VWMY(I,J,K,L)
C
                TOY1SA(I,J,K,L)=IPO*DELTO*FFY(I,J,K,L)*BOMY(I,J,K,L)*
*          DKRO(I,JME,K,L)/VOMY(I,J,K,L)
                TOY1SI(I,J,K,L)=IPO*(1-DELTO)*FFY(I,J,K,L)*BOMY(I,J,K,L)*
*          DKRO(I,J,K,L)/VOMY(I,J,K,L)
                TWY1SA(I,J,K,L)=IPW*DELTW*FFY(I,J,K,L)*BWMY(I,J,K,L)*
*          DKRW(I,JME,K,L)/VWMY(I,J,K,L)
                TWY1SI(I,J,L,K)=IPW*(1-DELTW)*FFY(I,J,K,L)*BWMY(I,J,K,L)*
*          DKRW(I,J,K,L)/VWMY(I,J,K,L)
C Repete-se a verificacao do sentido de fluxo
    IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP-1) GO TO 70
71    JMA=J+1
        IF(L.EQ.1.AND.J.EQ.NY) JMA=1
        HEAD=PO(I,J,K,L)-PO(I,JMA,K,L)+DOMY(I,JMA,K,L)*
*          (ZB(I,J,L)-ZB(I,JMA,L))
        DELTO=1

```

```

      IPO=1
      IF(HEAD.LT.0.0DO) DELTO=0
      IF(HEAD.EQ.0.0DO) IPO=0
      HEAD=PO(I,J,K,L)-PO(I,JMA,K,L)-PC(I,J,K,L)+PC(I,JMA,K,L)+
*      DWMY(I,JMA,K,L)*(ZB(I,J,L)-ZB(I,JMA,L))
      DELTW=1
      IPW=1
      IF(HEAD.LT.0.0DO) DELTW=0
      IF(HEAD.EQ.0.0DO) IPW=0
      TOY2SI(I,J,K,L)=IPO*DELTO*FFY(I,JMA,K,L)*BOMY(I,JMA,K,L)*
*      DKRO(I,J,K,L)/VOMY(I,JMA,K,L)
      TOY2SP(I,J,K,L)=IPO*(1-DELTO)*FFY(I,JMA,K,L)*BOMY(I,JMA,K,L)*
*      DKRO(I,JMA,K,L)/VOMY(I,JMA,K,L)
      TWY2SI(I,J,K,L)=IPW*DELTW*FFY(I,JMA,K,L)*BWMY(I,JMA,K,L)*
*      DKRW(I,J,K,L)/VWMY(I,JMA,K,L)
      TWY2SP(I,J,K,L)=IPW*(1-DELTW)*FFY(I,JMA,K,L)*BWMY(I,JMA,K,L)*
*      DKRW(I,JMA,K,L)/VWMY(I,JMA,K,L)
70  CONTINUE
C-----
C INTERF.Y:Transmissib. ao oleo e a agua e suas derivadas dT/dSw na
C      ultima camada de celulas (no limite do reserv. To e Tw = 0)
C-----
      IF(L.EQ.2) THEN
      DO 75 I=1,NX
      DO 75 K=1,NZ
      IF(K.GE.KP.AND.K.LT.KP+NBP) THEN
      IF(I.EQ.IP.AND.JP.EQ.NY.OR.I.EQ.IP.AND.JP.EQ.NY-1) GO TO 75
      ENDIF
      HEAD=PO(I,NY-1,K,L)-PO(I,NY,K,L)+DOMY(I,NY,K,L)*
*      (ZB(I,NY-1,L)-ZB(I,NY,L))
      DELTO=1
      IF(HEAD.LT.0.0DO) DELTO=0
      HEAD=PO(I,NY-1,K,L)-PO(I,NY,K,L)-PC(I,NY-1,K,L)+PC(I,NY,K,L)+
*      DWMY(I,NY,K,L)*(ZB(I,NY-1,L)-ZB(I,NY,L))
      DELTW=1
      IF(HEAD.LT.0.0DO) DELTW=0
      TOY(I,NY,K,L)=DELTO*FFY(I,NY,K,L)*BOMY(I,NY,K,L)*
*      KRO(I,NY-1,K,L)/VOMY(I,NY,K,L)+
*      (1-DELTO)*FFY(I,NY,K,L)*BOMY(I,NY,K,L)*
*      KRO(I,NY,K,L)/VOMY(I,NY,K,L)
      TWY(I,NY,K,L)=DELTW*FFY(I,NY,K,L)*BWMY(I,NY,K,L)*
*      KRW(I,NY-1,K,L)/VWMY(I,NY,K,L)+
*      (1-DELTW)*FFY(I,NY,K,L)*BWMY(I,NY,K,L)*
*      KRW(I,NY,K,L)/VWMY(I,NY,K,L)
C
      TOY1SA(I,NY,K,L)=DELTO*FFY(I,NY,K,L)*BOMY(I,NY,K,L)*
*      DKRO(I,NY-1,K,L)/VOMY(I,NY,K,L)
      TOY1SI(I,NY,K,L)=(1-DELTO)*FFY(I,NY,K,L)*BOMY(I,NY,K,L)*
*      DKRO(I,NY,K,L)/VOMY(I,NY,K,L)
C
      TWY1SA(I,NY,K,L)=DELTW*FFY(I,NY,K,L)*BWMY(I,NY,K,L)*
*      DKRW(I,NY-1,K,L)/VWMY(I,NY,K,L)
      TWY1SI(I,NY,K,L)=(1-DELTW)*FFY(I,NY,K,L)*BWMY(I,NY,K,L)*
*      DKRW(I,NY,K,L)/VWMY(I,NY,K,L)
75  CONTINUE
      ENDIF
C 1000 CONTINUE
C
      RETURN

```

END

 C SUBROT. DE REINICIALIZ. DE VAR. PARA REPETICAO DE PASSO DE TEMPO QUE
 C ULTRAPASSOU O N# MAX DE ITER OU QUE NAO TENHA VERIF. DSLIM OU DPLIM
 C -----

SUBROUTINE REPTS(NTS,PWFA,PWF)

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
 PARAMETER(NXM=08,NYM=05,NZM=04)
 COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
 * NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
 COMMON/TSA/POA,SOA,SGA,SWA
 COMMON/TS/PO,SO,SG,SW

DIMENSION POA(NXM,NYM,NZM,2),SOA(NXM,NYM,NZM,2),
 * SGA(NXM,NYM,NZM,2),SWA(NXM,NYM,NZM,2)
 DIMENSION PO(0:NXM+1,0:NYM+1,0:NZM+1,2),SO(NXM,NYM,NZM,2),
 * SG(NXM,NYM,NZM,2),SW(NXM,NYM,NZM,2)

DO 4 L=1,2

NX=NXR

NY=NYR

NZ=NZR

IF(L.EQ.2) THEN

NX=NXC

NY=NYC

NZ=NZC

ENDIF

DO 5 I=1,NX

DO 5 J=1,NY

DO 5 K=1,NZ

IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.

* K.LT.KP+NBP) GO TO 5
 PO(I,J,K,L)=POA(I,J,K,L)
 SO(I,J,K,L)=SOA(I,J,K,L)
 SW(I,J,K,L)=SWA(I,J,K,L)

SG(I,J,K,L)=SGA(I,J,K,L)

CONTINUE

CONTINUE

NTS=NTS-1

PWF=PWFA

RETURN

END

 C SUBROTINA DE CALCULO DAS FUNCOES DE RESIDUOS DO OLEO E DO GAS E DE
 C SUAS DERIVADAS PARA A MONTAGEM DA MATRIZ JACOBIANA
 C Obs.IDR eh para anular ou nao os termos de fluxo, vazao e suas deriva-
 C das no poco do GRID RADIAL.
 C IDC eh para anular ou nao a vazao e suas derivadas no bloco do
 C poco do GRID CARTESIANO.
 C -----

SUBROUTINE MONJAC

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
 PARAMETER(NXM=08,NYM=05,NZM=04)
 COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
 * NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
 COMMON/SIM/DT,DTMAX,DTMIN,TOLP,TOLS,DSLIM,DPLIM,TSIM,
 * QOMAX,QOMIN,PWMIN,BSOR,JPROD,MSOLU,ITMAX,ITMIN
 COMMON/RES/DGAS,CR,PRIN,TEMP,ZC,RW,RE,SWIN,PHI,PERX,PERY,

```

*      PERZ,ZCOW,INDPOR,INDPER
COMMON/MALHA/ZB,RB,VPOR,DRO,WX,WY,WZ,FFX,FFY,FFZ,DTETA
COMMON/TSA/POA,SOA,SGA,SWA
COMMON/PVTA/BOA,RSOA,BWA
COMMON/TS/PO,SO,SG,SW
COMMON/PVT/BO,DBO,VO,DVO,DO,DDO,RSO,DRSO,BW,DBW,VW,DVW,DW,DDW,CW
COMMON/IFACEX/BOMX,VOMX,DOMX,BWMX,VWMX,DWMX
COMMON/IFACEY/BOMY,VOMY,DOMY,BWMY,VWMY,DWMY
COMMON/IFACEZ/BOMZ,VOMZ,DOMZ,BWMZ,VWMZ,DWMZ
COMMON/PETR/KRO,KRW,DKRO,DKRW,PC,DPC
COMMON/DTRANS/TOX,TWX,TOY,TWY,TOZ,TWZ,
*      TOX1SA,TOX1SI,TWX1SA,TWX1SI,
*      TOX2SI,TOX2SP,TWX2SI,TWX2SP,
*      TOY1SA,TOY1SI,TWY1SA,TWY1SI,
*      TOY2SI,TOY2SP,TWY2SI,TWY2SP,
*      TOZ1SA,TOZ1SI,TWZ1SA,TWZ1SI,
*      TOZ2SI,TOZ2SP,TWZ2SI,TWZ2SP
COMMON/VAZAO/QO,QOT,QG,QGT,QW,QWT,DQOP,DQOS,
*      DQWP,DQWS,RWO,RGO,PWF,TEMPO
COMMON/ANULA/IX1,IX2,IY1,IY2,IZ1,IZ2,IDR,IDC,LBC,JMA,JME,IX,JX,LBR
COMMON/NBLC/A,D,NBC,NBR,NBT
COMMON/ACU/ACO,ACW,DACOP,DACOS,DACWP,DACWS
COMMON/POT/POTOX1,POTOX2,POTOY1,POTOY2,POTOZ1,POTOZ2,
*      POTWX1,POTWX2,POTWY1,POTWY2,POTWZ1,POTWZ2,
*      POX1PA,POX1PI,POY1PA,POY1PI,POZ1PA,POZ1PI,
*      POX2PI,POX2PP,POY2PI,POY2PP,POZ2PI,POZ2PP,
*      PWX1PA,PWX1PI,PWY1PA,PWY1PI,PWZ1PA,PWZ1PI,
*      PWX2PI,PWX2PP,PWY2PI,PWY2PP,PWZ2PI,PWZ2PP,
*      TOX1PA,TOX1PI,TOY1PA,TOY1PI,TOZ1PA,TOZ1PI,
*      TOX2PI,TOX2PP,TOY2PI,TOY2PP,TOZ2PI,TOZ2PP,
*      TWX1PA,TWX1PI,TWY1PA,TWY1PI,TWZ1PA,TWZ1PI,
*      TWX2PI,TWX2PP,TWY2PI,TWY2PP,TWZ2PI,TWZ2PP
COMMON/SDEOW/STPO,STPW,SDEOPI,SDEOSI,SDEWPI,SDEWSI,
*      DFOPX,DFOSX,DFWPX,DFWSX,DFOPZ,DFOSZ,DFWPZ,DFWSZ
DIMENSION PHI(NXM,NYM,NZM,2),PERX(NXM,NYM,NZM,2),
*      PERY(NXM,NYM,NZM,2),PERZ(NXM,NYM,NZM,2)
DIMENSION ZB(0:NXM+1,NYM,2),RB(NXM+1),VPOR(NXM,NYM,NZM,2),
*      FFX(NXM+1,NYM,NZM,2),FFY(NXM+1,NYM,NZM,2),
*      FFZ(NXM+1,NYM,NZM+1,2),DTETA(NYM+1)
DIMENSION POA(NXM,NYM,NZM,2),SOA(NXM,NYM,NZM,2),
*      SGA(NXM,NYM,NZM,2),SWA(NXM,NYM,NZM,2)
DIMENSION BOA(NXM,NYM,NZM,2),BWA(NXM,NYM,NZM,2)
DIMENSION PO(0:NXM+1,0:NYM+1,0:NZM+1,2),SO(NXM,NYM,NZM,2),
*      SG(NXM,NYM,NZM,2),SW(NXM,NYM,NZM,2)
DIMENSION BO(NXM,NYM,NZM,2),DBO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*      VO(NXM,NYM,NZM,2),DVO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*      DO(NXM,NYM,NZM,2),DDO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*      BW(NXM,NYM,NZM,2),DBW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*      VW(NXM,NYM,NZM,2),DVW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*      DW(NXM,NYM,NZM,2),DDW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*      CW(NXM,NYM,NZM,2)
DIMENSION BOMX(0:NXM+1,NYM,NZM,2),VOMX(0:NXM+1,NYM,NZM,2),
*      DOMX(0:NXM+1,NYM,NZM,2),BWMX(0:NXM+1,NYM,NZM,2),
*      VWMX(0:NXM+1,NYM,NZM,2),DWMX(0:NXM+1,NYM,NZM,2)
DIMENSION BOMY(NXM,NYM+1,NZM,2),VOMY(NXM,NYM+1,NZM,2),
*      DOMY(NXM,NYM+1,NZM,2),BWMY(NXM,NYM+1,NZM,2),
*      VWMY(NXM,NYM+1,NZM,2),DWMY(NXM,NYM+1,NZM,2)
DIMENSION BOMZ(NXM,NYM,NZM+1,2),VOMZ(NXM,NYM,NZM+1,2),

```

```

*      DOMZ(NXM,NYM,NZM+1,2),BWMZ(NXM,NYM,NZM+1,2),
*      VWMZ(NXM,NYM,NZM+1,2),DWMZ(NXM,NYM,NZM+1,2)
  DIMENSION DKRO(NXM,NYM,NZM,2),DKRW(NXM,NYM,NZM,2),
*      PC(0:NXM+1,0:NYM+1,0:NZM+1,2),DPC(0:NXM+1,0:NYM+1,0:NZM+1,2)
  REAL*8      KRO(NXM,NYM,NZM,2),KRW(NXM,NYM,NZM,2)
  DIMENSION TOX(NXM+1,NYM,NZM,2),TWX(NXM+1,NYM,NZM,2),
*      TOX1SA(NXM,NYM,NZM,2),TOX1SI(NXM,NYM,NZM,2),
*      TWX1SA(NXM,NYM,NZM,2),TWX1SI(NXM,NYM,NZM,2),
*      TOX2SI(NXM,NYM,NZM,2),TOX2SP(NXM,NYM,NZM,2),
*      TWX2SI(NXM,NYM,NZM,2),TWX2SP(NXM,NYM,NZM,2)
  DIMENSION TOY(NXM,NYM+1,NZM,2),TWY(NXM,NYM+1,NZM,2),
*      TOY1SA(NXM,NYM,NZM,2),TOY1SI(NXM,NYM,NZM,2),
*      TWY1SA(NXM,NYM,NZM,2),TWY1SI(NXM,NYM,NZM,2),
*      TOY2SI(NXM,NYM,NZM,2),TOY2SP(NXM,NYM,NZM,2),
*      TWY2SI(NXM,NYM,NZM,2),TWY2SP(NXM,NYM,NZM,2)
  DIMENSION TOZ(NXM,NYM,NZM+1,2),TWZ(NXM,NYM,NZM+1,2),
*      TOZ1SA(NXM,NYM,NZM,2),TOZ1SI(NXM,NYM,NZM,2),
*      TWZ1SA(NXM,NYM,NZM,2),TWZ1SI(NXM,NYM,NZM,2),
*      TOZ2SI(NXM,NYM,NZM,2),TOZ2SP(NXM,NYM,NZM,2),
*      TWZ2SI(NXM,NYM,NZM,2),TWZ2SP(NXM,NYM,NZM,2)
  DIMENSION A(NXM*NYM*NZM*4,NXM*NYM*NZM*4),D(NXM*NYM*NZM*4)
  DIMENSION QO(NYM,NZM),QG(NYM,NZM),QW(NYM,NZM),DQOP(NYM,NZM),
*      DQOS(NYM,NZM),DQWP(NYM,NZM),DQWS(NYM,NZM)
  DIMENSION STPO(0:6,NZM),STPW(0:6,NZM)
C      STPO --> Somatorio de Trans. X Pot. do Oleo
  DIMENSION SDEOPI(0:6,NZM),SDEWPI(0:6,NZM),SDEOSI(0:6,NZM),
*      SDEWSI(0:6,NZM)
C      SDEOP--> Som. da Derivada da Eq. do Oleo em rel. a Pres.
C-----
C Inicializacao do jacobiano (matriz A) e do termo ind. com zeros
C-----
      NBC=NXC*NYC*NZC-NBP
      NBR=NXR*NYR*NZR
      NBT=NBC+NBR
C-----
C para evitar if's no calculo do potencial
C-----
      DO 1 K=KP,KP+NBP-1
        PO(IP,JP,K,2)=0.0D0
        SW(IP,JP,K,2)=0.0D0
1      CONTINUE
C-----
      DO 5 I=1,NBT*2
        DO 5 J=1,NBT*2
          A(I,J)=0.0D0
5      CONTINUE
      DO 10 I=1,NBT*2
        D(I)=0.0D0
10     CONTINUE
C-----
      DO 6 K=1,NZR
        DO 6 I=0,6
          STPO(I,K)=0.0D0
          STPW(I,K)=0.0D0
          SDEOPI(I,K)=0.0D0
          SDEWPI(I,K)=0.0D0
          SDEOSI(I,K)=0.0D0
          SDEWSI(I,K)=0.0D0
6      CONTINUE
C-----
C Calc. das funcoes de residuos do oleo e do gas e dos elementos do jac.

```

```

C-----
DO 1000 L=1,2
  NX=NXR
  NY=NYS
  NZ=NZR
  NC=NBC
  IF(L.EQ.2) THEN
    NX=NXC
    NY=NYC
    NZ=NZC
    NC=0
  ENDIF
C
DO 15 I=1,NX
  DO 15 J=1,NY
    DO 15 K=1,NZ
      IF(L.EQ.2.AND.I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.
*      K.LT.KP+NBP) GO TO 15
C-----
C Fator de controle para a condicao de contorno das celulas
C   idr --> indicador do poco no radial (=0 se i=1 e =1 se i=/1)
C   idc --> indica se o bl cart eh viz do poco (=1 se sim, =0 se nao)
C   ix1 --> anulara as eqs. em i-1/2 qdo for face com o bloco do poco
C   ix2 --> anulara as eqs. em i+1/2 qdo for face com o bloco do poco
C   iy1, iy2, iz1 e iz2 --> idem a ix1 e ix2
C   lbc --> posicao do cartesiano vizinho ao poco
C       > lbc=1  cartesiano vizinho a direita ou esquerda do poco
C       > lbc=2  cartesiano vizinho acima do poco
C       > lbc=3  cartesiano vizinho abaixo do poco
C-----
      JMA=J+1
      JME=J-1
      IDR=1
      IDC=0
      LBC=0
      LBR=1
      IX1=1
      IX2=1
      IY1=1
      IY2=1
      IZ1=1
      IZ2=1
      IV=0
      IX=0
      IY=0
      DFOPX=0.000
      DFOSX=0.000
      DFWPX=0.000
      DFWSX=0.000
      DFOPZ=0.000
      DFOSZ=0.000
      DFWPZ=0.000
      DFWSZ=0.000

      IF(L.EQ.1) THEN
        IF(I.EQ.1) IDR=0
        IF(J.EQ.1) JME=NY
        IF(J.EQ.NY) JMA=1
        W=WX
      ELSE
        IF(K.GE.KP.AND.K.LT.KP+NBP) THEN
          IF(I.EQ.IP+1.AND.J.EQ.JP) THEN
            IDC=1
            IX1=0
            LBC=1
            LBR=K-KP+1
          ENDIF

```

```

      IF(I.EQ.IP.AND.J.EQ.JP+1) THEN
        IDC=1
        IY1=0
        LBC=2
        LBR=K-KP+1
      ENDIF
      IF(I.EQ.IP-1.AND.J.EQ.JP) THEN
        IDC=1
        IX2=0
        LBC=3
        LBR=K-KP+1
      ENDIF
      IF(I.EQ.IP.AND.J.EQ.JP-1) THEN
        IDC=1
        IY2=0
        LBC=4
        LBR=K-KP+1
      ENDIF
    ENDIF
    IF(I.EQ.IP.AND.J.EQ.JP.AND.K.EQ.KP-1) THEN
      IDC=1
      IZ2=0
      LBC=5
      LBR=1
    ENDIF
    IF(I.EQ.IP.AND.J.EQ.JP.AND.K.EQ.KP+NBP) THEN
      IDC=1
      IZ1=0
      LBC=6
      LBR=NZR
    ENDIF
    W=0.5D0
  ENDIF
C-----
C Numero da celula na ordenacao sequencial de celulas na malha
C-----
  NC=NC+1
C-----
C Endereco dos elementos da diagonal principal da celula i,j no jacob.
C-----
  LC=2*NC-1
C-----
C Calculo das dif.de potencial nas fases oleo e agua
C ix e iy dah as coordenadas do cartesiano vizinho
C-----
  IF(L.EQ.1.AND.I.EQ.NX) THEN
C lateral direita
    IF(J.LE.NQL) THEN
      IV=1
      IX=IP+1
      JX=JP
    ENDIF
C lado superior
    IF(J.GT.NQL.AND.J.LE.NQL+NQS) THEN
      IV=2
      IX=IP
      JX=JP+1
    ENDIF
C lateral esquerda
    IF(J.GT.NQL+NQS.AND.J.LE.NQS+2*NQL) THEN
      IV=3
      IX=IP-1
      JX=JP
    ENDIF
C lado inferior
    IF(J.GT.NQS+2*NQL) THEN
      IV=4
      IX=IP
      JX=JP-1
    ENDIF
  ENDIF

```

```

ENDIF
AUXDO=DOMX(NX+1,J,K,1)
AUXDW=DWMX(NX+1,J,K,1)
AUXBO=BOMX(NX+1,J,K,1)
AUXBW=BWMX(NX+1,J,K,1)
AUXVO=VOMX(NX+1,J,K,1)
AUXVW=VWMX(NX+1,J,K,1)
POTOX2=PO(IX,JX,KP+K-1,2)-PO(NX,J,K,1)+AUXDO*
*      (ZB(IX,JX,2)-ZB(NX,J,1))
POTWX2=PO(IX,JX,KP+K-1,2)-PO(NX,J,K,1)+AUXDW*
*      (ZB(IX,JX,2)-ZB(NX,J,1))-PC(IX,JX,KP+K-1,2)+PC(NX,J,K,1)
C
STPO(IV,K)=STPO(IV,K)-TOX(NX+1,J,K,L)*POTOX2
STPW(IV,K)=STPW(IV,K)-TWX(NX+1,J,K,L)*POTWX2
ELSE
POTOX2=PO(I+1,J,K,L)-PO(I,J,K,L)-DOMX(I+1,J,K,L)*
*      (ZB(I,J,L)-ZB(I+1,J,L))
POTWX2=PO(I+1,J,K,L)-PO(I,J,K,L)-DWMX(I+1,J,K,L)*
*      (ZB(I,J,L)-ZB(I+1,J,L))-PC(I+1,J,K,L)+PC(I,J,K,L)
ENDIF
POTOX1=IDR*(PO(I,J,K,L)-PO(I-1,J,K,L)-DOMX(I,J,K,L)*
*      (ZB(I-1,J,L)-ZB(I,J,L)))
POTWX1=IDR*(PO(I,J,K,L)-PO(I-1,J,K,L)-DWMX(I,J,K,L)*
*      (ZB(I-1,J,L)-ZB(I,J,L))-PC(I,J,K,L)+PC(I-1,J,K,L))
POTOY2=PO(I,JMA,K,L)-PO(I,J,K,L)-DOMY(I,JMA,K,L)*
*      (ZB(I,J,L)-ZB(I,JMA,L))
POTOY1=PO(I,J,K,L)-PO(I,JME,K,L)-DOMY(I,J,K,L)*
*      (ZB(I,JME,L)-ZB(I,J,L))
POTWY2=PO(I,JMA,K,L)-PO(I,J,K,L)-DWMY(I,JMA,K,L)*
*      (ZB(I,J,L)-ZB(I,JMA,L))-PC(I,JMA,K,L)+PC(I,J,K,L)
POTWY1=PO(I,J,K,L)-PO(I,JME,K,L)-DWMY(I,J,K,L)*
*      (ZB(I,JME,L)-ZB(I,J,L))-PC(I,J,K,L)+PC(I,JME,K,L)

```

C Na direcao Z a diferenca de ZB e zero

```

IF(L.EQ.1.AND.K.EQ.1.AND.KP.NE.1) THEN
IV=5
POTOZ1=PO(I,J,1,1)-PO(IP,JP,KP-1,2)-
*      DOMZ(I,J,1,1)*(ZB(IP,JP,2)-ZB(I,J,1))
POTWZ1=PO(I,J,1,1)-PO(IP,JP,KP-1,2)-
*      PC(I,J,1,1)+PC(IP,JP,KP-1,2)-
*      DWMZ(I,J,1,1)*(ZB(IP,JP,2)-ZB(I,J,1))
STPO(IV,K)=STPO(IV,K)+TOZ(I,J,1,1)*POTOZ1
STPW(IV,K)=STPW(IV,K)+TWZ(I,J,1,1)*POTWZ1
ELSE
POTOZ1=PO(I,J,K,L)-PO(I,J,K-1,L)
POTWZ1=PO(I,J,K,L)-PO(I,J,K-1,L)-
*      PC(I,J,K,L)+PC(I,J,K-1,L)
ENDIF
C
IF(L.EQ.1.AND.K.EQ.NZR.AND.KP+NBP-1.LT.NZC) THEN
IV=6
POTOZ2=PO(IP,JP,KP+NBP,2)-PO(I,J,NZR,1)-
*      DOMZ(I,J,NZR+1,1)*(ZB(I,J,1)-ZB(IP,JP,2))
POTWZ2=PO(IP,JP,KP+NBP,2)-PO(I,J,NZR,1)-
*      PC(IP,JP,KP+NBP,2)+PC(I,J,NZR,1)-
*      DWMZ(I,J,NZR+1,1)*(ZB(I,J,1)-ZB(IP,JP,2))
STPO(IV,K)=STPO(IV,K)-TOZ(I,J,NZR+1,1)*POTOZ2
STPW(IV,K)=STPW(IV,K)-TWZ(I,J,NZR+1,1)*POTWZ2
ELSE

```

```

      POTOZ2=PO(I,J,K+1,L)-PO(I,J,K,L)
      POTWZ2=PO(I,J,K+1,L)-PO(I,J,K,L)-
*      PC(I,J,K+1,L)+PC(I,J,K,L)
      ENDIF
C-----
C Calculo dos termos de acumulacao do oleo e agua na celula i,j
C-----
      AUX1=1.0DO+CR*(PO(I,J,K,L)-POA(I,J,K,L))
      ACW=VPOR(I,J,K,L)*(AUX1*BW(I,J,K,L)*SW(I,J,K,L)-
*      BWA(I,J,K,L)*SWA(I,J,K,L))/DT
      ACO=VPOR(I,J,K,L)*(AUX1*BO(I,J,K,L)*(1.0DO-
*      SW(I,J,K,L))-BOA(I,J,K,L)*SOA(I,J,K,L))/DT
C-----
C Determinacao das funcoes de residuos (termos ind. do sist. de eqs.)
C-----
      D(LC)=IX2*(TWX(I+1,J,K,L)*POTWX2)-IX1*(TWX(I,J,K,L)*POTWX1)+
*      IY2*(TWY(I,JMA,K,L)*POTWY2)-IY1*(TWY(I,J,K,L)*POTWY1)+
*      IZ2*(TWZ(I,J,K+1,L)*POTWZ2)-IZ1*(TWZ(I,J,K,L)*POTWZ1)-
*      ACW-(1-IDR)*QW(J,K)+IDC*STPW(LBC,LBR)
      D(LC)=-D(LC)
      D(LC+1)=IX2*(TOX(I+1,J,K,L)*POTOX2)-IX1*(TOX(I,J,K,L)*POTOX1)+
*      IY2*(TOY(I,JMA,K,L)*POTOY2)-IY1*(TOY(I,J,K,L)*POTOY1)+
*      IZ2*(TOZ(I,J,K+1,L)*POTOZ2)-IZ1*(TOZ(I,J,K,L)*POTOZ1)-
*      ACO-(1-IDR)*QO(J,K)+IDC*STPO(LBC,LBR)
      D(LC+1)=-D(LC+1)
C-----
C Calculo das derivadas das dif. de potencial em relacao a pressao
C-----
      IF(L.EQ.1.AND.I.EQ.NX) THEN
        POX2PI=-1.0DO-(1.0DO-WRC)*DDO(I,J,K,1)*
*        (ZB(NX,J,1)-ZB(IX,JX,2))
        POX2PP=1.0DO-WRC*DDO(IX,JX,K,2)*(ZB(NX,J,1)-ZB(IX,JX,2))
        PWX2PI=-1.0DO-(1.0DO-WRC)*DDW(I,J,K,1)*
*        (ZB(NX,J,1)-ZB(IX,JX,2))
        PWX2PP=1.0DO-WRC*DDW(IX,JX,KP,2)*(ZB(NX,J,1)-ZB(IX,JX,2))
      ELSE
        POX2PI=-1.0DO-(1.0DO-W)*DDO(I,J,K,L)*
*        (ZB(I,J,L)-ZB(I+1,J,L))
        POX2PP=1.0DO-W*DDO(I+1,J,K,L)*(ZB(I,J,L)-ZB(I+1,J,L))
        PWX2PI=-1.0DO-(1.0DO-W)*DDW(I,J,K,L)*
*        (ZB(I,J,L)-ZB(I+1,J,L))
        PWX2PP=1.0DO-W*DDW(I+1,J,K,L)*(ZB(I,J,L)-ZB(I+1,J,L))
      ENDIF
      POX1PA=IDR*(-1.0DO-(1.0DO-W)*DDO(I-1,J,K,L)*
*      (ZB(I-1,J,L)-ZB(I,J,L)))
      POX1PI=IDR*(1.0DO-W*DDO(I,J,K,L)*(ZB(I-1,J,L)-ZB(I,J,L)))
      PWX1PA=IDR*(-1.0DO-(1.0DO-W)*DDW(I-1,J,K,L)*
*      (ZB(I-1,J,L)-ZB(I,J,L)))
      PWX1PI=IDR*(1.0DO-W*DDW(I,J,K,L)*(ZB(I-1,J,L)-ZB(I,J,L)))
C
      POY1PA=-1.0DO-(1.0DO-WY)*DDO(I,JME,K,L)*
*      (ZB(I,JME,L)-ZB(I,J,L))
      POY1PI=1.0DO-WY*DDO(I,J,K,L)*(ZB(I,JME,L)-ZB(I,J,L))
      POY2PI=-1.0DO-(1.0DO-WY)*DDO(I,J,K,L)*(ZB(I,J,L)-ZB(I,JMA,L))
      POY2PP=1.0DO-WY*DDO(I,JMA,K,L)*(ZB(I,J,L)-ZB(I,JMA,L))
      PWY1PA=-1.0DO-(1.0DO-WY)*DDW(I,JME,K,L)*
*      (ZB(I,JME,L)-ZB(I,J,L))
      PWY1PI=1.0DO-WY*DDW(I,J,K,L)*(ZB(I,JME,L)-ZB(I,J,L))

```

```
PWY2PI=-1.0D0-(1.0D0-WY)*DDW(I,J,K,L)*(ZB(I,J,L)-ZB(I,JMA,L))
PWY2PP=1.0D0-WY*DDW(I,JMA,K,L)*(ZB(I,J,L)-ZB(I,JMA,L))
```

C

```
IF(L.EQ.1.AND.K.EQ.1.AND.KP.NE.1) THEN
  POZ1PA=-1.0D0-(1.0D0-WZ)*DDO(IP,JP,KP-1,2)*(ZB(IP,JP,2)-
    *      ZB(I,J,1))
  PWZ1PA=-1.0D0-(1.0D0-WZ)*DDW(IP,JP,KP-1,2)*(ZB(IP,JP,2)-
    *      ZB(I,J,1))
  POZ1PI=1.0D0+WZ*DDO(I,J,1,1)*(ZB(IP,JP,2)-ZB(I,J,1))
  PWZ1PI=1.0D0+WZ*DDW(I,J,1,1)*(ZB(IP,JP,2)-ZB(I,J,1))
ELSE
  POZ1PA=-1.0D0
  PWZ1PA=-1.0D0
  POZ1PI=1.0D0
  PWZ1PI=1.0D0
ENDIF
```

C

```
IF(L.EQ.1.AND.K.EQ.NZR.AND.KP+NBP-1.LT.NZC) THEN
  POZ2PI=-1.0D0+(1.0D0-WZ)*DDO(I,J,NZR,1)*(ZB(I,J,1)-
    *      ZB(IP,JP,2))
  PWZ2PI=-1.0D0+(1.0D0-WZ)*DDW(I,J,NZR,1)*(ZB(I,J,1)-
    *      ZB(IP,JP,2))
  POZ2PP=1.0D0+WZ*DDO(IP,JP,KP+NBP,2)*(ZB(I,J,1)-ZB(IP,JP,2))
  PWZ2PP=1.0D0+WZ*DDW(IP,JP,KP+NBP,2)*(ZB(I,J,1)-ZB(IP,JP,2))
ELSE
  POZ2PI=-1.0D0
  PWZ2PI=-1.0D0
  POZ2PP=1.0D0
  PWZ2PP=1.0D0
ENDIF
```

C-----
C Calculo das derivadas dos termos de acumulacao ACW e ACO em relacao a
C a Swi,j e Poi,j (nas demais variaveis essas derivadas se anulam)
C-----

```
DACWP=VPOR(I,J,K,L)*(CR*BW(I,J,K,L)*SW(I,J,K,L)+AUX1*
  *      SW(I,J,K,L)*DBW(I,J,K,L))/DT
DACWS=VPOR(I,J,K,L)*AUX1*BW(I,J,K,L)/DT
DACOP=VPOR(I,J,K,L)*(CR*BO(I,J,K,L)*(1.0D0-SW(I,J,K,L))+
  *      AUX1*(1.0D0-SW(I,J,K,L))*DBO(I,J,K,L))/DT
DACOS=-VPOR(I,J,K,L)*AUX1*BO(I,J,K,L)/DT
```

C-----
C Calculo das deriv. das transmissib. TOX e TWX em relacao a pressao
C-----

```
IF(L.EQ.1.AND.I.EQ.NX) THEN
  TOX2PI=(WRC-1.0D0)*TOX(I+1,J,K,L)*(DVO(I,J,K,L)/
    *      AUXVO-DBO(I,J,K,L)/AUXBO)
  TOX2PP=-WRC*TOX(I+1,J,K,L)*(DVO(IX,JX,KP,2)/
    *      AUXVO-DBO(IX,JX,KP,2)/AUXBO)
  TWX2PI=(WRC-1.0D0)*TWX(I+1,J,K,L)*(DVW(I,J,K,L)/
    *      AUXVW-DBW(I,J,K,L)/AUXBW)
  TWX2PP=-WRC*TWX(I+1,J,K,L)*(DVW(IX,JX,KP,2)/
    *      AUXVW-DBW(IX,JX,KP,2)/AUXBW)
```

C

```
IF(J.LE.NQL) THEN
  IV=1
ENDIF
IF(J.GT.NQL.AND.J.LE.NQL+NQS) THEN
  IV=2
ENDIF
IF(J.GT.NQL+NQS.AND.J.LE.NQS+2*NQL) THEN
  IV=3
ENDIF
IF(J.GT.NQS+2*NQL) THEN
```

```

      IV=4
    ENDIF
    IF(IV.EQ.1.OR.IV.EQ.2) THEN
      SDEOPI(IV,K)=SDEOPI(IV,K)-(TOX(I+1,J,K,1)*POX2PP+POTOX2*TOX2PP)
      SDEOSI(IV,K)=SDEOSI(IV,K)-(POTOX2*TOX2SP(I,J,K,1))
      SDEWPI(IV,K)=SDEWPI(IV,K)-(TWX(I+1,J,K,1)*PWX2PP+POTWX2*TWX2PP)
      SDEWSI(IV,K)=SDEWSI(IV,K)-(-TWX(I+1,J,K,1)*DPC(IX,JX,KP+K-1,2)+
*      POTWX2*TWX2SP(I,J,K,1))

```

```

    ENDIF
    IF(IV.EQ.3.OR.IV.EQ.4) THEN
      SDEOPI(IV,K)=SDEOPI(IV,K)-TOX(I+1,J,K,1)*POX2PP-POTOX2*TOX2PP
      SDEOSI(IV,K)=SDEOSI(IV,K)-POTOX2*TOX2SP(I,J,K,1)
      SDEWPI(IV,K)=SDEWPI(IV,K)-TWX(I+1,J,K,1)*PWX2PP-POTWX2*TWX2PP
      SDEWSI(IV,K)=SDEWSI(IV,K)+TWX(I+1,J,K,1)*DPC(IX,JX,KP+K-1,2)-
*      POTWX2*TWX2SP(I,J,K,1)

```

```

    ENDIF

```

```

      DFWPX=-TWX(I+1,J,K,1)*PWX2PI-POTWX2*TWX2PI
      DFWSX=-TWX(I+1,J,K,1)*DPC(I,J,K,1)-POTWX2*TWX2SI(I,J,K,1)
      DFOPX=-TOX(I+1,J,K,1)*POX2PI-POTOX2*TOX2PI
      DFOSX=-POTOX2*TOX2SI(I,J,K,1)

```

```

    ELSE

```

```

      TOX2PI=(W-1.0D0)*TOX(I+1,J,K,L)*(DVO(I,J,K,L)/
*      VOMX(I+1,J,K,L)-DBO(I,J,K,L)/BOMX(I+1,J,K,L))
      TOX2PP=-W*TOX(I+1,J,K,L)*(DVO(I+1,J,K,L)/
*      VOMX(I+1,J,K,L)-DBO(I+1,J,K,L)/BOMX(I+1,J,K,L))
      TWX2PI=(W-1.0D0)*TWX(I+1,J,K,L)*(DVW(I,J,K,L)/
*      VWMX(I+1,J,K,L)-DBW(I,J,K,L)/BWMX(I+1,J,K,L))
      TWX2PP=-W*TWX(I+1,J,K,L)*(DVW(I+1,J,K,L)/
*      VWMX(I+1,J,K,L)-DBW(I+1,J,K,L)/BWMX(I+1,J,K,L))

```

```

    ENDIF

```

```

      TOX1PA=IDR*((W-1.0D0)*TOX(I,J,K,L)*(DVO(I-1,J,K,L)/
*      VOMX(I,J,K,L)-DBO(I-1,J,K,L)/BOMX(I,J,K,L)))
      TOX1PI=IDR*(-W*TOX(I,J,K,L)*(DVO(I,J,K,L)/VOMX(I,J,K,L)-
*      DBO(I,J,K,L)/BOMX(I,J,K,L)))
      TWX1PA=IDR*((W-1.0D0)*TWX(I,J,K,L)*(DVW(I-1,J,K,L)/
*      VWMX(I,J,K,L)-DBW(I-1,J,K,L)/BWMX(I,J,K,L)))
      TWX1PI=IDR*(-W*TWX(I,J,K,L)*(DVW(I,J,K,L)/VWMX(I,J,K,L)-
*      DBW(I,J,K,L)/BWMX(I,J,K,L)))

```

C-----
C Calculo das deriv. das transmissib. TOY e TWY em relacao a pressao

```

C-----
      TOY1PA=(WY-1.0D0)*TOY(I,J,K,L)*(DVO(I,JME,K,L)/
*      VOMY(I,J,K,L)-DBO(I,JME,K,L)/BOMY(I,J,K,L))
      TOY1PI=-WY*TOY(I,J,K,L)*(DVO(I,J,K,L)/VOMY(I,J,K,L)-
*      DBO(I,J,K,L)/BOMY(I,J,K,L))
      TOY2PI=(WY-1.0D0)*TOY(I,JMA,K,L)*(DVO(I,J,K,L)/
*      VOMY(I,JMA,K,L)-DBO(I,J,K,L)/BOMY(I,JMA,K,L))
      TOY2PP=-WY*TOY(I,JMA,K,L)*(DVO(I,JMA,K,L)/VOMY(I,JMA,K,L)-
*      DBO(I,JMA,K,L)/BOMY(I,JMA,K,L))
      TWY1PA=(WY-1.0D0)*TWY(I,J,K,L)*(DVW(I,JME,K,L)/
*      VWMY(I,J,K,L)-DBW(I,JME,K,L)/BWMY(I,J,K,L))
      TWY1PI=-WY*TWY(I,J,K,L)*(DVW(I,J,K,L)/VWMY(I,J,K,L)-
*      DBW(I,J,K,L)/BWMY(I,J,K,L))
      TWY2PI=(WY-1.0D0)*TWY(I,JMA,K,L)*(DVW(I,J,K,L)/
*      VWMY(I,JMA,K,L)-DBW(I,J,K,L)/BWMY(I,JMA,K,L))
      TWY2PP=-WY*TWY(I,JMA,K,L)*(DVW(I,JMA,K,L)/VWMY(I,JMA,K,L)-
*      DBW(I,JMA,K,L)/BWMY(I,JMA,K,L))

```

C-----

C Calculo das deriv. das transmissib. TOZ e TWZ em relacao a pressao

```

C-----
      IF(L.EQ.1.AND.K.EQ.1.AND.KP.NE.1) THEN
        IV=5
        TOZ1PA=(WZ-1.0D0)*TOZ(I,J,1,1)*(DVO(IP,JP,KP-1,2)/
*          VOMZ(I,J,1,1)-DBO(IP,JP,KP-1,2)/BOMZ(I,J,1,1))
        TOZ1PI=-WZ*TOZ(I,J,1,1)*(DVO(I,J,1,1)/VOMZ(I,J,1,1)-
*          DBO(I,J,1,1)/BOMZ(I,J,1,1))
        TWZ1PA=(WZ-1.0D0)*TWZ(I,J,1,1)*(DVW(IP,JP,KP-1,2)/
*          VWMZ(I,J,1,1)-DBW(IP,JP,KP-1,2)/BWMZ(I,J,1,1))
        TWZ1PI=-WZ*TWZ(I,J,1,1)*(DVW(I,J,1,1)/VWMZ(I,J,1,1)-
*          DBW(I,J,1,1)/BWMZ(I,J,1,1))
        SDEOPI(IV,K)=SDEOPI(IV,K)+(TOZ(I,J,1,1)*POZ1PA+POTOZ1*TOZ1PA)
        SDEOSI(IV,K)=SDEOSI(IV,K)+(POTOZ1*TOZ1SA(I,J,1,1))
        SDEWPI(IV,K)=SDEWPI(IV,K)+(TWZ(I,J,1,1)*PWZ1PA+POTWZ1*TWZ1PA)
        SDEWSI(IV,K)=SDEWSI(IV,K)+(TWZ(I,J,1,1)*DPC(IP,JP,KP-1,2)+
*          POTWZ1*TWZ1SA(I,J,1,1))

C
        DFWPZ=TWZ(I,J,1,1)*PWZ1PI+POTWZ1*TWZ1PI
        DFOPZ=TOZ(I,J,1,1)*POZ1PI+POTOZ1*TOZ1PI
        DFOSZ=POTOZ1*TOZ1SI(I,J,1,1)
        DFWSZ=-TWZ(I,J,1,1)*DPC(I,J,1,1)+POTWZ1*TWZ1SI(I,J,1,1)
        ELSE
        TOZ1PA=(WZ-1.0D0)*TOZ(I,J,K,L)*(DVO(I,J,K-1,L)/
*          VOMZ(I,J,K,L)-DBO(I,J,K-1,L)/BOMZ(I,J,K,L))
        TOZ1PI=-WZ*TOZ(I,J,K,L)*(DVO(I,J,K,L)/VOMZ(I,J,K,L)-
*          DBO(I,J,K,L)/BOMZ(I,J,K,L))
        TWZ1PA=(WZ-1.0D0)*TWZ(I,J,K,L)*(DVW(I,J,K-1,L)/
*          VWMZ(I,J,K,L)-DBW(I,J,K-1,L)/BWMZ(I,J,K,L))
        TWZ1PI=-WZ*TWZ(I,J,K,L)*(DVW(I,J,K,L)/VWMZ(I,J,K,L)-
*          DBW(I,J,K,L)/BWMZ(I,J,K,L))
        ENDIF

C
        IF(L.EQ.1.AND.K.EQ.NZR.AND.KP+NBP-1.LT.NZC) THEN
          IV=6
          TOZ2PI=(WZ-1.0D0)*TOZ(I,J,NZR+1,1)*(DVO(I,J,NZR,1)/
*            VOMZ(I,J,NZR+1,1)-DBO(I,J,NZR,1)/BOMZ(I,J,NZR+1,1))
          TOZ2PP=-WZ*TOZ(I,J,NZR+1,1)*(DVO(IP,JP,KP+NBP,2)/
*            VOMZ(I,J,NZR+1,1)-DBO(IP,JP,KP+NBP,2)/BOMZ(I,J,NZR+1,1))
          TWZ2PI=(WZ-1.0D0)*TWZ(I,J,NZR+1,1)*(DVW(I,J,NZR,1)/
*            VWMZ(I,J,NZR+1,1)-DBW(I,J,NZR,1)/BWMZ(I,J,NZR+1,1))
          TWZ2PP=-WZ*TWZ(I,J,NZR+1,1)*(DVW(IP,JP,KP+NBP,2)/
*            VWMZ(I,J,NZR+1,1)-DBW(IP,JP,KP+NBP,2)/BWMZ(I,J,NZR+1,1))
          SDEOPI(IV,K)=SDEOPI(IV,K)-(TOZ(I,J,NZR+1,1)*POZ2PP+POTOZ2*TOZ2PP)
          SDEOSI(IV,K)=SDEOSI(IV,K)-(POTOZ2*TOZ2SI(I,J,NZR,1))
          SDEWPI(IV,K)=SDEWPI(IV,K)-(TWZ(I,J,NZR+1,1)*PWZ2PP+POTWZ2*TWZ2PP)
          SDEWSI(IV,K)=SDEWSI(IV,K)-(-TWZ(I,J,NZR+1,1)*DPC(IP,JP,KP+NBP,2)+
*            POTWZ2*TWZ2SI(I,J,NZR,1))

C
          DFWPZ=TWZ(I,J,NZR+1,1)*PWZ2PI+POTWZ2*TWZ2PI
          DFOPZ=TOZ(I,J,NZR+1,1)*POZ2PI+POTOZ2*TOZ2PI
          DFOSZ=POTOZ2*TOZ2SI(I,J,NZR,1)
          DFWSZ=TWZ(I,J,NZR,1)*DPC(I,J,NZR,1)+POTWZ2*TWZ2SI(I,J,NZR,1)
          DFWPZ=-DFWPZ
          DFOPZ=-DFOPZ
          DFOSZ=-DFWSZ
          DFWSZ=-DFWSZ
          ELSE
          TOZ2PI=(WZ-1.0D0)*TOZ(I,J,K+1,L)*(DVO(I,J,K,L)/
*            VOMZ(I,J,K+1,L)-DBO(I,J,K,L)/BOMZ(I,J,K+1,L))

```

```

      TOZ2PP=-WZ*TOZ(I,J,K+1,L)*(DVO(I,J,K+1,L)/VOMZ(I,J,K+1,L)-
*      DBO(I,J,K+1,L)/BOMZ(I,J,K+1,L))
      TWZ2PI=(WZ-1.0D0)*TWZ(I,J,K+1,L)*(DVW(I,J,K,L)/
*      VWMZ(I,J,K+1,L)-DBW(I,J,K,L)/BWMZ(I,J,K+1,L))
      TWZ2PP=-WZ*TWZ(I,J,K+1,L)*(DVW(I,J,K+1,L)/VWMZ(I,J,K+1,L)-
*      DBW(I,J,K+1,L)/BWMZ(I,J,K+1,L))
      ENDIF
C-----
C Determinacao das derivadas das funcoes de residuos em relacao as
C variaveis da malha : determinacao dos elementos da matriz jacobiana
C dFw/Poi,j,k
C-----
      CALL JAC (I,J,K,L)
C
15  CONTINUE
C 1000 CONTINUE
C      RETURN
C      END
C-----
C MONTAGEM DA MATRIZ JACOBIANA PARA GRID COMPOSTO
C MATRIZ COM BLOCOS 2X2
C-----
      SUBROUTINE JAC (I,J,K,L)
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      PARAMETER(NXM=08,NYM=05,NZM=04)
      COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*      NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
      COMMON/NBLC/A,D,NBC,NBR,NBT
      COMMON/ANULA/IX1,IX2,IY1,IY2,IZ1,IZ2,IDR,IDC,LBC,JMA,JME,IX,JX,LBR
      DIMENSION A(NXM*NYM*NZM*4,NXM*NYM*NZM*4),D(NXM*NYM*NZM*4)
C-----
C PARTE REFERENTE AO GRID CARTESIANO
C-----
      KP1=KP
      KPN=KP+NBP-1
      IF(L.EQ.2) THEN
C-----
C --1---- de (1,1,1) ate (ip-2,j,k)
C-----
      IF(I.LE.(IP-2)) THEN
          NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
          CALL MONTA (I,J,K,L,NCC)
      ENDIF
C-----
C --2---- de (ip-1,1,k) ate (ip-1,jp-1,k)
C-----
      IF(I.EQ.(IP-1).AND.J.LE.(JP-1)) THEN
          NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
          CALL MONTA (I,J,K,L,NCC)
      ENDIF
C-----
C --3---- de (ip-1,jp,1) ate (ip-1,jp,kp1-1)
C-----
      IF(I.EQ.(IP-1).AND.J.EQ.JP.AND.K.LT.KP1) THEN
          NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
          CALL MONTA (I,J,K,L,NCC)
      ENDIF
C-----
C --4---- de (ip-1,jp,kp1) ate (ip-1,jp,kpn)
C-----

```

```

      IF(I.EQ.(IP-1).AND.J.EQ.JP.AND.K.GE.KP1.AND.K.LE.KPN) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
C  zerar os elementos i+1
        IF(I.NE.NXC) THEN
C dfo/dpoi+1,j ,sgi+1,j, swi+1,j
        NH=NL+NYC*NZC*2
        A(NL,NH)=0.000
        A(NL,NH+1)=0.000
C dfw/dpoi+1,j ,sgi+1,j, swi+1,j
        A(NL+1,NH)=0.000
        A(NL+1,NH+1)=0.000
        ENDIF
      ENDIF
C-----
C --5---- de (ip-1,jp,kpn+1) ate (ip-1,jp,nzc)
C-----
      IF(I.EQ.(IP-1).AND.J.EQ.JP.AND.K.GT.KPN) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
C reduzir de 1 posicao os elementos i+1
        IF(I.NE.NXC) THEN
C dfo/dpoi+1,j ,sgi+1,j, swi+1,j
        NH=NL+NYC*NZC*2
        NP=NB*2
        A(NL,NH-NP)=A(NL,NH)
        A(NL,NH-NP+1)=A(NL,NH+1)
        A(NL,NH)=0.000
        A(NL,NH+1)=0.000
C dfw/dpoi+1,j ,sgi+1,j, swi+1,j
        A(NL+1,NH-NP)=A(NL+1,NH)
        A(NL+1,NH-NP+1)=A(NL+1,NH+1)
        A(NL+1,NH)=0.000
        A(NL+1,NH+1)=0.000
        ENDIF
      ENDIF
C-----
C --6---- de (ip-1,jp+1,1) ate (ip-1,nyc,nzc)
C-----
      IF(I.EQ.(IP-1).AND.J.GE.JP+1) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
C reduzir de 1 posicao os elementos i+1
        IF(I.NE.NXC) THEN
C dfo/dpoi+1,j ,sgi+1,j, swi+1,j
        NH=NL+NYC*NZC*2
        NP=NB*2
        A(NL,NH-NP)=A(NL,NH)
        A(NL,NH-NP+1)=A(NL,NH+1)
        A(NL,NH)=0.000
        A(NL,NH+1)=0.000
C dfw/dpoi+1,j ,sgi+1,j, swi+1,j
        A(NL+1,NH-NP)=A(NL+1,NH)
        A(NL+1,NH-NP+1)=A(NL+1,NH+1)
        A(NL+1,NH)=0.000
        A(NL+1,NH+1)=0.000
        ENDIF
      ENDIF

```

```

C-----
C --7---- de (ip,1,1) ate (ip,jp-2,nzc)
C-----
      IF(I.EQ.IP.AND.J.LE.(JP-2)) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
C reduzir de 1 posicao os elementos i+1
        IF(I.NE.NXC) THEN
C dfo/dpoi+1,j ,sgi+1,j, swi+1,j
          NH=NL+NYC*NZC*2
          NP=NBP*2
          A(NL,NH-NP)=A(NL,NH)
          A(NL,NH-NP+1)=A(NL,NH+1)
          A(NL,NH)=0.000
          A(NL,NH+1)=0.000
C dfw/dpoi+1,j ,sgi+1,j, swi+1,j
          A(NL+1,NH-NP)=A(NL+1,NH)
          A(NL+1,NH-NP+1)=A(NL+1,NH+1)
          A(NL+1,NH)=0.000
          A(NL+1,NH+1)=0.000
        ENDIF
      ENDIF
C-----
C --8---- de (ip,jp-1,1) ate (ip,jp-1,kp1-1)
C-----
      IF(I.EQ.IP.AND.J.EQ.(JP-1).AND.K.LT.KP1) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
C reduzir de 1 posicao os elementos i+1
        IF(I.NE.NXC) THEN
C dfo/dpoi+1,j ,sgi+1,j, swi+1,j
          NH=NL+NYC*NZC*2
          NP=NBP*2
          A(NL,NH-NP)=A(NL,NH)
          A(NL,NH-NP+1)=A(NL,NH+1)
          A(NL,NH)=0.000
          A(NL,NH+1)=0.000
C dfw/dpoi+1,j ,sgi+1,j, swi+1,j
          A(NL+1,NH-NP)=A(NL+1,NH)
          A(NL+1,NH-NP+1)=A(NL+1,NH+1)
          A(NL+1,NH)=0.000
          A(NL+1,NH+1)=0.000
        ENDIF
      ENDIF
C-----
C --9---- de (ip,jp-1,kp1) ate (ip,jp-1,kpn)
C-----
      IF(I.EQ.IP.AND.J.EQ.(JP-1).AND.K.GE.KP1.AND.K.LE.KPN) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
C zerar os elementos em j+1
C dfo/dpoi,j+1 ,sgi,j+1, swi,j+1
          NH=NL+NZC*2
          A(NL,NH)=0.000
          A(NL,NH+1)=0.000
C dfw/dpoi,j+1 ,sgi,j+1, swi,j+1
          A(NL+1,NH)=0.000

```

```

      A(NL+1,NH+1)=0.0D0
C reduzir de 1 posicao os elementos i+1
      IF(I.NE.NXC) THEN
C dfo/dpoi+1,j ,sgi+1,j, swi+1,j
      NH=NL+NYC*NZC*2
      NP=NBP*2
      A(NL,NH-NP)=A(NL,NH)
      A(NL,NH-NP+1)=A(NL,NH+1)
      A(NL,NH)=0.0D0
      A(NL,NH+1)=0.0D0
C dfw/dpoi+1,j ,sgi+1,j, swi+1,j
      A(NL+1,NH-NP)=A(NL+1,NH)
      A(NL+1,NH-NP+1)=A(NL+1,NH+1)
      A(NL+1,NH)=0.0D0
      A(NL+1,NH+1)=0.0D0
      ENDIF
C -----
C --10--- de (ip,jp-1,kpn+1) ate (ip,jp-1,nzc)
C -----
      IF(I.EQ.IP.AND.J.EQ.(JP-1).AND.K.GT.KPN) THEN
      NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
      CALL MONTA (I,J,K,L,NCC)
      NL=2*NCC-1
      NP=NBP*2
C reduzir de 1 posicao os elementos i+1
      IF(I.NE.NXC) THEN
C dfo/dpoi+1,j ,sgi+1,j, swi+1,j
      NH=NL+NYC*NZC*2
      A(NL,NH-NP)=A(NL,NH)
      A(NL,NH-NP+1)=A(NL,NH+1)
      A(NL,NH)=0.0D0
      A(NL,NH+1)=0.0D0
C dfw/dpoi+1,j ,sgi+1,j, swi+1,j
      A(NL+1,NH-NP)=A(NL+1,NH)
      A(NL+1,NH-NP+1)=A(NL+1,NH+1)
      A(NL+1,NH)=0.0D0
      A(NL+1,NH+1)=0.0D0
      ENDIF
C reduz os elementos em j+1
C dfo/dpoi,j+1 ,sgi,j+1, swi,j+1
      NH=NL+NZC*2
      A(NL,NH-NP)=A(NL,NH)
      A(NL,NH-NP+1)=A(NL,NH+1)
      A(NL,NH)=0.0D0
      A(NL,NH+1)=0.0D0
C dfw/dpoi,j+1 ,sgi,j+1, swi,j+1
      A(NL+1,NH-NP)=A(NL+1,NH)
      A(NL+1,NH-NP+1)=A(NL+1,NH+1)
      A(NL+1,NH)=0.0D0
      A(NL+1,NH+1)=0.0D0
      ENDIF
C -----
C --11--- de (ip,jp,1) ate (ip,jp,kp1-2)
C -----
      IF(I.EQ.IP.AND.J.EQ.JP.AND.K.LT.KP1-1) THEN
      NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
      CALL MONTA (I,J,K,L,NCC)
      NL=2*NCC-1

```

```

      NP=NB*2
C  reduzir de 1 posicao os elementos i+1
      IF(I.NE.NXC) THEN
C  dfo/dpoi+1,j ,sgi+1,j, swi+1,j
      NH=NL+NYC*NZC*2
      A(NL,NH-NP)=A(NL,NH)
      A(NL,NH-NP+1)=A(NL,NH+1)
      A(NL,NH)=0.000
      A(NL,NH+1)=0.000
C  dfw/dpoi+1,j ,sgi+1,j, swi+1,j
      A(NL+1,NH-NP)=A(NL+1,NH)
      A(NL+1,NH-NP+1)=A(NL+1,NH+1)
      A(NL+1,NH)=0.000
      A(NL+1,NH+1)=0.000
      ENDIF
C  reduz os elementos em j+1
      IF(J.NE.NYC) THEN
C  dfo/dpoi,j+1 ,sgi,j+1, swi,j+1
      NH=NL+NZC*2
      A(NL,NH-NP)=A(NL,NH)
      A(NL,NH-NP+1)=A(NL,NH+1)
      A(NL,NH)=0.000
      A(NL,NH+1)=0.000
C  dfw/dpoi,j+1 ,sgi,j+1, swi,j+1
      A(NL+1,NH-NP)=A(NL+1,NH)
      A(NL+1,NH-NP+1)=A(NL+1,NH+1)
      A(NL+1,NH)=0.000
      A(NL+1,NH+1)=0.000
      ENDIF
      ENDIF
C-----
C  --12--- (ip,jp,kp1-1)
C-----
      IF(I.EQ.IP.AND.J.EQ.JP.AND.K.EQ.(KP1-1)) THEN
      NCC=(I-1)*NYC*NZC+(J-1)*NZC+K
      CALL MONTA (I,J,K,L,NCC)
      NL=2*NCC-1
      NP=NB*2
C  reduzir de 1 posicao os elementos i+1
      IF(I.NE.NXC) THEN
C  dfo/dpoi+1,j ,sgi+1,j, swi+1,j
      NH=NL+NYC*NZC*2
      A(NL,NH-NP)=A(NL,NH)
      A(NL,NH-NP+1)=A(NL,NH+1)
      A(NL,NH)=0.000
      A(NL,NH+1)=0.000
C  dfw/dpoi+1,j ,sgi+1,j, swi+1,j
      A(NL+1,NH-NP)=A(NL+1,NH)
      A(NL+1,NH-NP+1)=A(NL+1,NH+1)
      A(NL+1,NH)=0.000
      A(NL+1,NH+1)=0.000
      ENDIF
C  reduz os elementos em j+1
      IF(J.NE.NYC) THEN
C  dfo/dpoi,j+1 ,sgi,j+1, swi,j+1
      NH=NL+NZC*2
      A(NL,NH-NP)=A(NL,NH)
      A(NL,NH-NP+1)=A(NL,NH+1)
      A(NL,NH)=0.000

```

```

      A(NL,NH+1)=0.000
C dfw/dpoi,j+1 ,sgi,j+1, swi,j+1
      A(NL+1,NH-NP)=A(NL+1,NH)
      A(NL+1,NH-NP+1)=A(NL+1,NH+1)
      A(NL+1,NH)=0.000
      A(NL+1,NH+1)=0.000
    ENDIF
C  anula os elementos em k+1
      IF(K.NE.NZC) THEN
C dfo/dpoi,j,k+1 ,sgi,j,k+1, swi,j,k+1
      A(NL,NL+2)=0.000
      A(NL,NL+3)=0.000
C dfw/dpoi,j,k+1 ,sgi,j,k+1, swi,j,k+1
      A(NL+1,NL+2)=0.000
      A(NL+1,NL+3)=0.000
    ENDIF
  ENDIF
C-----
C --13--- (ip,jp,kpn+1)
C-----
      IF(I.EQ.IP.AND.J.EQ.JP.AND.K.EQ.(KPN+1)) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
        NP=NBP*2
C avançar de 1 posicao os elementos i-1
        IF(I.NE.1) THEN
C dfo/dpoi-1,j ,sgi-1,j, swi-1,j
          NH=NL-NYC*NZC*2
          A(NL,NH+NP)=A(NL,NH)
          A(NL,NH+NP+1)=A(NL,NH+1)
          A(NL,NH)=0.000
          A(NL,NH+1)=0.000
C dfw/dpoi-1,j ,sgi-1,j, swi-1,j
          A(NL+1,NH+NP)=A(NL+1,NH)
          A(NL+1,NH+NP+1)=A(NL+1,NH+1)
          A(NL+1,NH)=0.000
          A(NL+1,NH+1)=0.000
        ENDIF
C avançar os elementos em j-1
        IF(J.NE.1) THEN
C dfo/dpoi,j-1 ,sgi,j-1, swi,j-1
          NH=NL-NZC*2
          A(NL,NH+NP)=A(NL,NH)
          A(NL,NH+NP+1)=A(NL,NH+1)
          A(NL,NH)=0.000
          A(NL,NH+1)=0.000
C dfw/dpoi,j-1 ,sgi,j-1, swi,j-1
          A(NL+1,NH+NP)=A(NL+1,NH)
          A(NL+1,NH+NP+1)=A(NL+1,NH+1)
          A(NL+1,NH)=0.000
          A(NL+1,NH+1)=0.000
        ENDIF
C  anula os elementos em k-1
        IF(K.NE.1) THEN
C dfo/dpoi,j,k-1 ,sgi,j,k-1, swi,j,k-1
          A(NL,NL-2)=0.000
          A(NL,NL-1)=0.000
C dfw/dpoi,j,k-1 ,sgi,j,k-1, swi,j,k-1

```

```

      A(NL+1,NL-2)=0.0D0
      A(NL+1,NL-1)=0.0D0
    ENDIF
  ENDIF
C-----
C --14--- (ip,jp,kpn+2) ate (ip,jp,nzc)
C-----
      IF(I.EQ.IP.AND.J.EQ.JP.AND.K.GT.(KPN+1)) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
        NP=NBP*2
C avançar de 1 posicao os elementos i-1
        IF(I.NE.1) THEN
C dfo/dpoi-1,j ,sgi-1,j, swi-1,j
          NH=NL-NYC*NZC*2
          A(NL,NH+NP)=A(NL,NH)
          A(NL,NH+NP+1)=A(NL,NH+1)
          A(NL,NH)=0.0D0
          A(NL,NH+1)=0.0D0
C dfw/dpoi-1,j ,sgi-1,j, swi-1,j
          A(NL+1,NH+NP)=A(NL+1,NH)
          A(NL+1,NH+NP+1)=A(NL+1,NH+1)
          A(NL+1,NH)=0.0D0
          A(NL+1,NH+1)=0.0D0
        ENDIF
C avançar os elementos em j-1
        IF(J.NE.1) THEN
C dfo/dpoi,j-1 ,sgi,j-1, swi,j-1
          NH=NL-NZC*2
          A(NL,NH+NP)=A(NL,NH)
          A(NL,NH+NP+1)=A(NL,NH+1)
          A(NL,NH)=0.0D0
          A(NL,NH+1)=0.0D0
C dfw/dpoi,j-1 ,sgi,j-1, swi,j-1
          A(NL+1,NH+NP)=A(NL+1,NH)
          A(NL+1,NH+NP+1)=A(NL+1,NH+1)
          A(NL+1,NH)=0.0D0
          A(NL+1,NH+1)=0.0D0
        ENDIF
      ENDIF
C-----
C --15--- (ip,jp+1,1) ate (ip,jp+1,kp1-1)
C-----
      IF(I.EQ.IP.AND.J.EQ.(JP+1).AND.K.LT.KP1) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
        NP=NBP*2
C avançar de 1 posicao os elementos i-1
        IF(I.NE.1) THEN
C dfo/dpoi-1,j ,sgi-1,j, swi-1,j
          NH=NL-NYC*NZC*2
          A(NL,NH+NP)=A(NL,NH)
          A(NL,NH+NP+1)=A(NL,NH+1)
          A(NL,NH)=0.0D0
          A(NL,NH+1)=0.0D0
C dfw/dpoi-1,j ,sgi-1,j, swi-1,j
          A(NL+1,NH+NP)=A(NL+1,NH)

```

```

      A(NL+1,NH+NP+1)=A(NL+1,NH+1)
      A(NL+1,NH)=0.0DO
      A(NL+1,NH+1)=0.0DO
    ENDIF
C  avançar os elementos em j-1
      IF(J.NE.1) THEN
C  dfo/dpoi,j-1 ,sgi,j-1, swi,j-1
      NH=NL-NZC*2
      A(NL,NH+NP)=A(NL,NH)
      A(NL,NH+NP+1)=A(NL,NH+1)
      A(NL,NH)=0.0DO
      A(NL,NH+1)=0.0DO
C  dfw/dpoi,j-1 ,sgi,j-1, swi,j-1
      A(NL+1,NH+NP)=A(NL+1,NH)
      A(NL+1,NH+NP+1)=A(NL+1,NH+1)
      A(NL+1,NH)=0.0DO
      A(NL+1,NH+1)=0.0DO
    ENDIF
  ENDIF
C-----
C --16--- (ip,jp+1,kp1) ate (ip,jp+1,kpn)
C-----
      IF(I.EQ.IP.AND.J.EQ.(JP+1).AND.K.GE.KP1.AND.K.LE.KPN) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
        NP=NBP*2
C  avançar de 1 posicao os elementos i-1
        IF(I.NE.1) THEN
C  dfo/dpoi-1,j ,sgi-1,j, swi-1,j
        NH=NL-NYC*NZC*2
        A(NL,NH+NP)=A(NL,NH)
        A(NL,NH+NP+1)=A(NL,NH+1)
        A(NL,NH)=0.0DO
        A(NL,NH+1)=0.0DO
C  dfw/dpoi-1,j ,sgi-1,j, swi-1,j
        A(NL+1,NH+NP)=A(NL+1,NH)
        A(NL+1,NH+NP+1)=A(NL+1,NH+1)
        A(NL+1,NH)=0.0DO
        A(NL+1,NH+1)=0.0DO
        ENDIF
C  anula os elementos em j-1
        IF(J.NE.1) THEN
C  dfo/dpoi,j-1 ,sgi,j-1, swi,j-1
        NH=NL-NZC*2
        A(NL,NH)=0.0DO
        A(NL,NH+1)=0.0DO
C  dfw/dpoi,j-1 ,sgi,j-1, swi,j-1
        A(NL+1,NH)=0.0DO
        A(NL+1,NH+1)=0.0DO
        ENDIF
      ENDIF
C-----
C --17--- (ip,jp+1,kpn+1) ate (ip,jp+1,nzc)
C-----
      IF(I.EQ.IP.AND.J.EQ.(JP+1).AND.K.GT.KPN) THEN
        NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
        CALL MONTA (I,J,K,L,NCC)
        NL=2*NCC-1
        NP=NBP*2

```

C avançar de 1 posicao os elementos i-1

IF(I.NE.1) THEN

C dfo/dpoi-1,j ,sgi-1,j, swi-1,j

NH=NL-NYC*NZC*2

A(NL,NH+NP)=A(NL,NH)

A(NL,NH+NP+1)=A(NL,NH+1)

A(NL,NH)=0.0DO

A(NL,NH+1)=0.0DO

C dfw/dpoi-1,j ,sgi-1,j, swi-1,j

A(NL+1,NH+NP)=A(NL+1,NH)

A(NL+1,NH+NP+1)=A(NL+1,NH+1)

A(NL+1,NH)=0.0DO

A(NL+1,NH+1)=0.0DO

ENDIF

ENDIF

C --17a-- (ip,jp+2,1) ate (ip,nyc,nzc)

C-----

IF(I.EQ.IP.AND.J.GE.(JP+2)) THEN

NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP

CALL MONTA (I,J,K,L,NCC)

NL=2*NCC-1

C avançar de 1 posicao os elementos i-1

C dfo/dpoi-1,j ,sgi-1,j, swi-1,j

IF(I.NE.1) THEN

NH=NL-NYC*NZC*2

NP=NBP*2

A(NL,NH+NP)=A(NL,NH)

A(NL,NH+NP+1)=A(NL,NH+1)

A(NL,NH)=0.0DO

A(NL,NH+1)=0.0DO

C dfw/dpoi-1,j ,sgi-1,j, swi-1,j

A(NL+1,NH+NP)=A(NL+1,NH)

A(NL+1,NH+NP+1)=A(NL+1,NH+1)

A(NL+1,NH)=0.0DO

A(NL+1,NH+1)=0.0DO

ENDIF

ENDIF

C --18--- (ip+1,1,1) ate (ip+1,nyc,nzc)

C-----

IF(I.EQ.(IP+1).AND.J.LE.(JP-1)) THEN

NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP

CALL MONTA (I,J,K,L,NCC)

NL=2*NCC-1

C avançar de 1 posicao os elementos i-1

IF(I.NE.1) THEN

C dfo/dpoi-1,j ,sgi-1,j, swi-1,j

NH=NL-NYC*NZC*2

NP=NBP*2

A(NL,NH+NP)=A(NL,NH)

A(NL,NH+NP+1)=A(NL,NH+1)

A(NL,NH)=0.0DO

A(NL,NH+1)=0.0DO

C dfw/dpoi-1,j ,sgi-1,j, swi-1,j

A(NL+1,NH+NP)=A(NL+1,NH)

A(NL+1,NH+NP+1)=A(NL+1,NH+1)

A(NL+1,NH)=0.0DO

A(NL+1,NH+1)=0.0DO

ENDIF

```

      ENDIF
C-----
C --19--- (ip+1,jp,1) ate (ip+1,jp,kp1-1)
C-----
      IF(I.EQ.(IP+1).AND.J.EQ.JP.AND.K.LT.KP1) THEN
          NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
          CALL MONTA (I,J,K,L,NCC)
          NL=2*NCC-1
C avançar de 1 posicao os elementos i-1
          IF(I.NE.1) THEN
C dfo/dpoi-1,j ,sgi-1,j, swi-1,j
              NH=NL-NYC*NZC*2
              NP=NBP*2
              A(NL,NH+NP)=A(NL,NH)
              A(NL,NH+NP+1)=A(NL,NH+1)
              A(NL,NH)=0.0D0
              A(NL,NH+1)=0.0D0
C dfw/dpoi-1,j ,sgi-1,j, swi-1,j
              A(NL+1,NH+NP)=A(NL+1,NH)
              A(NL+1,NH+NP+1)=A(NL+1,NH+1)
              A(NL+1,NH)=0.0D0
              A(NL+1,NH+1)=0.0D0
          ENDIF
      ENDIF
C-----
C --20--- (ip+1,jp,kp1) ate (ip+1,jp,kpn)
C-----
      IF(I.EQ.(IP+1).AND.J.EQ.JP.AND.K.GE.KP1.AND.K.LE.KPN) THEN
          NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
          CALL MONTA (I,J,K,L,NCC)
          NL=2*NCC-1
C anula os elementos i-1
          IF(I.NE.1) THEN
C dfo/dpoi-1,j ,sgi-1,j, swi-1,j
              NH=NL-NYC*NZC*2
              A(NL,NH)=0.0D0
              A(NL,NH+1)=0.0D0
C dfw/dpoi-1,j ,sgi-1,j, swi-1,j
              A(NL+1,NH)=0.0D0
              A(NL+1,NH+1)=0.0D0
          ENDIF
      ENDIF
C-----
C --21--- (ip+1,jp,kpn+1) ate (ip+1,jp,nzc)
C-----
      IF(I.EQ.(IP+1).AND.J.EQ.JP.AND.K.GT.KPN) THEN
          NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
          CALL MONTA (I,J,K,L,NCC)
      ENDIF
C-----
C --22--- (ip+1,jp+1,1) ate (ip+1,nyc,nzc)
C-----
      IF(I.EQ.(IP+1).AND.J.GE.(JP+1)) THEN
          NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
          CALL MONTA (I,J,K,L,NCC)
      ENDIF
C-----
C --23--- (ip+2,1,1) ate (nxc,nyc,nzc)
C-----
      IF(I.GE.(IP+2)) THEN
          NCC=(I-1)*NYC*NZC+(J-1)*NZC+K-NBP
          CALL MONTA (I,J,K,L,NCC)
      ENDIF

```

```

      ENDIF
      ENDIF
C-----
C
C PARTE REFERENTE AD GRID RADIAL
C-----
C
      IF(L.EQ.1) THEN
C
          NCR=NBC+(I-1)*NYR*NZR+(J-1)*NZR+K
          CALL MONTA (I,J,K,L,NCR)
      ENDIF
C-----
C
C PARTE REFERENTE AS INTERFACES DO RADIAL COM O CARTESIANO
C-----
C
      IF(L.EQ.1.AND.I.EQ.NXR) CALL MONTAIN (I,J,K,L,1)
      IF(L.EQ.1.AND.K.EQ.1.AND.KP.NE.1) CALL MONTAIN (I,J,K,L,2)
      IF(L.EQ.1.AND.K.EQ.NZR.AND.KP+NBP-1.LT.NZC) CALL MONTAIN(I,J,K,L,3)
C
      RETURN
      END
C-----
C
C MONTAGEM DA MATRIZ JACOBIANA
C-----
C
      SUBROUTINE MONTA (I,J,K,L,NC)
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      PARAMETER(NXM=08,NYM=05,NZM=04)
      COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*          NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
      COMMON/NBLC/A,D,NBC,NBR,NBT
      COMMON/PETR/KRO,KRW,DKRO,DKRW,PC,DPC
      COMMON/DTRANS/TOX,TWX,TOY,TWY,TOZ,TWZ,
*          TOX1SA,TOX1SI,TWX1SA,TWX1SI,
*          TOX2SI,TOX2SP,TWX2SI,TWX2SP,
*          TOY1SA,TOY1SI,TWY1SA,TWY1SI,
*          TOY2SI,TOY2SP,TWY2SI,TWY2SP,
*          TOZ1SA,TOZ1SI,TWZ1SA,TWZ1SI,
*          TOZ2SI,TOZ2SP,TWZ2SI,TWZ2SP
      COMMON/VAZAO/QO,QOT,QG,QGT,QW,QWT,DQOP,DQOS,
*          DQWP,DQWS,RWO,RGO,PWF,TEMPO
      COMMON/ANULA/IX1,IX2,IY1,IY2,IZ1,IZ2,IDR,IDC,LBC,JMA,JME,IX,JX,LBR
      COMMON/ACU/ACO,ACW,DACOS,DACWP,DACWS
      COMMON/POT/POTOX1,POTOX2,POTOY1,POTOY2,POTOZ1,POTOZ2,
*          POTWX1,POTWX2,POTWY1,POTWY2,POTWZ1,POTWZ2,
*          POX1PA,POX1PI,POY1PA,POY1PI,POZ1PA,POZ1PI,
*          POX2PI,POX2PP,POY2PI,POY2PP,POZ2PI,POZ2PP,
*          PWX1PA,PWX1PI,PWY1PA,PWY1PI,PWZ1PA,PWZ1PI,
*          PWX2PI,PWX2PP,PWY2PI,PWY2PP,PWZ2PI,PWZ2PP,
*          TOX1PA,TOX1PI,TOY1PA,TOY1PI,TOZ1PA,TOZ1PI,
*          TOX2PI,TOX2PP,TOY2PI,TOY2PP,TOZ2PI,TOZ2PP,
*          TWX1PA,TWX1PI,TWY1PA,TWY1PI,TWZ1PA,TWZ1PI,
*          TWX2PI,TWX2PP,TWY2PI,TWY2PP,TWZ2PI,TWZ2PP
      COMMON/SDEOW/STPO,STPW,SDEOPI,SDEOSI,SDEWPI,SDEWSI,
*          DFOPX,DFOSX,DFWPX,DFWSX,DFOPZ,DFOSZ,DFWPZ,DFWSZ
      DIMENSION DKRO(NXM,NYM,NZM,2),DKRW(NXM,NYM,NZM,2),
*          PC(0:NXM+1,0:NYM+1,0:NZM+1,2),DPC(0:NXM+1,0:NYM+1,0:NZM+1,2)
      REAL*8 KRO(NXM,NYM,NZM,2),KRW(NXM,NYM,NZM,2)
      DIMENSION TOX(NXM+1,NYM,NZM,2),TWX(NXM+1,NYM,NZM,2),
*          TOX1SA(NXM,NYM,NZM,2),TOX1SI(NXM,NYM,NZM,2),
*          TWX1SA(NXM,NYM,NZM,2),TWX1SI(NXM,NYM,NZM,2),

```

```

*      TOX2SI(NXM,NYM,NZM,2),TOX2SP(NXM,NYM,NZM,2),
*      TWX2SI(NXM,NYM,NZM,2),TWX2SP(NXM,NYM,NZM,2)
DIMENSION TOY(NXM,NYM+1,NZM,2),TWY(NXM,NYM+1,NZM,2),
*      TOY1SA(NXM,NYM,NZM,2),TOY1SI(NXM,NYM,NZM,2),
*      TWY1SA(NXM,NYM,NZM,2),TWY1SI(NXM,NYM,NZM,2),
*      TOY2SI(NXM,NYM,NZM,2),TOY2SP(NXM,NYM,NZM,2),
*      TWY2SI(NXM,NYM,NZM,2),TWY2SP(NXM,NYM,NZM,2)
DIMENSION TOZ(NXM,NYM,NZM+1,2),TWZ(NXM,NYM,NZM+1,2),
*      TOZ1SA(NXM,NYM,NZM,2),TOZ1SI(NXM,NYM,NZM,2),
*      TWZ1SA(NXM,NYM,NZM,2),TWZ1SI(NXM,NYM,NZM,2),
*      TOZ2SI(NXM,NYM,NZM,2),TOZ2SP(NXM,NYM,NZM,2),
*      TWZ2SI(NXM,NYM,NZM,2),TWZ2SP(NXM,NYM,NZM,2)
DIMENSION QO(NYM,NZM),QG(NYM,NZM),QW(NYM,NZM),DQOP(NYM,NZM),
*      DQOS(NYM,NZM),DQWP(NYM,NZM),DQWS(NYM,NZM)
DIMENSION A(NXM*NYM*NZM*4,NXM*NYM*NZM*4),D(NXM*NYM*NZM*4)
DIMENSION STPO(0:6,NZM),STPW(0:6,NZM)
C      STPO--> Somatorio de Trans. X Pot. do Oleo
DIMENSION SDEOPI(0:6,NZM),SDEWPI(0:6,NZM),SDEOSI(0:6,NZM),
*      SDEWSI(0:6,NZM)
C      SDEOP--> Som. da Derivada da Eq. do Oleo em rel. a Pres.
      NX=NXC
      NY=NYC
      NZ=NZC
      IF(L.EQ.1) THEN
        NX=NXR
        NY=NYR
        NZ=NZR
      ENDIF
C      NL=2*NC-1
C      dfw/dpoi,j,sgij,swij
      A(NL,NL)=IX2*(POTWX2*TWX2PI+TWX(I+1,J,K,L)*PWX2PI)-
*      IX1*(POTWX1*TWX1PI+TWX(I,J,K,L)*PWX1PI)+
*      IY2*(POTWY2*TWY2PI+TWY(I,JMA,K,L)*PWY2PI)-
*      IY1*(POTWY1*TWY1PI+TWY(I,J,K,L)*PWY1PI)+
*      IZ2*(POTWZ2*TWZ2PI+TWZ(I,J,K+1,L)*PWZ2PI)-
*      IZ1*(POTWZ1*TWZ1PI+TWZ(I,J,K,L)*PWZ1PI)-
*      DACWP-(1-IDR)*DQWP(J,K)+IDC*SDEWPI(LBC,LBR)
C      dFw/dSwi,j,k
      A(NL,NL+1)=
*      IX2*(POTWX2*TWX2SI(I,J,K,L)+TWX(I+1,J,K,L)*DPC(I,J,K,L))-
*      IX1*(POTWX1*TWX1SI(I,J,K,L)-IDR*TWX(I,J,K,L)*DPC(I,J,K,L))+
*      IY2*(POTWY2*TWY2SI(I,J,K,L)+TWY(I,JMA,K,L)*DPC(I,J,K,L))-
*      IY1*(POTWY1*TWY1SI(I,J,K,L)-TWY(I,J,K,L)*DPC(I,J,K,L))+
*      IZ2*(POTWZ2*TWZ2SI(I,J,K,L)+TWZ(I,J,K+1,L)*DPC(I,J,K,L))-
*      IZ1*(POTWZ1*TWZ1SI(I,J,K,L)-TWZ(I,J,K,L)*DPC(I,J,K,L))-
*      DACWS-(1-IDR)*DQWS(J,K)+IDC*SDEWSI(LBC,LBR)
C      dFo/dPoi,j,k
      A(NL+1,NL)=IX2*(POTOX2*TOX2PI+TOX(I+1,J,K,L)*POX2PI)-
*      IX1*(POTOX1*TOX1PI+TOX(I,J,K,L)*POX1PI)+
*      IY2*(POTOY2*TOY2PI+TOY(I,JMA,K,L)*POY2PI)-
*      IY1*(POTOY1*TOY1PI+TOY(I,J,K,L)*POY1PI)+
*      IZ2*(POTOZ2*TOZ2PI+TOZ(I,J,K+1,L)*POZ2PI)-
*      IZ1*(POTOZ1*TOZ1PI+TOZ(I,J,K,L)*POZ1PI)-
*      DACOP-(1-IDR)*DQOP(J,K)+IDC*SDEOPI(LBC,LBR)
C      dFo/dSwi,j,k
      A(NL+1,NL+1)=IX2*(POTOX2*TOX2SI(I,J,K,L))-
*      IX1*(POTOX1*TOX1SI(I,J,K,L))+
*      IY2*(POTOY2*TOY2SI(I,J,K,L))-

```

```

*          IY1*(POTOY1*TOY1SI(I,J,K,L))+
*          IZ2*(POTOZ2*TOZ2SI(I,J,K,L))-
*          IZ1*(POTOZ1*TOZ1SI(I,J,K,L))-
*          DACOS-(1-IDR)*DQOS(J,K)+IDC*SDEOSI(LBC,LBR)
C
  IF(I.NE.1) THEN
    NH=NY*NZ*2
C dFw/dPoi-1,j,k
    A(NL,NL-NH)=-POTWX1*TWX1PA-TWX(I,J,K,L)*PWX1PA
C dFw/dSwi-1,j,k
    A(NL,NL-NH+1)=-POTWX1*TWX1SA(I,J,K,L)-
    *          IDR*(TWX(I,J,K,L)*DPC(I-1,J,K,L))
C dFo/dPoi-1,j,k
    A(NL+1,NL-NH)=-POTOX1*TOX1PA-TOX(I,J,K,L)*POX1PA
C dFo/dSwi-1,j,k
    A(NL+1,NL-NH+1)=-POTOX1*TOX1SA(I,J,K,L)
  ENDIF
C
  IF(I.NE.NX) THEN
    NH=NY*NZ*2
C dFw/dPoi+1,j,k
    A(NL,NL+NH)=POTWX2*TWX2PP+TWX(I+1,J,K,L)*PWX2PP
C dFw/dSwi+1,j,k
    A(NL,NL+NH+1)=POTWX2*TWX2SP(I,J,K,L)-
    *          TWX(I+1,J,K,L)*DPC(I+1,J,K,L)
C dFo/dPoi+1,j,k
    A(NL+1,NL+NH)=POTOX2*TOX2PP+TOX(I+1,J,K,L)*POX2PP
C dFo/dSwi+1,j,k
    A(NL+1,NL+NH+1)=POTOX2*TOX2SP(I,J,K,L)
  ENDIF
C
  IF(J.NE.1) THEN
    NH=NZ*2
C dFw/dPoi,j-1,k
    A(NL,NL-NH)=-POTWY1*TWY1PA-TWY(I,J,K,L)*PWY1PA
C dFw/dSwi,j-1,k
    A(NL,NL-NH+1)=-POTWY1*TWY1SA(I,J,K,L)-TWY(I,J,K,L)*
    *          DPC(I,JME,K,L)
C dFo/dPoi,j-1,k
    A(NL+1,NL-NH)=-POTOY1*TOY1PA-TOY(I,J,K,L)*POY1PA
C dFo/dSwi,j-1,k
    A(NL+1,NL-NH+1)=-POTOY1*TOY1SA(I,J,K,L)
  ENDIF
C
  IF(J.NE.NY) THEN
    NH=NZ*2
C dFw/dPoi,j+1,k
    A(NL,NL+NH)=POTWY2*TWY2PP+TWY(I,JMA,K,L)*PWY2PP
C dFw/dSwi,j+1,k
    A(NL,NL+NH+1)=POTWY2*TWY2SP(I,J,K,L)-
    *          TWY(I,JMA,K,L)*DPC(I,JMA,K,L)
C dFo/dPoi,j+1,k
    A(NL+1,NL+NH)=POTOY2*TOY2PP+TOY(I,JMA,K,L)*POY2PP
C dFo/dSwi,j+1,k
    A(NL+1,NL+NH+1)=POTOY2*TOY2SP(I,J,K,L)
  ENDIF
C So radial
  IF(L.EQ.1.AND.J.EQ.1) THEN

```

```

      NH=(NY*NZ-NZ)*2
C dFw/dPoi,j-1,k
      A(NL,NL+NH)=-POTWY1*TWY1PA-TWY(I,J,K,L)*PWY1PA
C dFw/dSwi,j-1,k
      A(NL,NL+NH+1)=-POTWY1*TWY1SA(I,J,K,L)-TWY(I,J,K,L)*
      * DPC(I,JME,K,L)
C dFo/dPoi,j-1,k
      A(NL+1,NL+NH)=-POTOY1*TOY1PA-TOY(I,J,K,L)*POY1PA
C dFo/dSwi,j-1,k
      A(NL+1,NL+NH+1)=-POTOY1*TOY1SA(I,J,K,L)
      ENDIF
C
C So radial
      IF(L.EQ.1.AND.J.EQ.NY) THEN
      NH=(NY*NZ-NZ)*2
C dFw/dPoi,j+1,k
      A(NL,NL-NH)=POTWY2*TWY2PP+TWY(I,JMA,K,L)*PWY2PP
C dFw/dSwi,j+1,k
      A(NL,NL-NH+1)=POTWY2*TWY2SP(I,J,K,L)-
      * TWY(I,JMA,K,L)*DPC(I,JMA,K,L)
C dFo/dPoi,j+1,k
      A(NL+1,NL-NH)=POTOY2*TOY2PP+TOY(I,JMA,K,L)*POY2PP
C dFo/dSwi,j+1,k
      A(NL+1,NL-NH+1)=POTOY2*TOY2SP(I,J,K,L)
      ENDIF
C
C Celulas atraz da celula i,j,k
C
      IF(K.NE.1) THEN
C dFw/dPoi,j,k-1
      A(NL,NL-2)=-POTWZ1*TWZ1PA-TWZ(I,J,K,L)*PWZ1PA
C dFw/dSwi,j,k-1
      A(NL,NL-1)=-POTWZ1*TWZ1SA(I,J,K,L)-TWZ(I,J,K,L)*
      * DPC(I,J,K-1,L)
C dFo/dPoi,j,k-1
      A(NL+1,NL-2)=-POTOZ1*TOZ1PA-TOZ(I,J,K,L)*POZ1PA
C dFo/dSwi,j,k-1
      A(NL+1,NL-1)=-POTOZ1*TOZ1SA(I,J,K,L)
      ENDIF
C
C Celulas a frente da celula i,j,k
C
      IF(K.NE.NZ) THEN
C dFw/dPoi,j,k+1
      A(NL,NL+2)=POTWZ2*TWZ2PP+TWZ(I,J,K+1,L)*PWZ2PP
C dFw/dSwi,j,k+1
      A(NL,NL+3)=POTWZ2*TWZ2SP(I,J,K,L)-
      * TWZ(I,J,K+1,L)*DPC(I,J,K+1,L)
C dFo/dPoi,j,k+1
      A(NL+1,NL+2)=POTOZ2*TOZ2PP+TOZ(I,J,K+1,L)*POZ2PP
C dFo/dSwi,j,k+1
      A(NL+1,NL+3)=POTOZ2*TOZ2SP(I,J,K,L)
      ENDIF
C
      RETURN
      END
C-----
C PARTE REFERENTE AS INTERFACES DO RADIAL COM O CARTESIANO

```

```

C-----
C      SUBROUTINE MONTAIN (I,J,K,L,IND)
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      PARAMETER(NXM=08,NYM=05,NZM=04)
      COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*          NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
      COMMON/NBLC/A,D,NBC,NBR,NBT
      COMMON/PETR/KRO,KRW,DKRO,DKRW,PC,DPC
      COMMON/DTRANS/TOX,TWX,TOY,TWY,TOZ,TWZ,
*          TOX1SA,TOX1SI,TWX1SA,TWX1SI,
*          TOX2SI,TOX2SP,TWX2SI,TWX2SP,
*          TOY1SA,TOY1SI,TWY1SA,TWY1SI,
*          TOY2SI,TOY2SP,TWY2SI,TWY2SP,
*          TOZ1SA,TOZ1SI,TWZ1SA,TWZ1SI,
*          TOZ2SI,TOZ2SP,TWZ2SI,TWZ2SP
      COMMON/VAZAO/QO,QOT,QG,QGT,QW,QWT,DQOP,DQOS,
*          DQWP,DQWS,RWO,RGO,PWF,TEMPO
      COMMON/ANULA/IX1,IX2,IY1,IY2,IZ1,IZ2,IDR,IDC,LBC,JMA,JME,IX,JX,LBR
      COMMON/ACU/ACO,ACW,DACOP,DACOS,DACWP,DACWS
      COMMON/POT/POTOX1,POTOX2,POTY1,POTY2,POTZ1,POTZ2,
*          POTWX1,POTWX2,POTWY1,POTWY2,POTWZ1,POTWZ2,
*          POX1PA,POX1PI,POY1PA,POY1PI,POZ1PA,POZ1PI,
*          POX2PI,POX2PP,POY2PI,POY2PP,POZ2PI,POZ2PP,
*          PWX1PA,PWX1PI,PWY1PA,PWY1PI,PWZ1PA,PWZ1PI,
*          PWX2PI,PWX2PP,PWY2PI,PWY2PP,PWZ2PI,PWZ2PP,
*          TOX1PA,TOX1PI,TOY1PA,TOY1PI,TOZ1PA,TOZ1PI,
*          TOX2PI,TOX2PP,TOY2PI,TOY2PP,TOZ2PI,TOZ2PP,
*          TWX1PA,TWX1PI,TWY1PA,TWY1PI,TWZ1PA,TWZ1PI,
*          TWX2PI,TWX2PP,TWY2PI,TWY2PP,TWZ2PI,TWZ2PP
      COMMON/SDEOW/STPO,STPW,SDEOPI,SDEOSI,SDEWPI,SDEWSI,
*          DFOPX,DFOSX,DFWPX,DFWSX,DFOPZ,DFOSZ,DFWPZ,DFWSZ
      DIMENSION DKRO(NXM,NYM,NZM,2),DKRW(NXM,NYM,NZM,2),
*          PC(0:NXM+1,0:NYM+1,0:NZM+1,2),DPC(0:NXM+1,0:NYM+1,0:NZM+1,2)
      REAL*8 KRO(NXM,NYM,NZM,2),KRW(NXM,NYM,NZM,2)
      DIMENSION TOX(NXM+1,NYM,NZM,2),TWX(NXM+1,NYM,NZM,2),
*          TOX1SA(NXM,NYM,NZM,2),TOX1SI(NXM,NYM,NZM,2),
*          TWX1SA(NXM,NYM,NZM,2),TWX1SI(NXM,NYM,NZM,2),
*          TOX2SI(NXM,NYM,NZM,2),TOX2SP(NXM,NYM,NZM,2),
*          TWX2SI(NXM,NYM,NZM,2),TWX2SP(NXM,NYM,NZM,2)
      DIMENSION TOY(NXM,NYM+1,NZM,2),TWY(NXM,NYM+1,NZM,2),
*          TOY1SA(NXM,NYM,NZM,2),TOY1SI(NXM,NYM,NZM,2),
*          TWY1SA(NXM,NYM,NZM,2),TWY1SI(NXM,NYM,NZM,2),
*          TOY2SI(NXM,NYM,NZM,2),TOY2SP(NXM,NYM,NZM,2),
*          TWY2SI(NXM,NYM,NZM,2),TWY2SP(NXM,NYM,NZM,2)
      DIMENSION TOZ(NXM,NYM,NZM+1,2),TWZ(NXM,NYM,NZM+1,2),
*          TOZ1SA(NXM,NYM,NZM,2),TOZ1SI(NXM,NYM,NZM,2),
*          TWZ1SA(NXM,NYM,NZM,2),TWZ1SI(NXM,NYM,NZM,2),
*          TOZ2SI(NXM,NYM,NZM,2),TOZ2SP(NXM,NYM,NZM,2),
*          TWZ2SI(NXM,NYM,NZM,2),TWZ2SP(NXM,NYM,NZM,2)
      DIMENSION QO(NYM,NZM),QG(NYM,NZM),QW(NYM,NZM),DQOP(NYM,NZM),
*          DQOS(NYM,NZM),DQWP(NYM,NZM),DQWS(NYM,NZM)
      DIMENSION A(NXM*NYM*NZM*4,NXM*NYM*NZM*4),D(NXM*NYM*NZM*4)
      DIMENSION STPO(0:6,NZM),STPW(0:6,NZM)
C      STPO --> Somatorio de Trans. X Pot. do Oleo
C      DIMENSION SDEOPI(0:6,NZM),SDEWPI(0:6,NZM),SDEOSI(0:6,NZM),
*          SDEWSI(0:6,NZM)
C      SDEOP--> Som. da Derivada da Eq. do Oleo em rel. a Pres.
C-----

```

```

C  INTERFACES I=NXR
C-----
C      IF(IND.EQ.1) THEN
C-----
C  bloco abaixo do poco
C  cartesiano: i=ip j=jp-1 k=kc , radial: i j k
C-----
C      IF(J.GT.NQS+2*NQL.AND.JP.NE.1) THEN
C          NCC=(IP-1)*NYC*NZC+(JP-2)*NZC+(KP+K-1)
C          NLC=2*NCC-1
C          NCR=NBC+(I-1)*NYR*NZR+(J-1)*NZR+K
C          NLR=2*NCR-1
C  cartesiano com radial
C  dFw/dPoi,j+1,k
C          A(NLC,NLR)=DFWPX
C  dFw/dSwi,j+1,k
C          A(NLC,NLR+1)=DFWSX
C  dFo/dPoi,j+1,k
C          A(NLC+1,NLR)=DFOPX
C  dFo/dSwi,j+1,k
C          A(NLC+1,NLR+1)=DFOSX
C      ENDIF
C-----
C  bloco a esquerda do poco
C  cartesiano: i=ip-1 j=jp k=kc , radial: i j k
C-----
C      IF(J.GT.NQL+NQS.AND.J.LE.NQL+NQS.AND.IP.NE.1) THEN
C          NCC=(IP-2)*NYC*NZC+(JP-1)*NZC+(KP+K-1)
C          NLC=2*NCC-1
C          NCR=NBC+(I-1)*NYR*NZR+(J-1)*NZR+K
C          NLR=2*NCR-1
C  cartesiano com radial
C  dFw/dPoi+1,j,k
C          A(NLC,NLR)=DFWPX
C  dFw/dSwi+1,j,k
C          A(NLC,NLR+1)=DFWSX
C  dFo/dPoi+1,j,k
C          A(NLC+1,NLR)=DFOPX
C  dFo/dSwi+1,j,k
C          A(NLC+1,NLR+1)=DFOSX
C      ENDIF
C-----
C  bloco acima do poco
C  cartesiano: i=ip j=jp+1 k=kc , radial: i j k
C-----
C      IF(J.GT.NQL.AND.J.LE.NQL+NQS.AND.JP.NE.NYC) THEN
C          NCC=(IP-1)*NYC*NZC+JP*NZC+(KP+K-1)-NBP
C          NLC=2*NCC-1
C          NCR=NBC+(I-1)*NYR*NZR+(J-1)*NZR+K
C          NLR=2*NCR-1
C  cartesiano com radial
C  dFw/dPoi,j-1,k
C          A(NLC,NLR)=DFWPX
C  dFw/dSwi,j-1,k
C          A(NLC,NLR+1)=DFWSX
C  dFo/dPoi,j-1,k
C          A(NLC+1,NLR)=DFOPX
C  dFo/dSwi,j-1,k
C          A(NLC+1,NLR+1)=DFOSX
C      ENDIF
C-----

```

```

C  bloco a direita do poco
C  cartesiano: i=ip+1 j=jp k=kc , radial: i j k
C-----
      IF(J.LE.NQL.AND.IP.NE.NXC) THEN
          NCC=IP*NYC*NZC+(JP-1)*NZC+(KP+K-1)-NBP
          NLC=2*NCC-1
          NCR=NBC+(I-1)*NYR*NZR+(J-1)*NZR+K
          NLR=2*NCR-1
C  cartesiano com radial
C  dFw/dPoi-1,j,k
          A(NLC,NLR)=DFWPX
C  dFw/dSwi-1,j,k
          A(NLC,NLR+1)=DFWSX
C  dFo/dPoi-1,j,k
          A(NLC+1,NLR)=DFOPX
C  dFo/dSwi-1,j,k
          A(NLC+1,NLR+1)=DFOSX
      ENDIF
C-----
C  radial com cartesiano para qualquer posicao do cartesiano vizinho
C-----
C
C  dFw/dPoi+1,j,k
          A(NLR,NLC)=POTWX2*TWX2PP+TWX(I+1,J,K,L)*PWX2PP
C  dFw/dSwi+1,j,k
          A(NLR,NLC+1)=POTWX2*TWX2SP(I,J,K,L)-
          * TWX(I+1,J,K,L)*DPC(IX,JX,KP,2)
C  dFo/dPoi+1,j,k
          A(NLR+1,NLC)=POTOX2*TOX2PP+TOX(I+1,J,K,L)*POX2PP
C  dFo/dSwi+1,j,k
          A(NLR+1,NLC+1)=POTOX2*TOX2SP(I,J,K,L)
      ENDIF
C-----
C  INTERFACES K=1 E NXR
C-----
C-----
C  bloco anterior ao poco
C  cartesiano: i=ip j=jp k=kp-1 , radial: i j 1
C-----
      IF(IND.EQ.2) THEN
          NCC=(IP-1)*NYC*NZC+(JP-1)*NZC+KP-1
          NLC=2*NCC-1
          DO 10 M=1,NXR
          DO 10 N=1,NYR
              NCR=NBC+(M-1)*NYR*NZR+(N-1)*NZR+K
              NLR=2*NCR-1
C  cartesiano com radial
C  dFw/dPoi,j,k+1
              A(NLC,NLR)=DFWPZ
C  dFw/dSwi,j,k+1
              A(NLC,NLR+1)=DFWSZ
C  dFo/dPoi,j,k+1
              A(NLC+1,NLR)=DFOPZ
C  dFo/dSwi,j,k+1
              A(NLC+1,NLR+1)=DFOSZ
C  radial com cartesiano
C  dFw/dPoi,j,k-1
              A(NLR,NLC)=-POTWZ1*TWZ1PA-TWZ(I,J,K,L)*PWZ1PA
C  dFw/dSwi,j,k-1

```

```

      A(NLR,NLC+1)=-POTWZ1*TWZ1SA(I,J,K,L)-TWZ(I,J,K,L)*
      *      DPC(IP,JP,KP-1,2)
C dFo/dPoi,j,k-1
      A(NLR+1,NLC)=-POTOZ1*TOZ1PA-TOZ(I,J,K,L)*POZ1PA
C dFo/dSwi,j,k-1
      A(NLR+1,NLC+1)=-POTOZ1*TOZ1SA(I,J,K,L)
10  CONTINUE
    ENDIF
C-----
C  bloco posterior ao poco
C  cartesiano: i=ip j=jp k=kp+nbp , radial: i j nzs
C-----
      IF(IND.EQ.3) THEN
        NCC=(IP-1)*NYC*NZC+(JP-1)*NZC+KP
        NLC=2*NCC-1
        DO 11 M=1,NXR
          DO 11 N=1,NYR
            NCR=NBC+(M-1)*NYR*NZR+(N-1)*NZR+K
            NLR=2*NCR-1
C cartesiano com radial
C dFw/dPoi,j,k-1
            A(NLC,NLR)=DFWPZ
C dFw/dSwi,j,k-1
            A(NLC,NLR+1)=DFWSZ
C dFo/dPoi,j,k-1
            A(NLC+1,NLR)=DFOPZ
C dFo/dSwi,j,k-1
            A(NLC+1,NLR+1)=DFOSZ
C radial com cartesiano
C dFw/dPoi,j,k+1
            A(NLR,NLC)=POTWZ2*TWZ2PP+TWZ(I,J,NZR+1,L)*PWZ2PP
C dFw/dSwi,j,k+1
            A(NLR,NLC+1)=POTWZ2*TWZ2SP(I,J,NZR,L)-
      *      TWZ(I,J,NZR+1,L)*DPC(IP,JP,KP+NBp,2)
C dFo/dPoi,j,k+1
            A(NLR+1,NLC)=POTOZ2*TOZ2PP+TOZ(I,J,NZR+1,L)*POZ2PP
C dFo/dSwi,j,k+1
            A(NLR+1,NLC+1)=POTOZ2*TOZ2SP(I,J,NZR,L)
11  CONTINUE
    ENDIF
    RETURN
  END
C-----
C
C
C  SOLUCAO DO SISTEMA USANDO METODO DIRETO COM GAUSS
C-----
C  SUBROUTINE SOLVE1
C
  IMPLICIT DOUBLE PRECISION(A-H,O-Z)
  PARAMETER(NXM=08,NYM=05,NZM=04)
  COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
  *      NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
  COMMON/SEQS/XR,XC
  COMMON/NBLC/A,D,NBC,NBR,NBT
  DIMENSION XR(NXM*NYM*NZM*2),XC(NXM*NYM*NZM*2)
  DIMENSION A(NXM*NYM*NZM*4,NXM*NYM*NZM*4),D(NXM*NYM*NZM*4)
C-----
C  calculo do n. de incognitas (N)
C-----
  NR=NXR*NYR*NZR*2

```

```

NC=NXC*NYC*NZC*2-NBP*2
C-----
C  entra com A,D,NR e NC e sai com XR e XC
C-----
CALL GAUSS1(NR,NC)
RETURN
END
C-----
C SOLUCAO DA MATRIZ USANDO O METODO DE ELIMINACAO DE GAUSS
C-----
SUBROUTINE GAUSS1 (NR,NC)
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
PARAMETER(NXM=08,NYM=05,NZM=04)
COMMON/SEQS/XR,XC
COMMON/NBLC/A,D,NBC,NBR,NBT
DIMENSION XR(NXM*NYM*NZM*2),XC(NXM*NYM*NZM*2),X(NXM*NYM*NZM*4)
DIMENSION A(NXM*NYM*NZM*4,NXM*NYM*NZM*4),D(NXM*NYM*NZM*4)
C-----
N=NC+NR
C-----
C TRIANGULARIZACAO
C-----
DO 650 K=1,N-1
DO 640 I=K+1,N
FATOR=A(I,K)/A(K,K)
D(I)=D(I)-FATOR*D(K)
A(I,K)=0.0D0
DO 630 J=K+1,N
A(I,J)=A(I,J)-FATOR*A(K,J)
630 CONTINUE
640 CONTINUE
650 CONTINUE
C-----
C SUBSTITUICAO REGRESSIVA
C-----
X(N)=D(N)/A(N,N)
DO 670 I=N-1,1,-1
SOM=0.0D0
DO 660 J=N,I+1,-1
SOM=SOM+A(I,J)*X(J)
660 CONTINUE
X(I)=(D(I)-SOM)/A(I,I)
670 CONTINUE
C-----
C SEPARACAO DO VETOR SOLUCAO EM DOIS VETORES
C-----
DO 10 I=1,NC
XC(I)=X(I)
10 CONTINUE
IC=0
DO 20 I=NC+1,NR+NC
IC=IC+1
XR(IC)=X(I)
20 CONTINUE
C-----
RETURN
END
C-----
C SOLUCAO DO SISTEMA USANDO METODO DIRETO COM ALGORITMO OTIMIZADO
C-----
SUBROUTINE SOLVE2
C-----
IMPLICIT DOUBLE PRECISION(A-H,O-Z)

```

```

PARAMETER(NXM=08,NYM=05,NZM=04,MAX=1500)
COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*      NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
COMMON/SEQS/XR,XC
COMMON/NBLC/AA,B,NBC,NBR,NBT
DIMENSION XR(NXM*NYM*NZM*2),XC(NXM*NYM*NZM*2)
DIMENSION AA(NXM*NYM*NZM*4,NXM*NYM*NZM*4),B(NXM*NYM*NZM*4)
C
INTEGER R(NXM*NYM*NZM*4),C(NXM*NYM*NZM*4)
INTEGER ITEMP(8*NXM*NYM*NZM+MAX+2),RTEMP(4*NXM*NYM*NZM+MAX)
DIMENSION A(NXM*NXM*NYM*NYM*NZM*NZM*2)
DIMENSION X(NXM*NYM*NZM*4)
DIMENSION IA(NXM*NYM*NZM*4+1),JA(NXM*NYM*NZM*4),IC(NXM*NYM*NZM*4)
C-----
C  calculo do n. de incognitas (N)
C-----
      NR=NXR*NYR*NZR*2
      NC=NXC*NYC*NZC*2-NBP*2
      N=NC+NR
C-----
C  MONTAGEM DOS VETORES PARA A NSPIV
C-----
      INI=0
      II=0
      IFIRST=0
      ILAST=0
      LOC=1
      DO 20 I=1,N
        DO 15 J=1,N
          IF(AA(I,J).NE.0.0D000) THEN
            II=II+1
            A(II)=AA(I,J)
            JA(II)=J
            IF(INI.EQ.0) THEN
              IA(1)=J
              INI=1
            ENDIF
          ENDIF
        CONTINUE
        ILAST=II
        LOC=LOC+1
        IA(LOC)=IA(LOC-1)+ILAST-IFIRST
        IFIRST=II
      CONTINUE
      DO 26 I=1,N
        R(I)=I
        C(I)=I
        IC(I)=I
      CONTINUE
      CALL NSPIV(N,IA,JA,A,B,MAX,R,C,IC,X,ITEMP,RTEMP,IERR)
C-----
C  SEPARACAO DO VETOR SOLUCAO EM DOIS VETORES
C-----
      DO 30 I=1,NC
        XC(I)=X(I)

      CONTINUE
      IK=0
      DO 40 I=NC+1,NR+NC
        IK=IK+1
        XR(IK)=X(I)
      CONTINUE
      RETURN
      END

```

Subrotina NSPIV

Usa a eliminacao Gaussiana dispersa com intercambio de coluna para resolver um sistema linear do tipo $(A * X = B)$. A fase eliminacao realiza operacoes nas linhas de 'A' e 'B' para obter uma matriz triangular superior 'U' e um vetor 'Y'. A fase solucao resolve $(U * X = Y)$

Argumentos de entrada :

- N - numero de equacoes e incognitas
- IA - vetor de $(N + 1)$ entradas contendo indicadores de linha para 'A' (veja descricao do armazenamento da matriz abaixo)
- JA - vetor com uma entrada por elemento nao-zero em 'A' contendo os numeros das colunas de elementos nao-zero de 'A'
- A - vetor real com uma entrada por elemento nao-zero em 'A' contendo os verdadeiros elementos nao-zero (veja descricao do armazenamento de matriz abaixo)
- B - vetor real com N entradas contendo os dados do lado direito
- MAX - numero inteiro especificando o numero maximo de entradas de "diagonais extras com elementos nao-zero" de 'U' que devem ser armazenadas
- R - vetor inteiro com N entradas especificando a ordem das linhas de 'A'
- C - vetor inteiro com N entradas especificando a ordem das colunas de 'A'. 'C' e tambem um argumento de saida
- IC - vetor inteiro de N entradas que e o inverso de 'C' (i.e., $IC(C(I)) = I$). IC e tambem um argumento de saida
- ITEMP - vetor inteiro de $(2 * N + MAX + 2)$ entradas para uso interno
- RTEMP - vetor inteiro de $(N + MAX)$ entradas para uso interno

Argumentos de saida :

- C - vetor inteiro de N entradas especificando a ordem das colunas de 'U'. 'C' e tambem um argumento de entrada
- IC - vetor inteiro de N entradas que e o inverso de 'C' (i.e., $IC(C(I)) = I$). IC e tambem um argumento de entrada
- X - vetor real de N entradas contendo o vetor solucao
- IERR - numero inteiro que indica a condicao de erro ou o numero de entradas de "diagonais extras" em 'U' (para finalizacao bem-sucedida)

Os valores de IERR sao:

0 .LT. IERR

Finalizacao bem-sucedida. 'U' tem IERR entradas de "diagonais extras" nao-zero.

IERR = 0

Erro. N = 0

-N.LE.IERR.LT.0

Erro. Linha numero IABS(IERR) de 'A' e nula.

-2*N.LE.IERR.LT.-N

Erro.Linha numero IABS(IERR + N) tem uma entrada duplicada.

-3*N.LE.IERR.LT.-2*N

Erro.Linha numero IABS(IERR + 2*N) tem um pivo zero.

-4*N.LE.IERR.LT.IERR.LT.3*N

Erro.Linha numero IABS(IERR + 3*N) excede a capacidade de armazenamento

Armazenamento da matriz esparsa :

A matriz esparsa 'A' e armazenada usando tres vetores 'IA', 'JA' e 'A'. O vetor 'A' contem os elementos nao-zeros da matriz linha por linha, nao necessariamente em ordem crescente do numero de colunas. O vetor 'JA' contem os numeros das colunas correspondentes aos elementos nao-zero armazenados no vetor 'A' (i.e., se o elemento nao-zero armazenado em 'A(K))' esta na coluna 'J' entao, 'JA(K) = J'). O vetor 'IA' contem indicadores para as linhas de indices de colunas nao-zero no vetor 'A/JA' (i.e., 'A(IA(I))/JA(IA(I))' e a primeira entrada para a linha 'I' no vetor 'A/JA'). 'IA(N + 1)' e considerado tal que 'IA(N+1)=IA(I)' = numero de nao-zeros em 'A'.

```
SUBROUTINE NSPIV(N,IA,JA,A,B,MAX,R,C,IC,X,ITEMP,RTEMP,IERR)
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
DIMENSION A(1), B(1), X(1), RTEMP(1)
INTEGER IA(1), JA(1), R(1), C(1), IC(1), ITEM(1)
INTEGER IU, JU, U, Y, P
```

Coloca indices para dividir o armazenamento temporario por NSPIVI

```
Y = 1
U = Y + N
P = 1
IU = P + N + 1
JU = IU + N + 1
```

Chama a subrotina NSPIVI para realizar os calculos

```
CALL NSPIVI(N,IA,JA,A,B,MAX,R,C,IC,X,RTEMP(Y),ITEMP(P),
+          ITEM(IU),ITEM(JU),RTEMP(U),IERR)
RETURN
END
```

Subrotina NSPIVI

NSPIVI usa a eliminacao Gaussiana com troca de colunas para resolver o sistema linear 'A* X= B'. A fase eliminacao realiza operacoes em 'A' e em 'B' para obter uma unidade de matriz triangular superior 'U' e um vetor 'Y'. A fase solucao resolve 'U * X = Y'.

Argumentos de entrada (usados apenas internamente):

C Y - Vetor real de 'N' entradas usada para computar os valo-
 C res atualizados do lado direito
 C P - Vetor inteiro de 'N+1' entradas usado por uma lista
 C de conexao. 'P(N+1)' e o primeiro da lista, e a entra-
 C da seguinte 'P(K)' e em 'P(P(K))'. Entao 'P(N+1)' e o
 C primeiro item dado, 'P(P(N+1))' e o segundo, etc. Um
 C indicador de 'N+1' marca o fim da lista
 C IU - vetor inteiro de 'N+1' entradas usado como indicador
 C de linhas para 'U'
 C JU - vetor inteiro de 'MAX' entradas usados para as colunas de
 C elementos nao-zero no triangulo estritamente superior de
 C 'U'
 C U - vetor real de 'MAX' entradas usados para os elementos nao-
 C zero no triangulo estritamente superior de 'U'

Armazenamento da matriz esparsa:

C A matriz esparsa 'A' e armazenada usando tres vetore 'IA', 'JA'
 C e 'A'. O vetor 'A' contem os elementos nao-zero da matriz linha por
 C linha, nao necessariamente em ordem crescente de numeros de colunas.
 C O vetor 'JA' contem os numeros das colunas correspondentes aos ele-
 C mentos nao-zero armazenados no vetor 'A' (i.e., se o elemento nao-
 C zero armazenado em 'A(K)' esta na coluna 'J', entao 'JA(K) = J'. O
 C vetor 'IA' contem indicadores para as linhas de indices de coluna
 C 'nao-zero' no vetor 'A/JA' (i.e., 'A(IA(I))/JA(IA(I)) e a primeira
 C para a linha 'I' no vetor 'A/JA'). 'IA(N+1)=IA(I)' = numero de
 C elementos nao-zero em 'A'. 'IU', 'JU' e 'U' sao usado de maneira si-
 C milar para armazenar o triangulo estritamente superior de 'U', exce-
 C to que 'J(U)' contem na verdade 'C(J)' ao inves de 'J'.

```

SUBROUTINE NSPIVI(N,IA,JA,A,B,MAX,R,C,IC,X,Y,P,IU,JU,U,IERR)
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
DIMENSION A(1), B(1), U(1), X(1), Y(1)
REAL*8 DK, LKI, ONE, XPV, XPVMAX, YK, ZERO
INTEGER C(1), IA(1), IC(1), IU(1), JA(1), JU(1), P(1), R(1)
INTEGER CK, PK, PPK, PV, V, VI, VJ, VK
  
```

```

IF(N .EQ. 0) GO TO 1001
  
```

```

ONE = 1.
ZERO = 0.0D0
  
```

C Inicializa o trabalho de armazenamento e indicadores para 'JU'

```

DO 10 J=1,N
  X(J) = 0.0D0
10 CONTINUE
IU(1) = 1
JUPT = 0
  
```

C Realiza fatorizacao numerica e simbolica linha por linha
 C 'VK(VI,VJ)' e o vertice do grafico para a linha 'K-th' de 'U'

C Inicia lista de conexao e armazenamento livre para esta linha
 C a linha 'R(K)-th' de 'A' torna-se a linha 'K-th' de 'U'

```

DO 170 K = 1,N
  
```

```

      P(N+1) = N + 1
      VK = R(K)
C
C Estabelece a linha adjacente para 'VK', ordenado na ordem corren-
C te das colunas de 'U'. O indice do laço diminui para explorar qual-
C quer coluna de 'A' numa ordem relativa correta
C
      JMIN = IA(VK)
      JMAX = IA(VK+1) - 1
      IF(JMIN .GT. JMAX) GO TO 1002
      J = JMAX
20    JAJ = JA(J)
      VJ = IC(JAJ)
C
C Armazena 'A(K,J)' no vetor de trabalho
C
      X(VJ) = A(J)
C Este código insere 'VJ' na lista adjacente de 'VK'
      PPK = N + 1
30    PK = PPK
      PPK = P(PK)
      IF(PPK - VJ) 30,1003,40
40    P(VJ) = PPK
      P(PK) = VJ
      J = J - 1
      IF(J .GE. JMIN) GO TO 20
C
C O código seguinte calcula a 'K-th' linha de 'U'
C
      VI = N + 1
      YK = B(VK)
50    VI = P(VI)
      IF(VI .GE. K) GO TO 110
C
C 'VI .LT. VK' -- processa o elemento L(K,I) e margeia a adjacência
C de 'VI' com a adjacência ordenada de 'VK'
C
      LKI = - X(VI)
      X(VI) = ZERO
C
C Ajusta o lado direito para refletir a eliminação
C
      YK = YK + LKI * Y(VI)
      PPK = VI
      JMIN = IU(VI)
      JMAX = IU(VI+1) - 1
      IF(JMIN .GT. JMAX) GO TO 50
      DO 100 J=JMIN,JMAX
        JUJ = JU(J)
        VJ = IC(JUJ)
C
C Se 'VJ' já está na adjacência de 'VK', pule a inserção
C
      IF(X(VJ) .NE. ZERO) GO TO 90
C
C Insere 'VJ' na lista adjacente de 'VK'
C "reset" 'PPK' to 'VI' se nos tivermos passado o ponto de inserção
C correto. (Isto acontece quando a adjacência de 'VI' não está na or-
C dem corrente da coluna devido ao pivoteamento)
C
      IF(VJ - PPK) 60,90,70
60    PPK = VI
70    PK = PPK

```

```

      PPK = P(PK)
      IF(PPK - VJ) 70,90,80
80    P(VJ) = PPK
      P(PK) = VJ
      PPK = VJ
C
C  Calcula 'L(K,J) = L(K,I)*U(I,J)' para 'L(K,I)' nao-zero, compute
C  'U*(K,J) = U*(K,J) - L(K,I) * U(I,J)' para 'U(K,J)' nao - zero
C  '(U*(K,J) = U(K,J) * D(K,K))'
C
90    X(VJ) = X(VJ) + LKI * U(J)
100   CONTINUE
      GO TO 50
C
C  Pivo - troca a maior entrada da 'K-th' linha de 'U' com a entrada
C  da diagonal
C
C  Encontra a maior entrada, contando as diagonais extras nao-zero
C
110   IF(VI .GT. N) GO TO 1004
      XPVMAX = DABS(X(VI))
      MAXC = VI
      NZCNT = 0
      PV = VI
120   V = PV
      PV = P(PV)
      IF(PV .GT. N) GO TO 130
      NZCNT = NZCNT + 1
      XPV = DABS(X(PV))
      IF(XPV .LE. XPVMAX) GO TO 120
      XPVMAX = XPV
      MAXC = PV
      MAXCL = V
      GO TO 120
130   IF(XPVMAX .EQ. ZERO) GO TO 1004
C
C  Se 'VI=K', entao existe uma entrada para diagonal que deve ser dele-
C  tada. Senao delete a entrada que ira ser a entrada da diagonal.
C
      IF(VI .EQ. K) GO TO 140
      IF(VI .EQ. MAXC) GO TO 140
      P(MAXCL) = P(MAXC)
      GO TO 150
140   VI = P(VI)
C
C  Calcula 'D(K) = 1/L(K,K)' e faz a troca
C
150   DK = ONE / X(MAXC)
      X(MAXC) = X(K)
      I = C(K)
      C(K) = C(MAXC)
      C(MAXC) = I
      CK = C(K)
      IC(CK) = K
      IC(I) = MAXC
      X(K) = ZERO
C
C  Atualiza lado direito
C
      Y(K) = YK * DK
C
C  Calcula valor para 'IU(K+1)' e compare para armazenamento "overflow"
C
      IU(K+1) = IU(K) + NZCNT

```

```

      IF(IU(K+1) .GT. MAX+1) GO TO 1005
C
C Move indices das colunas da lista de conexao para 'JU'.
C As colunas sao armazenadas em ordem corrente com a numeracao origi -
C nal das colunas ('C(J)') e armazenado na coluna corrente 'J'
C
      IF(VI .GT. N) GO TO 170
      J = VI
160    JUPTR = JUPTR + 1
      JU(JUPTR) = C(J)
      U(JUPTR) = X(J) * DK
      X(J) = ZERO
      J = P(J)
      IF(J .LE. N) GO TO 160
170  CONTINUE
C Volta e resolve 'U*X= Y' e reordena 'X' ao correspondente com 'A'
C
      K = N
      DO 200 I = 1, N
        YK = Y(K)
        JMIN = IU(K)
        JMAX = IU(K+1) - 1
        IF(JMIN .GT. JMAX) GO TO 190
        DO 180 J = JMIN, JMAX
          JUJ = JU(J)
          JUJ = IC(JUJ)
          YK = YK - U(J) * Y(JUJ)
180      CONTINUE
190      Y(K) = YK
          CK = C(K)
          X(CK) = YK
          K = K - 1
200  CONTINUE
C
C Retorne com 'IERR' = numero de diagonais extras de nao-zeros em 'U'
C
      IERR = IU(N+1) - IU(1)
      RETURN
C
C Retorno dos erros
C
      N = 0
1001  IERR = 0
      RETURN
C
C Linha 'K' de 'A' e nula
1002  IERR = -K
      RETURN
C
C Linha 'K' de 'A' tem entrada duplicada
1003  IERR = -(N+K)
      RETURN
C
C Pivo zero na linha 'K'
1004  IERR = -(2 * N + K)
      RETURN
C
C Armazenamento de 'U' excedido na linha 'K'
1005  IERR = -(3 * N + K)
      RETURN
      END

```

```

C-----
C
C      SOLUCAO DO SISTEMA USANDO METODO ITERATIVO BSOR COM GAUSS
C-----
C
C      SUBROUTINE SOLVE3
C      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
C      PARAMETER(NXM=08,NYM=05,NZM=04)
C      COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
C      *      NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
C      COMMON/SIM/DT,DTMAX,DTMIN,TOLP,TOLS,DSLIM,DPLIM,TSIM,
C      *      QOMAX,QOMIN,PWMIN,BSOR,JPROD,MSOLU,ITMAX,ITMIN
C      COMMON/SEQS/XR,XC
C      COMMON/NBLC/A,D,NBC,NBR,NBT
C
C      DIMENSION XR(NXM*NYM*NZM*2),XC(NXM*NYM*NZM*2)
C      DIMENSION A(NXM*NYM*NZM*4,NXM*NYM*NZM*4),D(NXM*NYM*NZM*4)
C      DIMENSION ACG(NXM*NYM*NZM*2,NXM*NYM*NZM*2)
C      DIMENSION ARG(NXM*NYM*NZM*2,NXM*NYM*NZM*2)
C      DIMENSION DRA(NXM*NYM*NZM*2),DCA(NXM*NYM*NZM*2)
C      DIMENSION DR(NXM*NYM*NZM*2),DC(NXM*NYM*NZM*2)
C      DIMENSION XRA(NXM*NYM*NZM*2),XCA(NXM*NYM*NZM*2)
C      LOGICAL CONV
C-----
C      entra com a matriz jacobiana total (A) e o vetor (D) termo indep.
C      para o GAUSS, faz:
C      transforma a parte da matriz A ref. ao radial, na matriz ARG
C      transforma a parte da matriz A ref. ao cartesiano, na matriz ACG
C      transforma a parte do vetor D ref. ao radial, no vetor DRA
C      transforma a parte do vetor D ref. ao cartesiano, no vetor DCA
C      transforma a parte do vetor X ref. ao radial, no vetor XRA
C      transforma a parte do vetor X ref. ao cartesiano, no vetor XCA
C      a solucao e representada pelos vetores XR e XC
C-----
C      calculo do n. de incognitas (N) e largura da banda (M)
C-----
C      NR=NXR*NYR*NZR*2
C      NC=NXC*NYC*NZC*2-NBP*2
C-----
C      transformacao do vetor X em dois vetores
C-----
C      DO 1 I=1,NR
C          XRA(I)=0.0D0
C      1  CONTINUE
C      DO 2 I=1,NC
C          XCA(I)=0.0D0
C      2  CONTINUE
C-----
C      transformacao do vetor D em dois vetores
C-----
C      DO 5 I=1,NBC*2
C          DCA(I)=D(I)
C      5  CONTINUE
C      IC=0
C      DO 6 I=(NBC*2+1),NBT*2
C          IC=IC+1
C          DRA(IC)=D(I)
C      6  CONTINUE
C-----
C      transformacao da matriz A nas matrizes ARG e ACG
C-----
C      DO 10 I=1,NC
C          DO 10 J=1,NC

```

```

      ACG(I,J)=A(I,J)
10  CONTINUE
C
      LX=0
      DO 20 I=NC+1,NC+NR
          LX=LX+1
          LY=0
          DO 20 J=NC+1,NC+NR
              LY=LY+1
              ARG(LX,LY)=A(I,J)
20  CONTINUE
C-----
C  PONTO DE RETORNO DO PROCESSO ITERATIVO
C-----
C
      iter=0
1000 CONTINUE
      iter=iter+1
C-----
C  montagem do termo independente
C-----
      DO 60 L=1,2
          IF(L.EQ.1) THEN
              LIX=NC+1
              LSX=NC+NR
              LIY=1
              LSY=NC
          ELSE
              LIX=1
              LSX=NC
              LIY=NC+1
              LSY=NC+NR
          ENDIF
          IC=0
          DO 40 I=LIX,LSX
              AUX=0.000
              ID=0
              DO 50 J=LIY,LSY
                  ID=ID+1
                  IF(L.EQ.1) AUX=AUX+A(I,J)*XCA(ID)
                  IF(L.EQ.2) AUX=AUX+A(I,J)*XR(ID)
50          CONTINUE
              IC=IC+1
              IF(L.EQ.1) DR(IC)=DRA(IC)-AUX
              IF(L.EQ.2) DC(IC)=DCA(IC)-AUX
40          CONTINUE
              IF(L.EQ.1) THEN
                  CALL GAUSS2 (NR,ARG,DR,XR)
                  DO 70 K=1,NR
                      XR(K)=BSOR*XR(K)+(1.000-BSOR)*XRA(K)
C              write(*,*)'i=',k,' xr=',xr(k)
C              write(*,*)'i=',k,' xra-xr=',xra(k)-xr(k)
70          CONTINUE
                  GO TO 60
              ELSE
                  CALL GAUSS2 (NC,ACG,DC,XC)
                  DO 80 K=1,NC
                      XC(K)=BSOR*XC(K)+(1.000-BSOR)*XCA(K)
C              write(*,*)'i=',k,' xc=',xc(k)
C              write(*,*)'i=',k,' xca-xc=',xca(k)-xc(k)
80          CONTINUE
              ENDIF
60          CONTINUE
C-----
C  teste do convergencia
C-----
      CONV=.TRUE.

```

```

DO 110 I=1,NR
  IF(DABS(XR(I)-XRA(I)).GT.1.0D-9) CONV=.FALSE.
110 CONTINUE
DO 120 I=1,NC
  IF(DABS(XC(I)-XCA(I)).GT.1.0D-9) CONV=.FALSE.
120 CONTINUE
-----
C atualizacao dos resultados
C-----
DO 90 I=1,NR
  XRA(I)=XR(I)
90 CONTINUE
DO 100 I=1,NC
  XCA(I)=XC(I)
100 CONTINUE
-----
C IF(.NOT.CONV) GO TO 1000
C write(*,*)'bsor=',bsor,' convergiu iter=',iter
C RETURN
C END
C-----
C SOLUCAO DA MATRIZ USANDO O METODO DE ELIMINACAO DE GAUSS
C-----
SUBROUTINE GAUSS2 (N,A,D,X)
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
PARAMETER(NXM=08,NYM=05,NZM=04)
C
  DIMENSION A(NXM*NYM*NZM*2,NXM*NYM*NZM*2),D(NXM*NYM*NZM*2),
  *          X(NXM*NYM*NZM*2)
C-----
C TRIANGULARIZACAO
C-----
DO 650 K=1,N-1
  DO 640 I=K+1,N
    FATOR=A(I,K)/A(K,K)
    D(I)=D(I)-FATOR*D(K)
    A(I,K)=0.0D0
    DO 630 J=K+1,N
      A(I,J)=A(I,J)-FATOR*A(K,J)
630 CONTINUE
640 CONTINUE
650 CONTINUE
C-----
C SUBSTITUICAO REGRESSIVA
C-----
X(N)=D(N)/A(N,N)
DO 670 I=N-1,1,-1
  SOM=0.0D0
  DO 660 J=N,I+1,-1
    SOM=SOM+A(I,J)*X(J)
660 CONTINUE
  X(I)=(D(I)-SOM)/A(I,I)
670 CONTINUE
C
  RETURN
  END
C-----
C SUBROT. DE CALCULO DE PRODUCOES DE OLEO E GAS POR BALANCO DE MATERIAIS
C-----
SUBROUTINE BALM
C
  IMPLICIT DOUBLE PRECISION(A-H,O-Z)
  PARAMETER(NXM=08,NYM=05,NZM=04)
  COMMON/ACUM/QOAC,QGAC,QWAC,VOORR,VWORR,VOORC,VWORC,BNPR,BWPR,

```

```

*          BNPC,BWPC,PWFA
COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
*          NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP

COMMON/MALHA/ZB,RB,VPOR,DRO,WX,WY,WZ,FFX,FFY,FFZ,DTETA
COMMON/PVT/BO,DBO,VO,DVO,DO,DDO,RSO,DRSO,BW,DBW,VW,DVW,DW,DDW,CW
COMMON/TS/PO,SO,SG,SW
COMMON/TF/TFLUXO,TFLUXW,SUMTO,SUMTW,AFXO,AFXW
DIMENSION ZB(0:NXM+1,NYM,2),RB(NXM+1),VPOR(NXM,NYM,NZM,2),
*          FFX(NXM+1,NYM,NZM,2),FFY(NXM+1,NYM,NZM,2),
*          FFZ(NXM+1,NYM,NZM+1,2),DTETA(NYM+1)
DIMENSION BO(NXM,NYM,NZM,2),DBO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          VO(NXM,NYM,NZM,2),DVO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          DO(NXM,NYM,NZM,2),DDO(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          BW(NXM,NYM,NZM,2),DBW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          VW(NXM,NYM,NZM,2),DVW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          DW(NXM,NYM,NZM,2),DDW(0:NXM+1,0:NYM+1,0:NZM+1,2),
*          CW(NXM,NYM,NZM,2)
DIMENSION PO(0:NXM+1,0:NYM+1,0:NZM+1,2),SO(NXM,NYM,NZM,2),
*          SG(NXM,NYM,NZM,2),SW(NXM,NYM,NZM,2)

C      DO 10 L=1,2
        IF(L.EQ.1) THEN
            BNPR=0.0D0
            BWPR=0.0D0
            DO 4 J=1,NYR
                DO 4 I=1,NXR
                    DO 4 K=1,NZR
                        BNPR=BNPR+VPOR(I,J,K,L)*SO(I,J,K,L)*BO(I,J,K,L)
                        BWPR=BWPR+VPOR(I,J,K,L)*SW(I,J,K,L)*BW(I,J,K,L)
4          CONTINUE
        ELSE
            BNPC=0.0D0
            BWPC=0.0D0
            DO 5 J=1,NYC
                DO 5 I=1,NXC
                    DO 5 K=1,NZC
                        IF(I.EQ.IP.AND.J.EQ.JP.AND.K.GE.KP.AND.K.LT.KP+NBP) GO TO 5
                        BNPC=BNPC+VPOR(I,J,K,L)*SO(I,J,K,L)*BO(I,J,K,L)
                        BWPC=BWPC+VPOR(I,J,K,L)*SW(I,J,K,L)*BW(I,J,K,L)
5          CONTINUE
        ENDIF
10     CONTINUE
C      RETURN
      END

```

```

C-----
C      SUBROTINA DE IMPRESSAO DE RESULTADOS PARA RELATORIO
C

```

```

C      SUBROUTINE IMPRER(DT,NTS)

```

```

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)

```

```

      PARAMETER(NXM=08,NYM=05,NZM=04)

```

```

      COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,

```

```

*          NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP

```

```

      COMMON/IMP/TIMPR,TIMPG,IRIMP

```

```

      COMMON/ACUM/QOAC,QGAC,QWAC,VOORR,VWORR,VOORC,VWORC,BNPR,BWPR,

```

```

*          BNPC,BWPC,PWFA

```

```

      COMMON/MALHA/ZB,RB,VPOR,DRO,WX,WY,WZ,FFX,FFY,FFZ,DTETA

```

```

      COMMON/TS/PO,SO,SG,SW

```

```

      COMMON/VAZAO/QO,QOT,QG,QGT,QW,QWT,DQOP,DQOS,

```

```

*          DQWP,DQWS,RWO,RGO,PWF,TEMPO
DIMENSION PO(0:NXM+1,0:NYM+1,0:NZM+1,2),SO(NXM,NYM,NZM,2),
*          SG(NXM,NYM,NZM,2),SW(NXM,NYM,NZM,2)
DIMENSION ZB(0:NXM+1,NYM,2),RB(NXM+1),VPOR(NXM,NYM,NZM,2),
*          FFX(NXM+1,NYM,NZM,2),FFY(NXM+1,NYM,NZM,2),
*          FFZ(NXM+1,NYM,NZM+1,2),DTETA(NYM+1)
DIMENSION QO(NYM,NZM),QG(NYM,NZM),QW(NYM,NZM),DQOP(NYM,NZM),
*          DQOS(NYM,NZM),DQWP(NYM,NZM),DQWS(NYM,NZM)
CHARACTER*8 TIT

C
C1=86400.0D0
C2=1.0D+6
C3=C2/1000.0D0
REC=QOAC/(VOORR+VOORC)*100.0
PID=QOT*C1/((PO(NXC,JP,KP,2)-PWF)/98067.0D0)

C
WRITE(4,110)NTS
110  FORMAT(' ',T9,43('-')),/,T20,'RESUMO DO TIME STEP',I3,/,T9,43('-'))
C
IF(QOAC.NE.0.0D0) BALO=(QOAC+(BNPR+BNPC))/(VOORR+VOORC)
IF(QWAC.NE.0.0D0) BALW=(QWAC+(BWPR+BWPC))/(VWORR+VWORC)
BALG=1.0D0
WRITE(4,120)TEMPO/C1,DT/C1,C1*QOT,
*C1*QGT,C1*QWT,RGO,RWO,PWFA/C2,
*QOAC,QGAC,QWAC,BALO,BALG,BALW,REC,PID
120  FORMAT(T10,'Tempo (dia) ',T31,'!',F10.5,'!',/,
*T10,'Dt (dia) ',T31,'!',F10.5,'!',/,
*T10,'Qo (m3/dia) ',T31,'!',D10.4,'!',/,
*T10,'Qg (m3/dia) ',T31,'!',D10.4,'!',/,
*T10,'Qw (m3/dia) ',T31,'!',D10.4,'!',/,
*T10,'RGO (m3/m3) ',T31,'!',F10.4,'!',/,
*T10,'RWO (m3/m3) ',T31,'!',F10.4,'!',/,
*T10,'Pwf (MPa) ',T31,'!',F10.4,'!',/,
*T10,'Np (m3) ',T31,'!',D10.4,'!',/,
*T10,'Gp (m3) ',T31,'!',D10.4,'!',/,
*T10,'Wp (m3) ',T31,'!',D10.4,'!',/,
*T9,43('-')),/,T9,'Balanco de Materais',/,T9,43('-')),/,
*T10,'Dleo (%) ',T31,'!',D10.4,'!',/,
*T10,'Gas (%) ',T31,'!',D10.4,'!',/,
*T10,'Agua (%) ',T31,'!',D10.4,'!',/,
*T9,43('-')),/,
*T10,'Recup. (%) ',T31,'!',F10.4,/,
*T10,'IP (m3/d/kg/cm2) ',T31,'!',F10.4,/,
*T9,43('-'))

C-----
C Mapa de pressao e saturacao do grid cartesiano
C-----
WRITE(4,400)
400  FORMAT(T22,'Mapa de Pressao e Saturacao (Presao, So e Sw)')
WRITE(4,100)
100  FORMAT(/,5X,'GRID CARTESIANO')
DO 410 K=1,NZC
WRITE(4,420)K
420  FORMAT(1X,'Camada ',I2,/,70('-'))
WRITE(4,450)
450  FORMAT(T4,'!',01,'!',02,'!',03,'!',04,'!',
*' 05,'!',06,'!',/,70('-'))
DO 410 J=NYC,1,-1

```

```

        WRITE(4,460)(PO(I,J,K,2)/C2,I=1,NXC)
460  FORMAT(T4,'! ',6(F8.4,'! '))
        WRITE(4,470)J,(SO(I,J,K,2),I=1,NXC)
470  FORMAT(I2,'! ',6(F8.5,'! '))
        WRITE(4,480)(SW(I,J,K,2),I=1,NXC)
480  FORMAT(T4,'! ',6(F8.5,'! '))
        WRITE(4,490)
490  FORMAT(70('-',))
410  CONTINUE
C-----
C Mapa de pressao e saturacao do grid radial
C-----
        LIX=1
        IF(NXR.LE.6) THEN
            LIM=1
            LSX=NXR
        ENDIF
        IF(NXR.GT.6.AND.NXR.LE.12) THEN
            LIM=2
            LSX=6
        ENDIF
        WRITE(4,500)
500  FORMAT(/,5X,'GRID RADIAL')
        DO 421 K=1,NZR
            WRITE(4,420)K
            DO 510 M=1,LIM
                IF(M.EQ.1)WRITE(4,450)
                IF(M.EQ.2)WRITE(4,451)
451  FORMAT(T4,'! 07 ! 08 ! 09 ! 10 !',
* ' 11 ! 12 !',/,70('-',))
                DO 610 J=1,NYR
                    WRITE(4,460)(PO(I,J,K,1)/C2,I=LIX,LSX)
                    WRITE(4,470)J,(SO(I,J,K,1),I=LIX,LSX)
                    WRITE(4,480)(SW(I,J,K,1),I=LIX,LSX)
                    WRITE(4,490)
610  CONTINUE
                    IF(LIM.EQ.2) THEN
                        LIX=7
                        LSX=NXR
                    ENDIF
510  CONTINUE
421  CONTINUE
C-----
C DISTRIBUICAO DAS VAZDES
C-----
        DO 61 K=1,NZR
            WRITE(4,420)K
            WRITE(4,45)
45  FORMAT(/,6X,'BLOCO ',2X,'PRESSAO ',3X,' Qoj,k',3X,
* ' Qwj,k',10X,'RWD',
* /,3X,'I=1 J (MPa) (m3/D) (m3/D)',8X,'(m3/m3)')
            DO 55 J=1,NYR
                AUX=0.0D0
                IF(QO(J,K).NE.0.0D0)AUX=QW(J,K)/QO(J,K)
                WRITE(4,60)J,PO(1,J,K,1)/C2,QO(J,K)*C1,QW(J,K)*C1,AUX
60  FORMAT(7X,I2,1X,4(F10.5,2X))
55  CONTINUE
            WRITE(4,35)
35  FORMAT(1X)
61  CONTINUE
C-----
C Imprime o perfil vertical de saturacao da primeira fatia do 4 diedro
C-----

```

```

      WRITE(9,880)TEMPO/C1
880  FORMAT('tempo=',F10.2,'      parametros abaixo:dvpr,sw')
      JJ=2*NQL+NQS+NQI/2+1
      DO 870 I=1,NXR
          DELSW=SW(I,JJ,1,1)-0.12D0
          WRITE(9,890)-RB(I),SW(I,JJ,1,1)*100.0
890  FORMAT(2(F10.2,2X))
870  CONTINUE
C
      RETURN
      END

```

```

C-----
C
C SUBROTINA DE IMPRESSAO DE RESULTADOS PARA GRAFICOS
C-----

```

```

C
      SUBROUTINE IMPREG
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      PARAMETER(NXM=08,NYM=05,NZM=04)
      COMMON/ACUM/QOAC,QGAC,QWAC,VOORR,VWORR,VOORC,VWORC,BNPR,BWPR,
      *      BNPC,BWPC,PWFA
      COMMON/VAZAO/QO,QOT,QG,QGT,QW,QWT,DQOP,DQOS,
      *      DQWP,DQWS,RWO,RGO,PWF,TEMPO
      DIMENSION QO(NYM,NZM),QG(NYM,NZM),QW(NYM,NZM),DQOP(NYM,NZM),
      *      DQOS(NYM,NZM),DQWP(NYM,NZM),DQWS(NYM,NZM)
C
      C1=86400.0D0
      C2=1.0D+6
      WRITE(5,10)TEMPO/C1,C1*QOT,C1*QWT,QGT*C1,PWF/C2
      WRITE(7,20)TEMPO/C1,QOAC,QWAC,QGAC
      WRITE(8,30)TEMPO/C1,QWT/QOT,QGT/QOT
      10  FORMAT(E10.4,3X,E10.4,3X,E10.4,3X,E10.4,3X,E10.4)
      20  FORMAT(E10.4,3X,E10.4,3X,E10.4,3X,E10.4)
      30  FORMAT(E10.4,3X,E10.4,3X,E10.4)
C
      RETURN
      END

```

```

C-----
C
C SUBROTINA DE IMPRESSAO DOS DADOS DO TESTE DE FORMACAO
C-----

```

```

C
      SUBROUTINE IMPRET(DT,OPT,L)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      PARAMETER(NXM=08,NYM=05,NZM=04)
      COMMON/NCEL/DBDX,DBDY,DBDZ,WRC,NBLOC,NXC,NYC,NZC,NBP,
      *      NXR,NYR,NZR,NQL,NQS,NQI,IP,JP,KP
      COMMON/TS/PO,SO,SG,SW
      COMMON/MALHA/ZB,RB,VPOR,DRO,WX,WY,WZ,FFX,FFY,FFZ,DTETA
      COMMON/VAZAO/QO,QOT,QG,QGT,QW,QWT,DQOP,DQOS,
      *      DQWP,DQWS,RWO,RGO,PWF,TEMPO
      COMMON/ACUM/QOAC,QGAC,QWAC,VOORR,VWORR,VOORC,VWORC,BNPR,BWPR,
      *      BNPC,BWPC,PWFA
      COMMON/IMP/TIMPR,TIMPG,IRIMP
      DIMENSION PO(0:NXM+1,0:NYM+1,0:NZM+1,2),SO(NXM,NYM,NZM,2),
      *      SG(NXM,NYM,NZM,2),SW(NXM,NYM,NZM,2)
      DIMENSION ZB(0:NXM+1,NYM,2),RB(NXM+1),VPOR(NXM,NYM,NZM,2),
      *      FFX(NXM+1,NYM,NZM,2),FFY(NXM+1,NYM,NZM,2),
      *      FFZ(NXM+1,NYM,NZM+1,2),DTETA(NYM+1)
      DIMENSION QO(NYM,NZM),QG(NYM,NZM),QW(NYM,NZM),DQOP(NYM,NZM),
      *      DQOS(NYM,NZM),DQWP(NYM,NZM),DQWS(NYM,NZM)

```

```

C CHARACTER*1 OPT,SIT*5
C1=3600.0D0
C2=1.0D+6
C3=24.0D0*C1
IF(OPT.EQ.'F') THEN
  SIT='FLUXO'
ELSE
  SIT='ESTAT'
ENDIF
WRITE(11,5)SIT,NXC,NYC,NZC,TEMPO/C1,DT/C1,C3*QOT,C3*QWT,RWO,PWF/C2
5 FORMAT(' ',/,15X,'RESULTADOS DO SIMULADOR ACTIOW4',/,
*5X,'FASE DE TESTE DE FORMACAO - FINAL ',A5,/,
*5X,'N# DE BLOCOS NA DIRECAO X .....=',I3,/,
*5X,'N# DE BLOCOS NA DIRECAO Y .....=',I3,/,
*5X,'N# DE BLOCOS NA DIRECAO Z .....=',I3,/,
*5X,'TEMPO ACUMULADO (horas).....=',E10.4,/,
*5X,'INCREMENTO DE TEMPO (horas).....=',F10.4,/,
*5X,'VAZAO DE OLEO (m3/dia).....=',F10.4,/,
*5X,'VAZAO DE GAS (m3/dia).....=',E10.4,/,
*5X,'RAZAO GAS OLEO (m3/dia / m3/dia)...=',E10.4,/,
*5X,'PRESSAO Pwf NO POCO (MPa).....=',F10.4)
DO 10 K=1,NZR
  WRITE(11,15)K
15  FORMAT(/,5X,'CAMADA:',I3)
  WRITE(11,20)
20  FORMAT(/,5X,'GRID RADIAL',/,5X,' BLOCO ',5X,'PRESSAO(MPA)',
*      3X,'SAT OLEO',4X,'SAT GAS',5X,'SAT AGUA')
  DO 10 I=1,NXR
    DO 10 J=1,NYR
      WRITE(11,25)I,J,PO(I,J,K,1)/C2,SO(I,J,K,1),
*          SG(I,J,K,1),SW(I,J,K,1)
25  FORMAT(5X,2(I2,1X),5(F10.4,2X))
10  CONTINUE
C
DO 30 K=1,NZC
  WRITE(11,35)K
35  FORMAT(/,5X,'CAMADA:',I3)
  WRITE(11,40)
40  FORMAT(/,5X,'GRID CARTESIANO',/,5X,' BLOCO ',5X,'PRESSAO(MPA)',
*      3X,'SAT OLEO',4X,'SAT GAS',5X,'SAT AGUA')
  DO 30 I=1,NXC
    DO 30 J=1,NYC
      WRITE(11,55)I,J,PO(I,J,K,2)/C2,SO(I,J,K,2),
*          SG(I,J,K,2),SW(I,J,K,2)
55  FORMAT(5X,2(I2,1X),5(F10.4,2X))
30  CONTINUE
  RETURN
END
-----
C
C SUBROTINA DE INTERPOLACAO POR PESQUISA BINARIA E CALC. DA 1# DERIVADA
C
SUBROUTINE INTER(X,Y,N,XO,YO,YOP)
REAL*8 X(N),Y(N),XO,YO,YOP
INTEGER IN,MAX,MEIO
C
IF((XO.LT.X(1)).OR.(XO.GT.X(N))) THEN
  PRINT*, 'NAO FOI POSSIVEL INTERPOLAR XO=',XO
  DO 5 J=1,N
    PRINT*,J,' X=',X(J),' Y=',Y(J)
5  CONTINUE

```

[illegible]