

UNIVERSIDADE ESTADUAL DO NORTE FLUMINENSE
LABORATÓRIO DE ENGENHARIA E EXPLORAÇÃO DE
PETRÓLEO

PROJETO ENGENHARIA
DESENVOLVIMENTO DO SOFTWARE
APLICAÇÃO DE CORRELAÇÕES PARA CÁLCULO DE
PARÂMETROS DE RESERVATÓRIO A PARTIR DE
PROPRIEDADES DOS FLUIDOS – MODELAGEM *BLACKOIL*
TRABALHO DA DISCIPLINA PROGRAMAÇÃO PRÁTICA

VERSÃO 1.0
JOÃO BOSCO MACIEL FILHO
SUSANE CHELLI CAIRES GOIS
Prof. André Duarte Bueno

MACAÉ - RJ
Julho - 2017

Sumário

1	Introdução	1
1.1	Escopo do problema	1
1.2	Objetivos	2
2	Especificação	3
2.1	Nome do sistema/produto	3
2.2	Especificação	3
2.2.1	Requisitos funcionais	4
2.2.2	Requisitos não funcionais	4
2.3	Casos de uso	4
2.3.1	Diagrama de caso de uso geral	5
2.3.2	Diagrama de caso de uso específico	5
3	Elaboração	8
3.1	Análise de domínio	8
3.2	Formulação teórica	9
3.2.1	Correlação Standing (1947)	10
3.2.2	Correlação Vasquez & Beggs (1976)	11
3.2.3	Correlação de Sutton	12
3.2.4	Correlação Papay	12
3.2.5	Correlação Beggs & Brill	12
3.2.6	Correlação <i>Black-Oil</i> - Fator Volume de Formação do gás - Bg . . .	13
3.3	Identificação de pacotes – assuntos	13
3.4	Diagrama de pacotes – assuntos	14
4	AOO – Análise Orientada a Objeto	15
4.1	Diagramas de classes	15
4.1.1	Dicionário de classes	15
4.2	Diagrama de sequência – eventos e mensagens	20
4.2.1	Diagrama de sequência geral	20
4.2.2	Diagrama de sequência específico	20

4.3	Diagrama de comunicação – colaboração	21
4.4	Diagrama de máquina de estado	21
4.5	Diagrama de atividades	22
5	Projeto	23
5.1	Projeto do sistema	23
5.2	Projeto orientado a objeto – POO	24
5.3	Diagrama de componentes	27
5.4	Diagrama de implantação	29
6	Implementação	30
6.1	Código fonte	30
7	Teste	69
7.1	Teste 1: Entrada de dados pelo teclado	69
7.2	Teste 2: Estrada de dados por arquivo de disco	71
8	Documentação	77
8.1	Documentação do usuário	77
8.1.1	Como rodar o software	77
8.2	Documentação para desenvolvedor	79
8.2.1	Dependências	79
8.2.2	Como gerar a documentação usando doxygen	80
9	Sugestões para Trabalhos Futuros	82

Capítulo 1

Introdução

No presente projeto de engenharia desenvolve-se o software CCPRMB (Correlações para Cálculo de Parâmetros de Reservatório – Modelagem *BlackOil*), um software aplicado a engenharia de petróleo e que utiliza o paradigma da orientação a objetos.

Neste projeto são implementadas equações de correlações empíricas baseadas na abordagem de modelamento de fluido *BlackOil*. Por meio dessas correlações é possível calcular parâmetros de um reservatório a partir de variáveis de entrada referentes às características dos fluidos produzidos.

1.1 Escopo do problema

O conhecimento das propriedades físico-químicas dos fluidos no reservatório é necessário para a determinação da quantidade de óleo presente e como este irá se comportar quanto à produção, entre outros aspectos. O gerenciamento do reservatório e implementação de estratégias de produção dependem diretamente dessas condições.

São necessários modelos matemáticos para descrever as propriedades dos fluidos e as interações entre elas. Cada parâmetro é calculado por uma equação (correlação) a partir de variáveis de entrada. O método de modelagem de fluido *BlackOil* aplica-se bem quando os parâmetros estão dentro das faixas utilizadas nos experimentos laboratoriais para o desenvolvimento da correlação porém, a extrapolação para faixas fora daquelas utilizadas nos experimentos laboratoriais deve ser feita com cautela, e pode-se gerar erros consideráveis.

1.2 Objetivos

Os objetivos deste projeto de engenharia são:

- Objetivo geral:
 - Utilizar correlações empíricas para cálculo de parâmetros de um reservatório a partir de propriedades dos fluidos presentes no mesmo – Modelagem *BlackOil*.
 - Calcular propriedades dos fluidos a partir de dados externos.
 - Calcular parâmetros de um reservatório utilizando correlações específicas.
- Objetivos específicos:
 - Modelar física e matematicamente o problema
 - Calcular Pressão de Bolha - P_b
 - Calcular Razão de Solubilidade - R_s
 - Calcular Fator Volume de Formação do óleo - B_o
 - Calcular Fator Volume de Formação do gás - B_g
 - Calcular Fator de Compressibilidade do gás - Z
 - Calcular Compressibilidade - C_o
 - Utilizar a correlação de Standing
 - Utilizar a correlação de Vasquez & Beggs
 - Utilizar a correlação de Papay
 - Utilizar a correlação de Beggs & Brill
 - Gerar gráficos a partir do software externo Gnuplot

Capítulo 2

Especificação

Apresenta-se neste capítulo do projeto de engenharia a concepção, a especificação do sistema a ser modelado e desenvolvido.

2.1 Nome do sistema/produto

Nome	CCPRMB
Componentes principais	Sistema para cálculo de parâmetros de reservatório.
Missão	Calcular parâmetros de reservatório a partir de correlações de modelagem <i>Black-Oil</i> .

2.2 Especificação

Este programa de engenharia que tem por objetivo calcular e descrever as propriedades dos fluidos e as iterações entre elas para analisar a produção de um reservatório de petróleo utilizando correlações empíricas a partir da abordagem de modelagem *Black-Oil*.

O Programa deve ler os dados do poço armazenados no disco, e a partir deles, calcular as propriedades desejadas para diferentes correlações, como por exemplo Standing, Vasquez e Beggs, Beggs e Brill etc, [Vieira,].

O usuário poderá, caso prefira, digitar os valores de pressão manualmente para verificar as propriedades para pontos de pressão desejados, bom como alterar os valores de parâmetros do óleo e das correlações.

O software será desenvolvido utilizando o conceito de programação orientada a objeto e utilizará interface em modo texto e inicialmente irá plotar gráficos que serão gerados com um programa externo.

2.2.1 Requisitos funcionais

Apresenta-se a seguir os requisitos funcionais.

RF-01	Os cálculos devem ser feitos utilizando-se correlações de modelagem <i>Black-Oil</i> .
RF-02	O usuário deverá ter liberdade para escolher qual parâmetro deseja calcular.
RF-03	Deve permitir o carregamento de arquivos de disco a serem utilizados pelo software.
RF-04	Deve permitir a escolha da correlação a ser utilizada.
RF-05	O software plotará os resultados em gráficos. Os gráficos poderão ser salvos como imagem ou ter seus dados exportados como texto.

2.2.2 Requisitos não funcionais

RNF-01	O programa deverá ser multi-plataforma, podendo ser executado em <i>Windows</i> , <i>GNU/Linux</i> ou <i>Mac</i> .
---------------	--

2.3 Casos de uso

Tabela 2.1: Exemplo de caso de uso

Nome do caso de uso:	Cálculo de um parâmetro de reservatório.
Resumo/descrição:	Cálculo de um parâmetro de caracterização de reservatório em determinadas condições.
Etapas:	<ol style="list-style-type: none"> 1. Inserir dados de entrada. 2. Escolher qual(is) parâmetro(s) deseja calcular. 3. Escolher qual correlação utilizar. 4. Calcular o(s) parâmetro(s) escolhido(s). 5. Gerar gráficos.
Cenários alternativos:	Um cenário alternativo envolve uma entrada errada do usuário (por exemplo, valores de pressão negativos ou fator volume de formação menor que a unidade). O software apresentará uma mensagem quando essas premissas forem detectadas.

2.3.1 Diagrama de caso de uso geral

O diagrama de caso de uso geral da Figura 2.1 mostra o usuário acessando os sistemas de ajuda do software, calculando um parâmetro de caracterização de reservatório. Este diagrama de caso de uso ilustra as etapas a serem executadas pelo usuário ou sistema, a interação do usuário com o sistema.

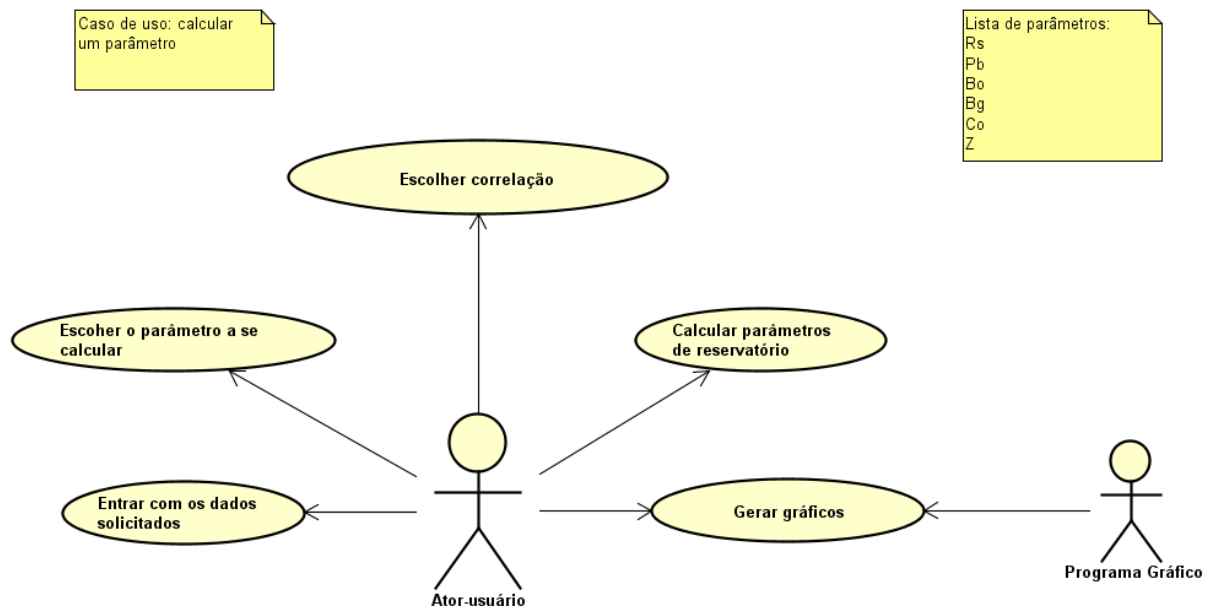


Figura 2.1: Diagrama de caso de uso geral – Calcular um parâmetro

2.3.2 Diagrama de caso de uso específico

O caso de uso “Calcular parâmetro de reservatório” descrito na Figura 2.1 e na Tabela 2.1 é detalhado na Figura 2.2 para o caso específico da “Razão de Solubilidade Rs” a partir da correlação de Vasquez & Beggs. O usuário definirá o parâmetro Rs que deseja calcular assim como a correlação Vasquez & Beggs a ser utilizada e após o cálculo, o software gerará o gráfico de Rs versus a Pressão com os resultados obtidos utilizando um sistema externo, como o software *gnuplot*. Este diagrama de caso de uso ilustra as etapas a serem executadas pelo usuário ou sistema, a interação do usuário com o sistema.

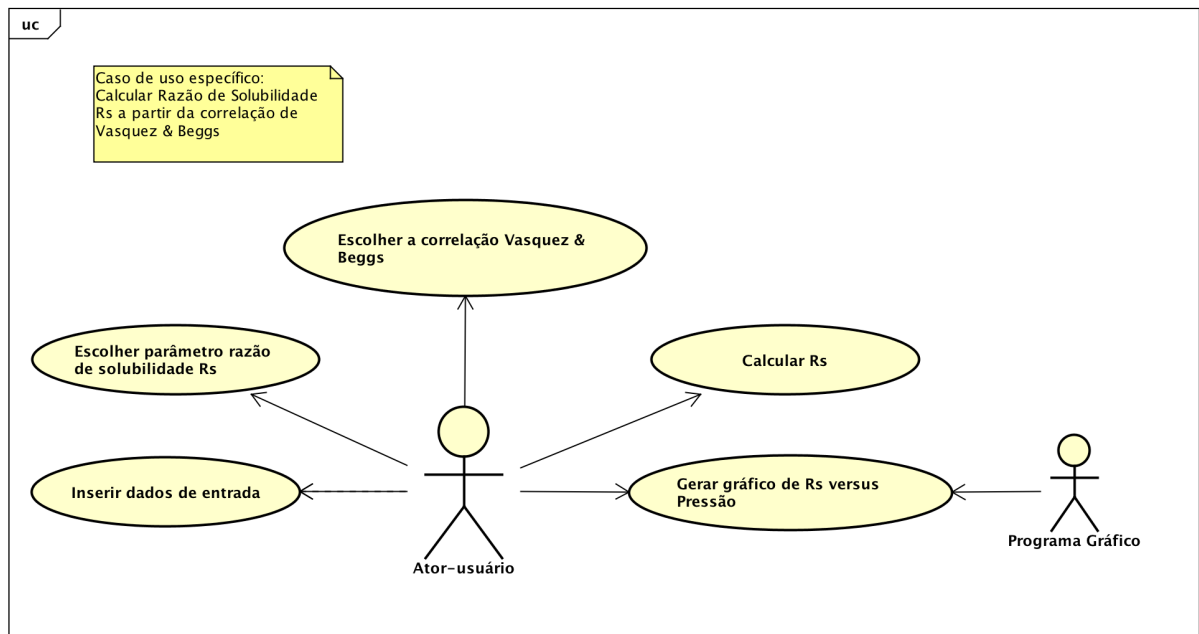


Figura 2.2: Diagrama de caso de uso específico – Cálculo da razão de solubilidade.

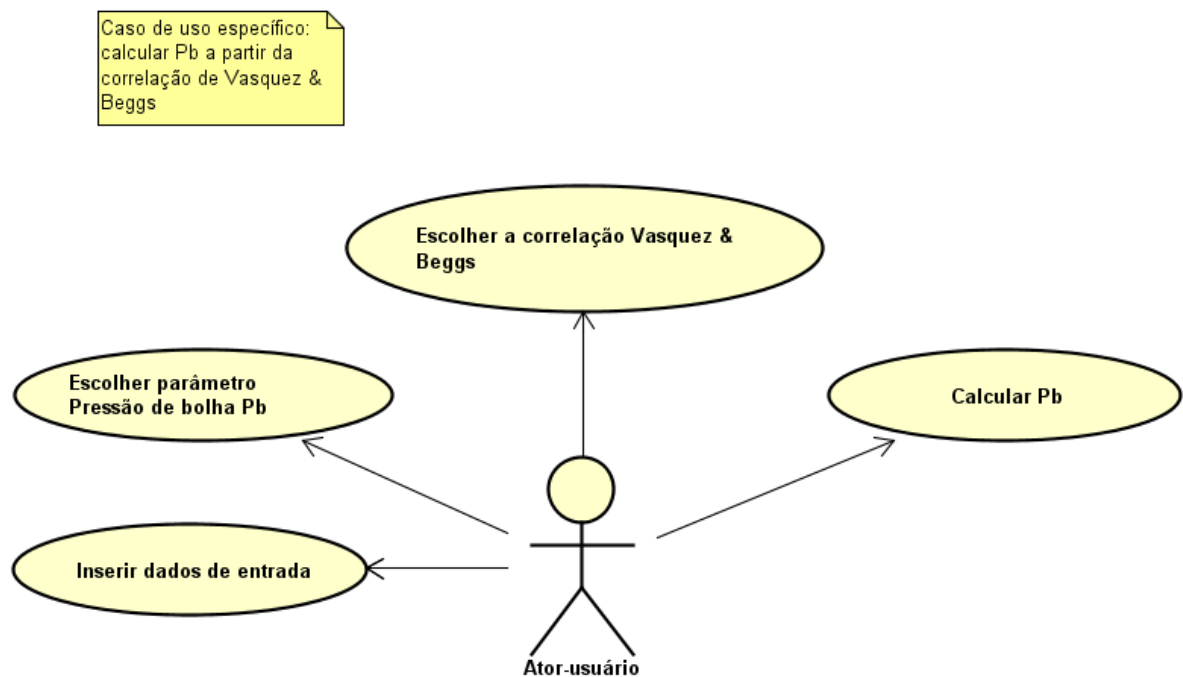


Figura 2.3: Diagrama de caso de uso específico – Cálculo da pressão bolha

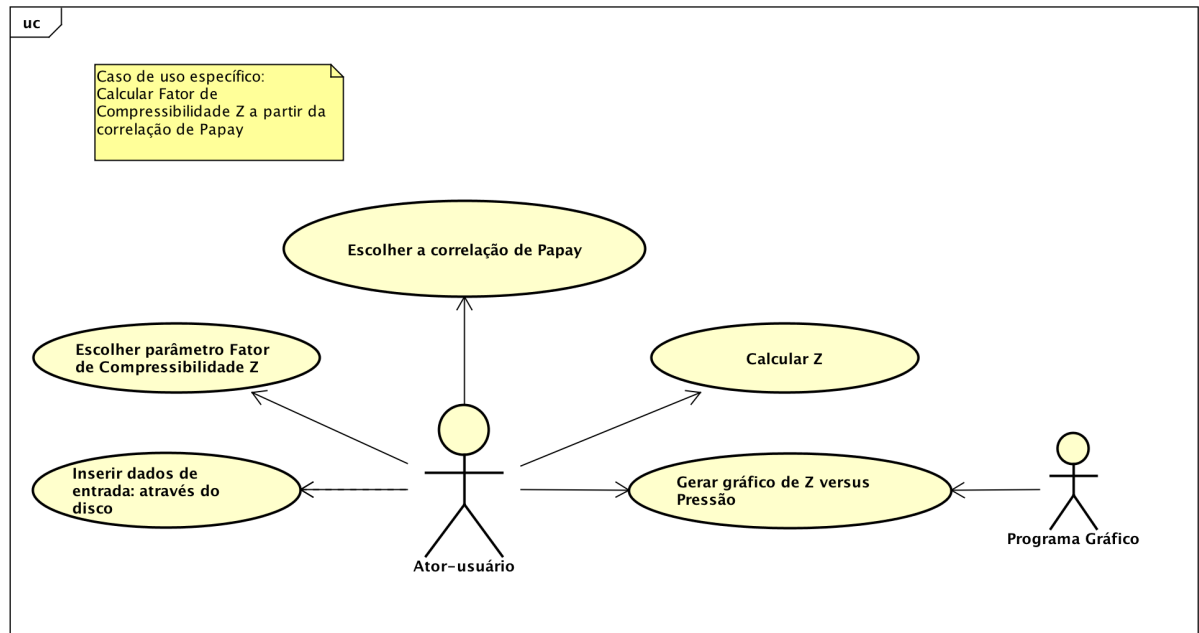


Figura 2.4: Diagrama de caso de uso específico – Cálculo do fator de compressibilidade Z

Capítulo 3

Elaboração

Depois da definição dos objetivos, da especificação do software e da montagem dos primeiros diagramas de caso de uso, neste capítulo será apresentada a elaboração, que envolve o estudo de conceitos relacionados ao sistema a ser desenvolvido, a análise de domínio e a identificação de pacotes. Esse processo é feito através de pesquisas bibliográficas e entrevistas, que nos mostram o que é necessário para a formulação do programa. Uma análise dos requisitos para o funcionamento do programa será feita a fim de avaliar as condições necessárias para o desenvolvimento de um sistema útil, que satisfaça as necessidades requeridas e seja passível de posterior extensão.

3.1 Análise de domínio

O programa que será desenvolvido trata dos parâmetros que determinam o comportamento do reservatório em termos produtivos, uma área da engenharia de reservatório muito estudada e frequentemente utilizada como uma importante ferramenta na solução de diversos problemas da indústria. Gerenciamento de reservatórios, por definição, é uma ciência que utiliza elementos da geologia e da engenharia de petróleo para predizer o comportamento do óleo e do gás natural nas formações rochosas sub-superficiais.

A Figura 3.2 mostra as diferentes áreas relacionadas ao software. Essas vão desde elementos relacionados aos parâmetros advindos do estudo geológico do reservatório e tem implicações na análise, otimização da recuperação e outros aspectos relevantes à produção.

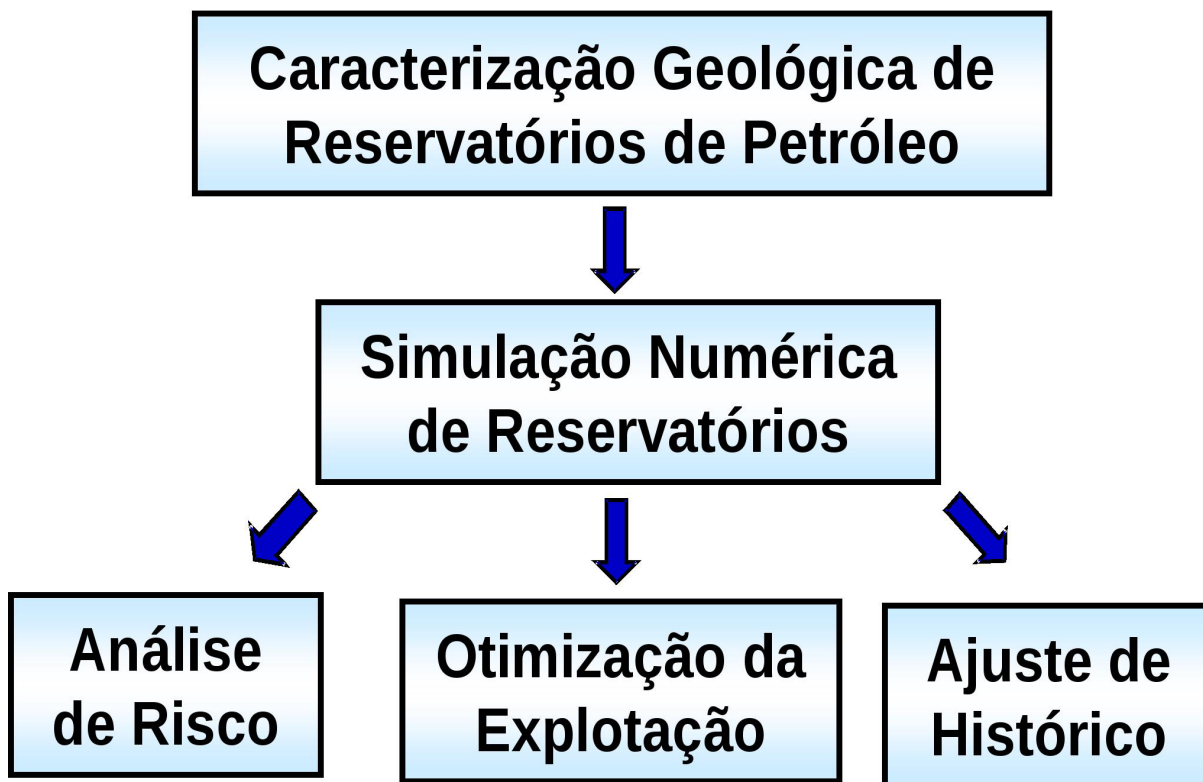


Figura 3.1: Análise de domínio. <disponível em: <http://www.unisim.cepetro.unicamp.br/br/2-uncategorised/21-caracterizacao-de-reservatorios>>

O software calculará, através do uso de correlações empíricas, parâmetros importantes na análise do reservatório do ponto de vista comportamental dos fluidos presentes no mesmo. Isso implica diretamente na tomada de decisões quanto a engenharia de reservatório a ser empregada.

3.2 Formulação teórica

Conforme a produção de petróleo avança e a pressão do reservatório cai, empresas petrolíferas utilizam modelos matemáticos para simularem como a permeabilidade e as saturações de óleo e gás irão se comportar [Thomas, 2001]. O uso de modelos simplificados para a realização dessa previsão se deve, na maioria das vezes, a não existência de alguns dados devido à dificuldade de obtê-los no início de um projeto de poço.

Para simular esse comportamento dos parâmetros de um reservatório ao longo da produção de um campo há diversos modelos matemáticos, uns mais abrangentes, outros mais específicos [Leite, 2016].

Em simulação, chama-se de modelagem *blackoil* aquela que pode ser assumida uma composição constante para o óleo durante sua vida produtora. As correlações “*blackoil*” foram desenvolvidas especificamente para sistemas de óleo cru / gás / água e são muito úteis para prever o comportamento das fases no fluxo de um poço de petróleo

[Banzer, 1996]. Quando usadas em conjunto com as opções de calibração, as correlações *blackoil* podem produzir dados de comportamento de fases precisos, a partir de um mínimo de dados de entrada. Elas são particularmente convenientes em estudos de “gás lift”, onde os efeitos da variação da razão gás óleo (RGO) e corte de água estão sob investigação. A modelagem *blackoil* é largamente utilizada e a grande maioria dos estudos de simulações de reservatório e de escoamento adotam esta modelagem [MACHADO, 2002].

BRILL e MUKHERJEE (1999) definem *blackoil* como um termo que se refere a qualquer fase líquida que contenha gás dissolvido, como hidrocarbonetos, por exemplo [Velarde, 1997]. Esses óleos são tipicamente escuros e tem densidades menores que 40^o API. Sua principal característica, no entanto, é que praticamente não apresenta variações na sua composição em um envelope de duas fases, sendo por isso dito como um modelo de composição constante.

Esse modelo tem capacidade de simular todos os mecanismos de produção, incluindo gás em solução, capa de gás e influxo de água, com ou sem injeção de água ou gás.

No entanto, é de fundamental importância a escolha adequada do modelo a ser utilizado para modelar um campo, de acordo com as propriedades e características já conhecidas, assim como validar o equacionamento do mesmo. A caracterização *blackoil* é determinada através de correlações empíricas e equações [Banzer, 1996].

3.2.1 Correlação Standing (1947)

- Dados experimentais [Leite, 2016]:
 - 22 misturas de óleo-cru/gás natural de campos da Califórnia
 - 105 pontos experimentais
 - pb (pressão de bolha): 130 a 7000 psia
 - T (temperatura): 100 a 258 °F
 - API: 16,5 a 63,8 °API
 - dg (densidade do gás na condição standard): 0,59 a 0,95
 - Rsb (gás em solução à pressão do ponto de bolha): 20 a 1425 scf/STB

A Correlação de Standing permite o cálculo dos seguintes parâmetros do reservatório:

Pressão de Saturação ou Pressão de Bolha

$$p_b = 18.2 \left[\frac{1}{10^{(0.0125API - 0.00091T)}} \left(\frac{Rs_b}{d_g} \right)^{0.83} - 1.4 \right] \quad (3.1)$$

Razão de solubilidade

$$Rs = d_g \left[\left(\frac{p}{18.2} + 1.4 \right) 10^{(0.0125API - 0.00091T)} \right]^{1.2048} \quad (3.2)$$

Fator Volume de Formação do óleo

$$B_{ob} = 0.972 + 1.47 \times 10^{-4} \left[R_{sb} \left(\frac{d_g}{d_o} \right)^{0.5} + 1.25T \right]^{1.175} \quad (3.3)$$

3.2.2 Correlação Vasquez & Beggs (1976)

- Dados experimentais [Leite, 2016]:
 - 600 tipos de óleo diferentes de campos de todo o mundo
 - ~6000 pontos experimentais
 - pb (pressão de bolha): 15 a 6055 psia
 - T (temperatura): 70 a 295 °F
 - API: 15,3 a 59,5 °API
 - dg (densidade do gás na condição standard): 0,511 a 1,351
 - Rsb (gás em solução à pressão do ponto de bolha): 0 a 2199 scf/STB

A Correlação de Vasquez & Beggs permite o calculo dos seguintes parâmetros do reservatório:

Pressão de Saturação ou Pressão de Bolha

$$p_b = \left[\frac{R_{sb}}{C_1 * d_{gs} * e^{\frac{C_3 API}{(T+459,67)}}} \right]^{\frac{1}{C_2}} \quad (3.4)$$

Razão de solubilidade

$$Rs = C_1 * d_{gs} * p^{C_2} * e^{\frac{C_3 API}{(T+459,67)}} \quad (3.5)$$

Tabela 3.1: Coeficientes C da Correlação - Vasquez & Beggs

Coeficiente	API > 30	API ≤ 30
C1	0.0178	0.0362
C2	1.1870	1.0937
C3	23.931	25.742

Fator Volume de Formação do óleo

$$B_{ob} = 1 + A_1 * R_{sb} + A_2(T - 60) \left(\frac{API}{d_{gs}} \right) + A_3 * R_{sb} * (T - 60) \left(\frac{API}{d_{gs}} \right) \quad (3.6)$$

Tabela 3.2: Coeficientes A da Correlação - Vasquez & Beggs

Coeficiente	API > 30	API ≤ 30
A1	4,670 x10 ⁻⁴	4,677 x10 ⁻⁴
A2	1,100 x10 ⁻⁵	1,751 x10 ⁻⁵
A3	1,377 x10 ⁻⁹	-1,811 x10 ⁻⁸

$$B_o = B_{ob} e^{c_o(p_b - p)} \quad (3.7)$$

Compressibilidade isotérmica do óleo

$$c_o = \frac{(-1433 + 5Rs + 17.2T - 1180d_{gs} + 12.61API)}{p * 10^5} \quad (3.8)$$

3.2.3 Correlação de Sutton

A correlação de Sutton (2005) fornece as propriedades pseudo-críticas para gás associado que posteriormente são usadas para obter as condições de pressão e temperatura reduzidas e, conseqüentemente, o fator de compressibilidade Z [Leite, 2016].

$$P_{pc} = 671,1 + (14 - 34,3 * dg) * dg \quad (3.9)$$

$$T_{pc} = 120,1 + (429 - 62,9 * dg) * dg \quad (3.10)$$

3.2.4 Correlação Papay

- Propriedades do óleo [Leite, 2016]:

$$- \text{Range de validade da equação: } \begin{cases} 0.2 \leq p_r \leq 15 & 1.2 \leq T_r \leq 3 \end{cases}$$

A Correlação de Papay permite o cálculo dos seguintes parâmetros do reservatório:

Fator de compressibilidade Z

$$Z = 1 - \frac{3,52 * p_r}{10^{0,9813 * T_r}} + \frac{0,274 * p_r^2}{10^{0,8157 * T_r}} \quad (3.11)$$

3.2.5 Correlação Beggs & Brill

A Correlação de Papay permite o cálculo dos seguintes parâmetros do reservatório [Leite, 2016]:

Fator de compressibilidade Z

$$Z = A + \frac{1 - A}{e^B} + C * p_r^D \quad (3.12)$$

$$A = 1,39 * (T_r - 0,92)^{0,5} - 0,36 * T_r - 0,1$$

$$B = (0,62 - 0,23 * T_r) * p_r + \left(\frac{0,066}{T_r - 0,86} - 0,037\right) * p_r^2 + \left(\frac{0,32}{10^{9*(T_r-1)}}\right) * p_r^6$$

$$C = 0,132 - 0,32 * \log T_r$$

$$D = 10^{(0,3106 - 0,49 * T_r + 0,1824 * T_r^2)}$$

3.2.6 Correlação *Black-Oil* - Fator Volume de Formação do gás - Bg

$$B_g = \frac{0.00504 * Z * (T + 459.67)}{p} \quad (3.13)$$

3.3 Identificação de pacotes – assuntos

A partir da análise dos modelos apresentados, pode-se identificar os seguintes assuntos/pacotes:

- Pacote Banco de dados: Composto por arquivos de disco com dados de poços de petróleo necessários para o cálculo dos parâmetros requeridos.
- Pacote Correlações: Calcula os parâmetros termodinâmicos usando as correlações listadas, se mais de uma estiver disponível para o mesmo parâmetro, o usuário poderá escolher qual correlação deseja usar.
- Pacote Gráficos: Usando um software externo (GNUPLOT), será possível gerar gráficos relacionando determinado parâmetro com a variação de pressão.
- Pacote Simulador: Relaciona os pacotes acima, sendo responsável pela criação e destruição de objetos, assim como interagir com o usuário através de um interface via texto para definir todas as ações a serem tomadas.

3.4 Diagrama de pacotes – assuntos

Na Figura 3.2 está representado o diagrama de pacotes, mostrando a dependência entre as partes do sistema.

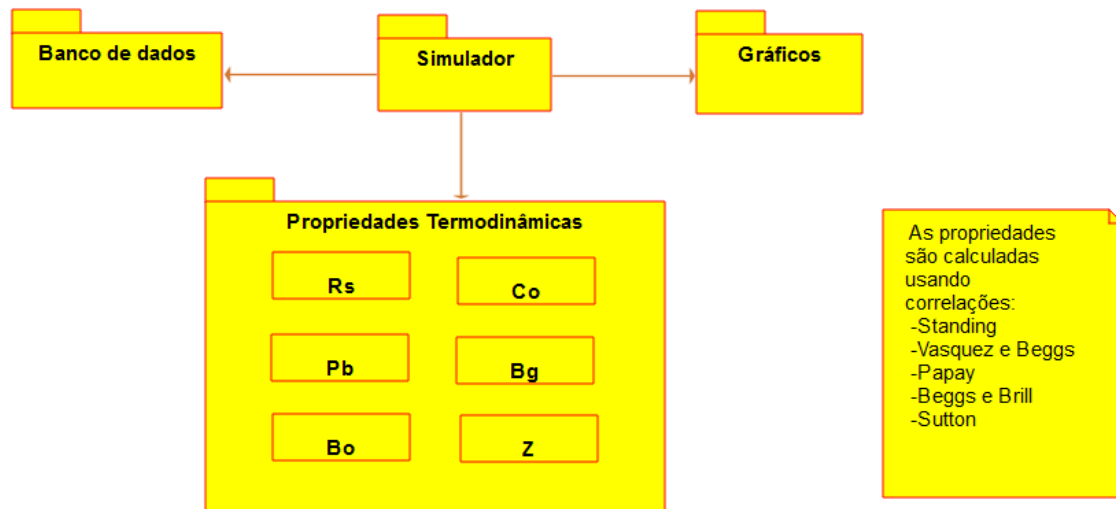


Figura 3.2: Diagrama de Pacotes

Capítulo 4

AOO – Análise Orientada a Objeto

A AOO – Análise Orientada a Objeto utiliza algumas regras para identificar os objetos de interesse, as relações entre os pacotes, as classes, os atributos, os métodos, as heranças, as associações, as agregações, as composições e as dependências.

4.1 Diagramas de classes

O diagrama de classes é apresentado na Figura 4.1.

4.1.1 Dicionário de classes

- Classe CSimulador: classe responsável
- Classe CFluidoBlackOil: classe responsável por receber as escolhas do usuário sobre qual forma de entrada de dados, sendo responsável também por receber e armazenar estes dados que serão usados para calcular os parâmetros de reservatório.
- CPressaoBolha: Classe responsável pela chamada das classes de correlações relacionadas aos cálculos de pressão de bolha. A passagem dos parâmetros e chamada dos métodos é feita por ela.
- CRazaoSolubilidade: Classe responsável pela chamada das classes de correlações relacionadas aos cálculos de razão de solubilidade. A passagem dos parâmetros e chamada dos métodos é feita por ela, bem como a plotagem do gráfico de Rs com relação a variação da pressão.
- CFatorVolumeFormacaoOleo: Classe responsável pela chamada das classes de correlações relacionadas aos cálculos de fator volume de formação do óleo. A passagem dos parâmetros e chamada dos métodos é feita por ela e ao final a mesma plotará o gráfico de Bo com relação a variação da pressão.

- CCompresOleo: Essa classe realiza a chamada da classe da correlação de Vasquez e Beggs para cálculo da compressibilidade do óleo. A passagem dos parâmetros e chamada do método da classe derivada é feita por ela e ao final a mesma plotará o gráfico de Co com relação a variação da pressão.
- CFatorCompZ: Classe responsável pela chamada das classes de correlações relacionadas aos cálculos de fator Z. A passagem dos parâmetros e chamada dos métodos é feita por ela. Ao final, plotará o gráfico de Z com a variação da pressão.
- CFatorVolumeFormacaoGas: Essa classe realiza o do fator volume de formação do gás e plotagem do gráfico de Bg com relação a variação da pressão, acionando o Gnuplot.
- Classe CPb_Standing: representa a correlação de Standing para cálculo de Pressão de bolha (Pb).
- Classe CPb_VasquezeBeggs: representa a correlação de Vasquez e Beggs para cálculo de Pressão de bolha (Pb).
- Classe CRs_Standing: representa a correlação de Standing para cálculo de Razão de solubilidade (Rs).
- Classe CRs_VasquezeBeggs: representa a correlação de Vasquez e Beggs para cálculo de Razão de solubilidade (Rs).
- Classe CBo_Standing: representa a correlação de Standing para cálculo de Fator volume de formação (Bo).
- Classe CBo_VasquezeBeggs: rrepresenta a correlação de Vasquez e Beggs para cálculo de Fator volume de formação (Bo).
- Classe CCo_VasquezeBeggs: rrepresenta a correlação de Vasquez e Beggs para cálculo de Compressibilidade isotérmica do óleo subsaturado (Co).
- Classe CFatorZ_BeggsBrill: representa a correlação de Beggs e Brill, com ela poderá ser calculado os parâmetros de Fator de compressibilidade do gás (Z).
- Classe CFatorZ_Papay: representa a correlação de Papay, com ela poderá ser calculado os parâmetros de Fator de compressibilidade do gás (Z).
- Classe CSutton: representa a correlação de Sutton, com ela poderá ser calculado os parâmetros de Pressão pseudo-crítica (Ppc) e Temperatura pseudo-crítica (Tpc) que serão usados nas classes CFatorZ_BeggsBrill e CFatorZ_Papay para o cálculo do Fator Z.

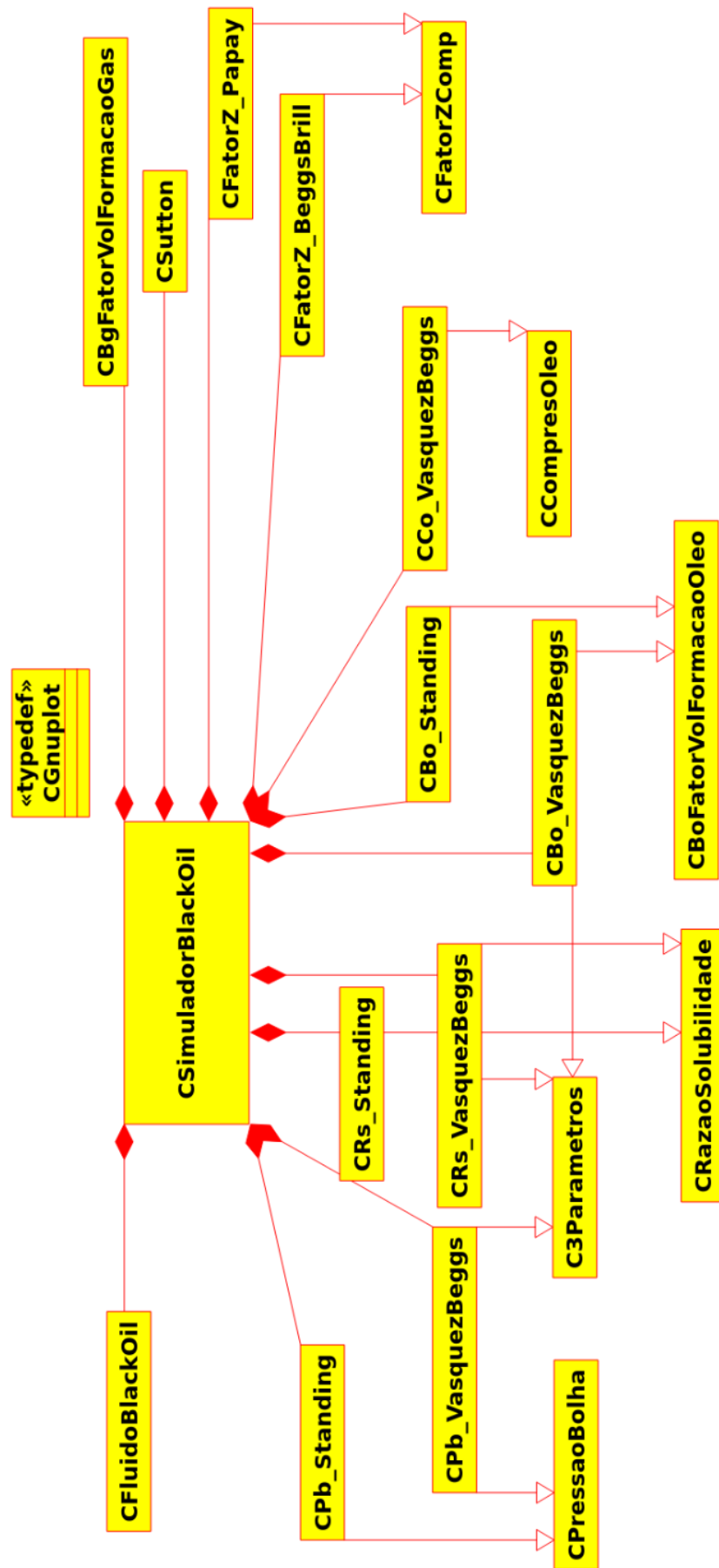


Figura 4.1: Diagrama de classes simplificado

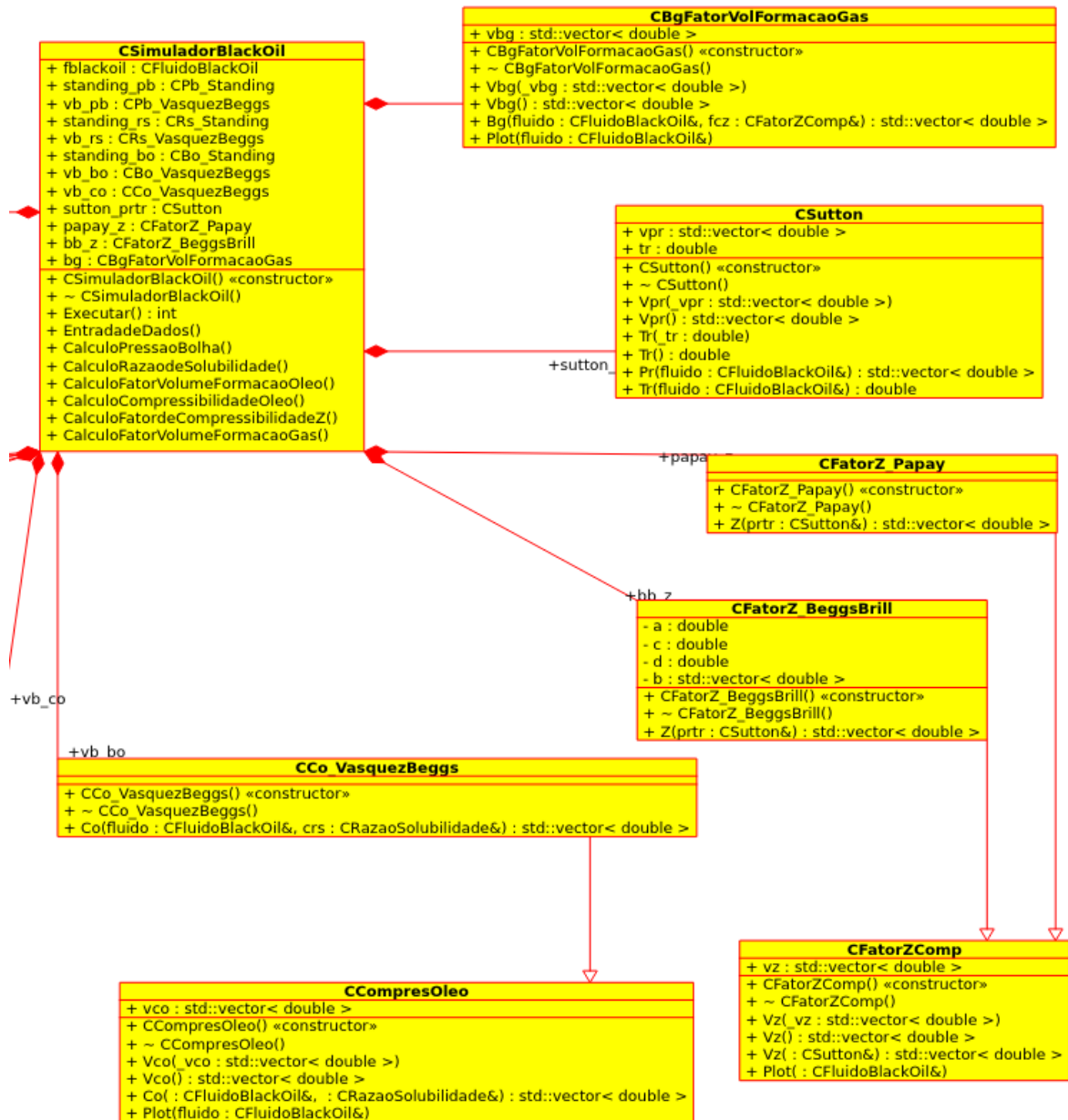


Figura 4.3: Diagrama de classes parte 2

4.2 Diagrama de sequência – eventos e mensagens

O diagrama de sequência enfatiza a troca de eventos e mensagens e sua ordem temporal. Contém informações sobre o fluxo de controle do software.

4.2.1 Diagrama de sequência geral

Veja o diagrama de sequência na Figura 4.4.

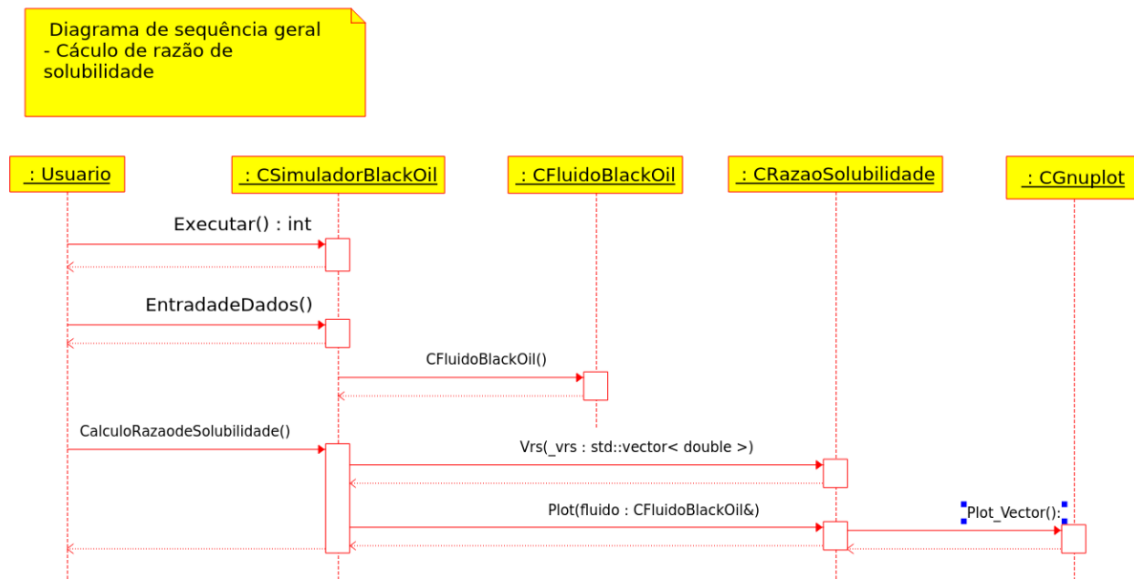


Figura 4.4: Diagrama de sequência geral

4.2.2 Diagrama de sequência específico

Para o caso de um diagrama de sequência específico de cálculo de um parâmetro específico, é necessário que o usuário selecione tanto o parâmetro que deseja calcular, bem como a correlação que deseja e a forma de entrada de dados. Veja o diagrama de sequência na Figura 4.5.

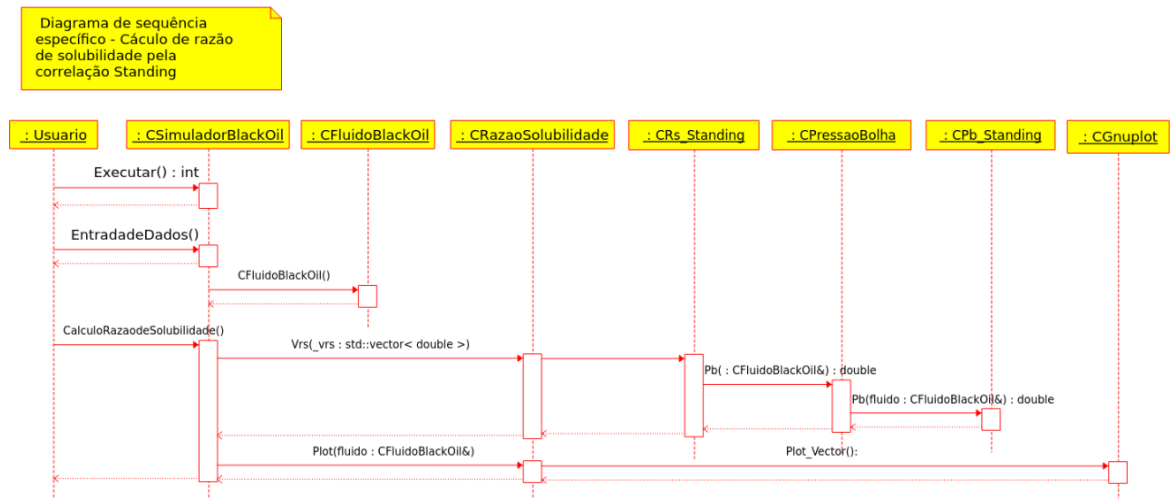


Figura 4.5: Diagrama de sequência - Cálculo de Rs

4.3 Diagrama de comunicação – colaboração

No diagrama de comunicação o foco é a interação e a troca de mensagens e dados entre os objetos.

Veja na Figura 4.3 o diagrama de comunicação mostrando a sequência de cálculo do Fator Z de compressibilidade do gás pela correlação de Standing.

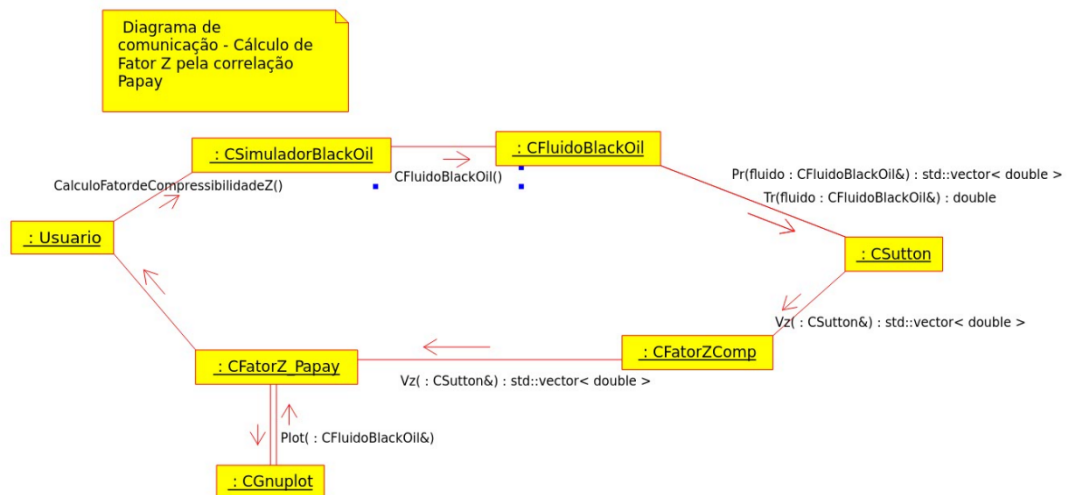


Figura 4.6: Diagrama de comunicação - Cálculo do fator Z

4.4 Diagrama de máquina de estado

Um diagrama de máquina de estado representa os diversos estados que o objeto assume e os eventos que ocorrem ao longo de sua vida ou mesmo ao longo de um processo (histórico

do objeto). É usado para modelar aspectos dinâmicos do objeto.

Veja na Figura 4.7 o diagrama de máquina de estado para o objeto da classe CFatorZ_BeggsBrill.

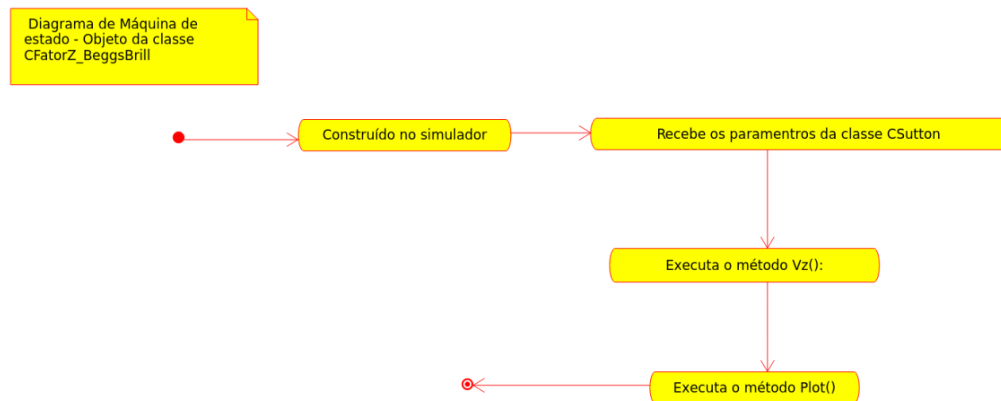


Figura 4.7: Diagrama de máquina de estado: CFatoZ_BeggsBrill

4.5 Diagrama de atividades

Veja na Figura 4.8 o diagrama de atividades correspondente ao procedimento de cálculo do fator volume de formação Bo pela correlação de Vasquez e Beggs.

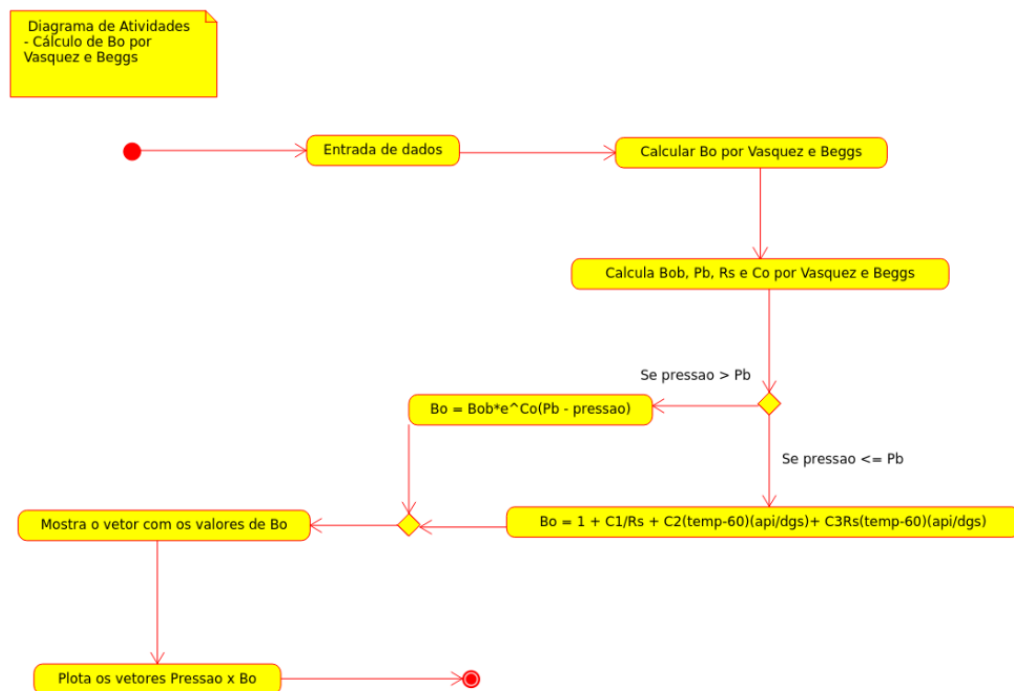


Figura 4.8: Diagrama de atividades: CBo_VasquezBeggs::Pb()

Capítulo 5

Projeto

Neste capítulo do projeto de engenharia veremos questões associadas ao projeto do sistema, incluindo protocolos, recursos, plataformas suportadas, implicações nos diagramas feitos anteriormente, diagramas de componentes e implantação.

5.1 Projeto do sistema

Depois da análise orientada a objeto desenvolve-se o projeto do sistema, qual envolve etapas como a definição dos protocolos, da interface API, o uso de recursos, a subdivisão do sistema em subsistemas, a alocação dos subsistemas ao hardware e a seleção das estruturas de controle, a seleção das plataformas do sistema, das bibliotecas externas, dos padrões de projeto, além da tomada de decisões conceituais e políticas que formam a infraestrutura do projeto.

Deve-se definir padrões de documentação, padrões para o nome das classes, padrões de retorno e de parâmetros em métodos, características da interface do usuário e características de desempenho.

O projeto do sistema é a estratégia de alto nível para resolver o problema e elaborar uma solução.

1. Protocolos

- Neste projeto o software irá se comunicar com o componente externo Gnuplot, que gerará os gráficos escolhidos pelo usuário.
- Será efetuada a entrada de dados via arquivo de texto, formato ascii, com a extensão .dat.

2. Recursos

- O presente programa utilizará o HD, o processador, o teclado, a memória, a tela e os demais componentes internos do computador

- Será utilizado também um arquivo de dados no formato ascii com as informações do fluido do reservatório.

3. Plataformas

- A linguagem usada para o programa será C++, portanto este será multiplataforma, podendo ser executado no Windows, Mac OS X e GNU/Linux.
- O software utilizará a biblioteca externa CGnuplot que permitirá o acesso ao programa Gnuplot. Uma cópia dos arquivos .h e .cpp está disponibilizada.
- O ambiente para montar a interface de desenvolvimento – IDE será o Code-Blocks (Windows) e o Xcode (Mac OS X). O compilador gcc/g++ do Gnu.

5.2 Projeto orientado a objeto – POO

O projeto orientado a objeto é a etapa posterior ao projeto do sistema. Baseia-se na análise, mas considera as decisões do projeto do sistema. Acrescenta a análise desenvolvida e as características da plataforma escolhida (hardware, sistema operacional e linguagem de softwareção). Passa pelo maior detalhamento do funcionamento do software, acrescentando atributos e métodos que envolvem a solução de problemas específicos não identificados durante a análise.

Envolve a otimização da estrutura de dados e dos algoritmos, a minimização do tempo de execução, de memória e de custos. Existe um desvio de ênfase para os conceitos da plataforma selecionada.

Por exemplo, na análise define-se que existe um método para salvar um arquivo em disco, define um atributo nomeDoArquivo, mas não se preocupa com detalhes específicos da linguagem. Já no projeto, inclui-se as bibliotecas necessárias para acesso ao disco, cria-se um objeto específico para acessar o disco, podendo, portanto, acrescentar novas classes àquelas desenvolvidas na análise.

Efeitos do projeto no modelo estrutural

- Aqui são estabelecidos as dependências e restrições do programa de parâmetros de fluidos do reservatório.
 - O programa utiliza o HD, o processador e o teclado do computador;
 - O Software pode ser executado nas plataformas GNU/Linux ou Windows;
 - No Sistema Operacional Windows e GNU/Linux, existe a necessidade de instalação do software Gnuplot para o funcionamento do programa;
 - O código possui comentários com explicações dos algoritmos a serem executados;

- Adicionar nos diagramas de pacotes as bibliotecas e subsistemas selecionados no projeto do sistema (exemplo: a biblioteca gráfica selecionada).
 - Foi utilizada como biblioteca gráfica CGnuplot, para tanto foi necessária a instalação do software GNUPLOT para plotagem dos gráficos gerados;
- Novas classes e associações oriundas das bibliotecas selecionadas e da linguagem escolhida devem ser acrescentadas ao modelo.
 - Neste projeto foi feita uma associação entre a biblioteca CGnuplot com as classes CRazaoSolubilidade, CCompresOleo, CFatorVolumeFormacaoOleo, CFatorZ, CFatorVolumeFormacaoGas para a geração dos gráficos.

Efeitos do projeto no modelo dinâmico

- Revisar os diagramas de sequência e de comunicação considerando a plataforma escolhida.
 - Não foi realizada nessa etapa do projeto uma vez que os diagramas de sequência e de comunicação serão modificados durante o desenvolvimento do código.
- Verificar a necessidade de se revisar, ampliar e adicionar novos diagramas de máquinas de estado e de atividades.
 - Não foi realizada nessa etapa do projeto uma vez que os diagramas de máquinas de estado e de atividades serão modificados durante o desenvolvimento do código caso seja necessário.

Efeitos do projeto nos atributos

- Atributos novos podem ser adicionados a uma classe, como, por exemplo, atributos específicos de uma determinada linguagem de softwareção (acesso a disco, ponteiros, constantes e informações correlacionadas).
 - Como alguns atributos estavam presentes na maioria dos cálculos das correlações de Vasquez e Beggs, por conta disso eles foram retirados das classes de cálculo e implementados externamente na classe C3Parametros.
 - Os atributos in e fin foram criados a fim de possibilitar a leitura dos dados necessários aos cálculos nas classes de correlações a partir da classe CFluido-BlackOil.

Efeitos do projeto nos métodos

- Em função da plataforma escolhida, verifique as possíveis alterações nos métodos. O projeto do sistema costuma afetar os métodos de acesso aos diversos dispositivos (exemplo: hd, rede).
 - Em virtude de usar leitura de disco, um método de inserção de dados através do teclado foi adicionado a fim de que o usuário possa digitar os dados desejados e não ficar preso somente a leitura do disco.
- Revise os diagramas de classes, de seqüência e de máquina de estado.
 - O diagrama de classes foi reformulado uma vez que houve subdivisões de classes já existentes e criação de novas classes.

Efeitos do projeto nas heranças

- Reorganização das classes e dos métodos (criar métodos genéricos com parâmetros que nem sempre são necessários e englobam métodos existentes).
 - Inicialmente alguns métodos haviam sido pensados como podendo ser uma classe, mas foram distribuídos tanto na classe base quanto nas derivadas para diminuição do número total de classes.
- Revise as heranças no diagrama de classes.
 - Algumas heranças puderam ser excluídas do diagrama, uma vez que alguns atributos necessários inicialmente puderam ser passados através da chamada das funções.

Efeitos do projeto nas associações

- Algumas heranças foram trocadas por associações e novas associações foram criadas para relacionamento com novas classes.

Efeitos do projeto nas otimizações

- A ordem de execução pode ser alterada.
 - Inicialmente pensou-se em solicitar ao usuário que informasse primeiramente qual parâmetro seria calculado, logo após a correlação que se deseja utilizar para então definir qual tipo de entrada de dados seria feita, porém para o caso de ele selecionar outros tipos de parâmetros para cálculo posterior, que necessitassem de propriedades dos fluidos ainda não informadas, novamente teria que haver entrada de dados no programa. Dessa forma, inverteu-se a ordem e

o primeiro quesito a ser preenchido pelo usuário refere-se ao carregamento dos dados, que podem ser feitos tanto pelo teclado ou por leitura do disco.

- Revise as associações nos diagramas de classes.
 - A classe CSimuladorBlackOil não precisa mais ter associação com a classe CGnuplot uma vez que as classes que realizarão as plotagens não são mais herdeiras do simulador. Elas são as que calculam e geram os gráficos a partir dos resultados.
 - A classe CFluidoBlackOil foi criada a fim de conter todas as informações necessárias referentes ao fluido em questão, possuindo métodos de entrada de dados pelo teclado e leitura do disco.
 - A classe C3Parametros foi criada a fim de reunir alguns atributos que estavam presentes na maioria dos cálculos das correlações de Vasquez e Beggs, por conta disso eles foram retirados das classes de cálculo e implementados externamente.

As dependências dos arquivos e bibliotecas podem ser descritos pelo diagrama de componentes, e as relações e dependências entre o sistema e o hardware podem ser ilustradas com o diagrama de implantação. Veja na Figura 4.1**** do diagrama de classes

5.3 Diagrama de componentes

O diagrama de componentes mostra a forma como os componentes do software se relacionam, suas dependências. Inclui itens como: componentes, subsistemas, executáveis, nós, associações, dependências, generalizações, restrições e notas. Exemplos de componentes são bibliotecas estáticas, bibliotecas dinâmicas, dlls, componentes Java, executáveis, arquivos de disco, código-fonte.

Na Figura 5.1 temos o diagrama de componentes, a geração de objetos depende dos arquivos de extensão .h e .cpp. O subsistema banco de dados representa o arquivo com os dados do reservatório. O programa executável a ser gerado depende das bibliotecas, dos arquivos .h e .cpp e dos arquivos de entrada.

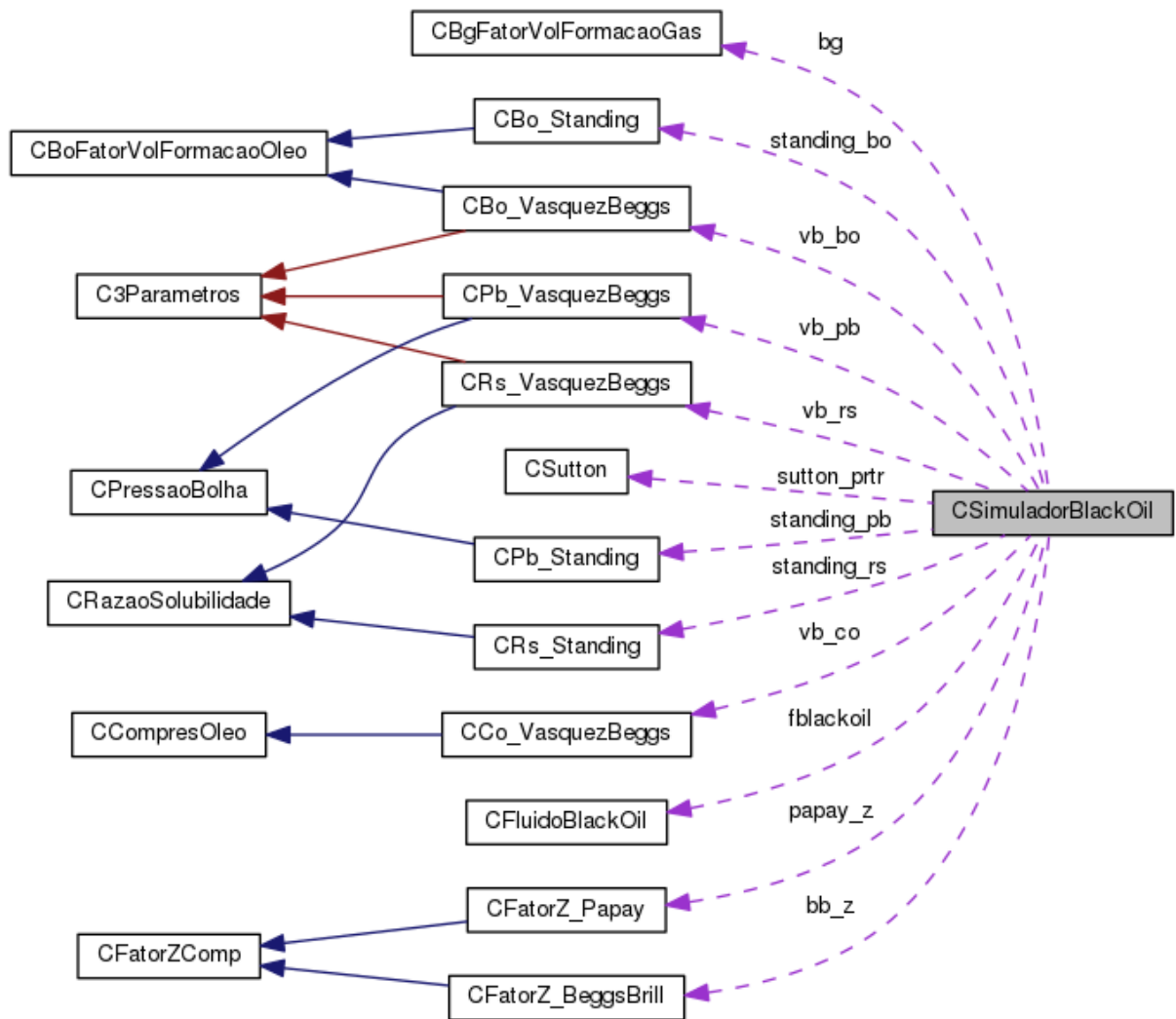


Figura 5.1: Diagrama de componentes

5.4 Diagrama de implantação

O diagrama de implantação é um diagrama de alto nível que inclui relações entre o sistema e o hardware e que se preocupa com os aspectos da arquitetura computacional escolhida. Seu enfoque é o hardware, a configuração dos nós em tempo de execução.

O diagrama de implantação deve incluir os elementos necessários para que o sistema seja colocado em funcionamento: computador, periféricos, processadores, dispositivos, nós, relacionamentos de dependência, associação, componentes, subsistemas, restrições e notas.

Veja na Figura 5.2 o diagrama de implantação. Foram obtidos dados dos parâmetros de poço através de testes de pressão, testes em amostras e testes em células pvt, os quais foram salvos em arquivos com extensão .dat no computador. O programa importa estes arquivos e utiliza o teclado e monitor para a comunicação com o usuário.

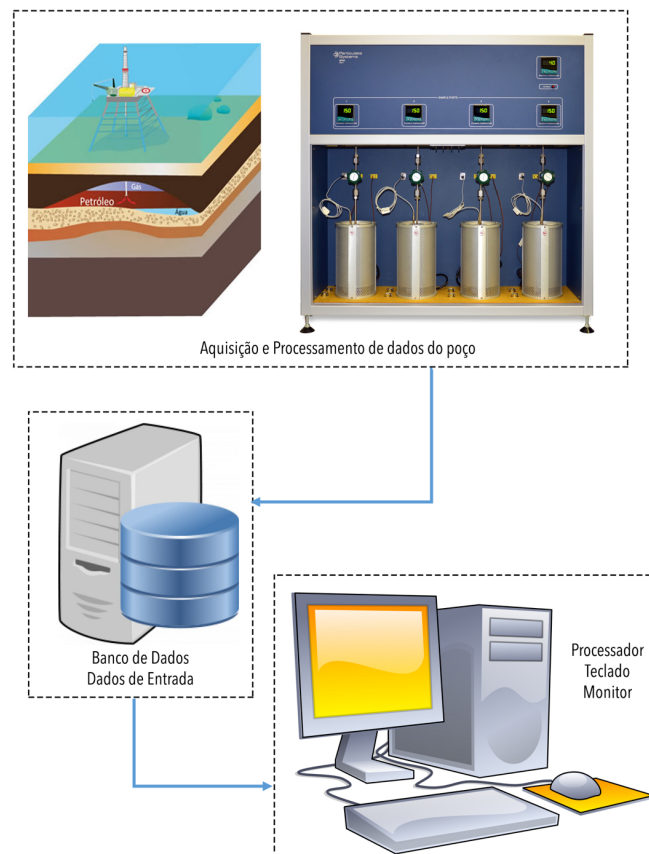


Figura 5.2: Diagrama de implantação

Capítulo 6

Implementação

Neste capítulo do projeto de engenharia apresentamos os códigos fonte que foram desenvolvidos.

Nota: os códigos devem ser documentados usando padrão **javadoc**. Posteriormente usar o programa **doxygen** para gerar a documentação no formato html.

- Veja informações gerais aqui <http://www.doxygen.org/>.
- Veja exemplo aqui <http://www.stack.nl/~dimitri/doxygen/manual/docblocks.html>.

Nota: ao longo deste capítulo usamos inclusão direta de arquivos externos usando o pacote *listings* do L^AT_EX. Maiores detalhes de como a saída pode ser gerada estão disponíveis nos links abaixo.

- http://en.wikibooks.org/wiki/LaTeX/Source_Code_Listings.
- <http://mirrors.ctan.org/macros/latex/contrib/listings/listings.pdf>.

6.1 Código fonte

Apresenta-se a seguir um conjunto de classes (arquivos .h e .cpp) além do programa `main`.

Apresenta-se na listagem 6.1 o arquivo com código da classe `CFluidorBlackOil`.

Listing 6.1: Arquivo de cabeçalho da classe `CFluidorBlackOil`.

```
1 #ifndef CFluidorBlackOil_h
2 #define CFluidorBlackOil_h
3
4 #include <iostream>
5 #include <fstream>
6 #include <vector>
7 #include <string>
8
```

```

9/**
10@brief Classe que representa as caracteristicas do fluido blackoil
11@class CFluidoBlackOil
12@file CFluidoBlackOil.h
13 */
14class CFluidoBlackOil {
15
16///Atributos
17public:
18    /// Grau API do oleo
19    double api;
20    /// Temperatura do fluido
21    double temp;
22    /// Razao de solubilidade do gas no oleo no ponto de bolha
23    double rsb;
24    /// Pressao no ponto desejado
25    double p;
26    /// Pressao maxima do reservatorio
27    double pmax;
28    /// Variacao de pressao desejada
29    double deltap;
30    /// Densidade do gas no separador
31    double dgs;
32    /// Densidade do gas
33    double dg;
34    /// Densidade do oleo
35    double d_o;
36    /// Vetor com as pressoes
37    std::vector<double> vpressao;
38
39///Metodos
40public:
41    ///Construtor default
42    CFluidoBlackOil(){};
43    ///Destrutor
44    ~CFluidoBlackOil (){};
45
46    /// Metodo para criar o vetor de pressoes
47    std::vector<double> Gera_vp (double p, double pmax, double deltap);
48
49    /// Metodos set e get
50    double P(){    return p;    }
51    void P(double _p){    p=_p;    }
52
53    double PMax(){    return pmax;    }
54    void PMax(double _pm){    pmax=_pm;    }
55
56    double API(){    return api;    }

```

```

57     void API(double _API){          api=_API;      }
58
59     double Temp(){          return temp;      }
60     void Temp(double _Temp){      temp=_Temp;      }
61
62     double Rsb(){          return rsb;      }
63     void Rsb(double _Rsb){      rsb=_Rsb;      }
64
65     double Dg(){          return dg;      }
66     void Dg(double _Dg){      dg=_Dg;      }
67
68     double Dgs(){          return dgs;      }
69     void Dgs(double _Dgs){      dgs=_Dgs;      }
70
71     double D_o(){          return d_o;      }
72     void D_o(double _d_o){      d_o=_d_o;      }
73
74     /// Sobrecarga de operadores >>, entrada de dados pelo disco e
       teclado
75     friend std::istream& operator >> (std::istream& in, CFluidoBlackOil&
       obj);
76     friend std::ifstream& operator >> (std::ifstream& fin,
       CFluidoBlackOil& obj);
77
78 };
79 #endif // CFluidoBlackOil_h

```

Apresenta-se na listagem 6.2 o arquivo de implementação da classe CFluidoBlackOil.

Listing 6.2: Arquivo de implementação da classe CFluidoBlackOil.

```

80 #include <iostream>
81 #include <fstream>
82
83 #include "CFluidoBlackOil.h"
84
85 /**
86  @brief Classe que representa as características do fluido blackoil
87  @class CFluidoBlackOil
88  @file CFluidoBlackOil.cpp
89  */
90
91 using namespace std;
92
93 ///Sobrecarga do operador >>, entrada de dados pelo teclado
94 istream& operator >> (istream& in, CFluidoBlackOil& obj){
95
96     cout << "Entre com os valores das variaveis conforme sequencia:\n";
97     cout << "1- Grau API\n";
98     in >> obj.api; in.get();

```

```

99     cout << "\n2- Temperatura no Reservatorio (em Fahrenheit) \n";
100    in >> obj.temp; in.get();
101    cout << "\n3- Razao de Solubilidade no ponto de bolha = Rsb (em
        scf/STB) \n";
102    in >> obj.rsb; in.get();
103    cout << "\n4- Pressao do ponto desejado (em Psia) (caso queira
        calcular de varios pontos digitar 0 para essa variavel) \n";
104    in >> obj.p; in.get();
105    cout << "\n5- Pressao maxima do reservatorio (caso queira calcular
        em um unico ponto digitar 0 para essa variavel) \n";
106    in >> obj.pmax; in.get();
107    cout << "\n6- Variacao de pressao desejada para acrescimo de
        pressao \n";
108    in >> obj.deltap; in.get();
109    cout << "\n7- Densidade do gas no separador \n";
110    in >> obj.dgs; in.get();
111    cout << "\n8- Densidade do gas \n";
112    in >> obj.dg; in.get();
113    cout << "\n9- Densidade do oleo \n";
114    in >> obj.d_o; in.get();
115
116    return in;
117}
118///Sobrecarga do operador >>, entrada de dados pelo disco
119ifstream& operator >> (ifstream& fin, CFluidoBlackOil& obj){
120
121    fin.ignore(500, '\n');
122    fin>>obj.p;
123    fin.ignore(500, '\n');
124    fin>>obj.pmax;
125    fin.ignore(500, '\n');
126    fin>>obj.deltap;
127    fin.ignore(500, '\n');
128    fin>>obj.api;
129    fin.ignore(500, '\n');
130    fin>>obj.dg;
131    fin.ignore(500, '\n');
132    fin>>obj.temp;
133    fin.ignore(500, '\n');
134    fin>> obj.dgs;
135    fin.ignore (500, '\n');
136    fin>>obj.rsb;
137    fin.ignore(500, '\n');
138    fin>>obj.d_o;
139    fin.close();
140    return fin;
141}
142/// Metodo para criar o vetor de pressoes

```

```

143 vector<double> CFluidoBlackOil :: Gera_vp(double p, double pmax, double
    deltap) {
144
145 /// Caso o calculo seja para um vetor de pressoes
146 if (p==0.0) {
147     for(double i =14.7; i<= pmax; i += deltap) {
148         vpressao.push_back(i);
149     }
150 }
151 /// Caso o calculo seja para um unico de valor de pressao
152 if(pmax==0.0 && deltap ==0.0){
153     vpressao.push_back(p);
154 }
155 /// Teste e instrucoes para entrada de dados
156 if (((p!=0.0) && (pmax!=0.0) && (deltap!=0.0)) || ((p==0.0) && (pmax
    ==0.0) && (deltap==0.0)))
157 {
158     cout << "\nDados inseridos incorretamente!\nFavor digitar 0 e
        verificar regras para calculo em um ponto ou um vetor.\n";
159     cout << "- Caso queira calcular os parametros em um unico valor
        de pressao, \ninserir 0 em pressao maxima e 0 em variacao de
        pressao.\n";
160     cout << "- Caso queira calcular os parametros em diversos
        valores de pressao inserir 0 em Pressao do ponto desejado.\n
        " << endl;
161 }
162 return vpressao;
163 }

```

Apresenta-se na listagem 6.3 o arquivo com código da classe C3Parametros.

Listing 6.3: Arquivo de cabeçalho da classe C3Parametros.

```

164 #ifndef C3Parametros_h
165 #define C3Parametros_h
166
167 /**
168 @brief Classe responsavel pelos parametros da correlacao Vasquez e Beggs
169 @class C3Parametros
170 @file C3Parametros.h
171 */
172 class C3Parametros{
173 ///Atributos
174 protected:
175     ///Parametros
176     double c1, c2, c3;
177
178 /// Metodos set e get
179 public:
180     void C1(double _c1){

```

```

181         c1=_c1;
182     };
183     double C1(){
184         return c1;
185     };
186     void C2(double _c2){
187         c2=_c2;
188     };
189     double C2(){
190         return c2;
191     };
192     void C3(double _c3){
193         c3=_c3;
194     };
195     double C3(){
196         return c3;
197     };
198
199 };
200 #endif

```

Apresenta-se na listagem 6.4 o arquivo de implementação da classe C3Parametros.

Listing 6.4: Arquivo de implementação da classe C3Parametros.

```

202 #include "C3Parametros.h"

```

Apresenta-se na listagem 6.5 o arquivo com código da classe CPressaoBolha.

Listing 6.5: Arquivo de cabeçalho da classe CPressaoBolha.

```

204 #ifndef CPressaoBolha_h
205 #define CPressaoBolha_h
206
207 #include "CFluidoBlackOil.h"
208
209 /**
210  @brief Classe responsavel pelo calculo da Pressao de bolha
211  @class CPressaoBolha
212  @file CPressaoBolha.h
213  */
214 class CPressaoBolha {
215
216     ///Atributos
217 public:
218     /// Pressao de bolha
219     double pb;
220
221     ///Metodos
222 public:
223     ///Construtor default

```

```

224     CPressaoBolha  () {};
225     ///Destrutor
226     ~CPressaoBolha () {};
227
228     /// Metodos get e set
229     void Pb(double _pb) {pb = _pb;}
230     double Pb()          {return pb;}
231
232     /// Metodo para calculo da pressão de bolha
233     double Pb(CFluidoBlackOil&) { return pb; }
234
235 };
236
237 #endif // CPressaoBolha_h

```

Apresenta-se na listagem 6.6 o arquivo de implementação da classe CPressaoBolha.

Listing 6.6: Arquivo de implementação da classe CPressaoBolha.

```

238 #include "CPressaoBolha.h"

```

Apresenta-se na listagem 6.7 o arquivo com código da classe CPb_Standing.

Listing 6.7: Arquivo de cabeçalho da classe CPb_{standing}.

```

239 #ifndef CPb_Standing_h
240 #define CPb_Standing_h
241
242 #include "CPressaoBolha.h"
243 #include "CFluidoBlackOil.h"
244
245 /**
246  @brief Classe responsavel pelo calculo da Pressao de bolha pela
247         correlacao Standing
248  @class CPb_Standing
249  @file CPb_Standing.h
250  */
251 class CPb_Standing : public CPressaoBolha ///Herdeira da classe
252         CPressaoBolha
253 {
254     ///Metodos
255     public:
256         ///Construtor default
257         CPb_Standing () {};
258         ///Destrutor
259         ~CPb_Standing () {};
260
261         /// Metodo para calculo da pressao de bolha usando a correlacao
262         Standing
263         double Pb(CFluidoBlackOil& fluido);

```

```

262
263};
264
265#endif // CPb_Standing_h

```

Apresenta-se na listagem 6.8 o arquivo de implementação da classe CPb_Standing.

Listing 6.8: Arquivo de implementação da classe CPb_{standing}.

```

266#include <cmath>
267
268#include "CPb_Standing.h"
269
270/**
271 @class CPb_Standing
272 @file CPb_Standing.cpp
273 */
274using namespace std;
275
276/// Metodo para calculo da pressao de bolha usando a correlação Standing
277double CPb_Standing::Pb(CFluidoBlackOil& fluido){
278
279pb =18.2*((1.0/(pow(10.0,(0.0125*fluido.api-0.00091*fluido.temp)))*(pow(
    fluido.rsb/fluido.dg,0.83)))-1.4);
280
281return pb;
282
283}

```

Apresenta-se na listagem 6.9 o arquivo com código da classe CPb_VasquezBeggs.

Listing 6.9: Arquivo de cabeçalho da classe CPb_{VasquezBeggs}.

```

285#ifndef CPb_VasquezBeggs_h
286#define CPb_VasquezBeggs_h
287
288#include "CPressaoBolha.h"
289#include "C3Parametros.h"
290
291/**
292 @brief Classe responsavel pelo calculo da Pressao de bolha pela
    correlacao Vasquez e Beggs
293 @class CPb_VasquezBeggs
294 @file CPb_VasquezBeggs.h
295 */
296class CPb_VasquezBeggs: public CPressaoBolha , C3Parametros ///Herdeira
    das classes CPressaBolha e C3Parametros
297{
298 ///Metodos
299public:
300    ///Construtor default

```



```

301     CPb_VasquezBeggs() {} ;
302     ///Destrutor
303     ~CPb_VasquezBeggs () {} ;
304
305     ///Metodo para calculo da pressao de bolha usando a correlacao
306         Vasquez e Beggs
307     double Pb(CFluidoBlackOil& fluido);
308 };
309
310 #endif // CPb_VasquezBeggs_h

```

Apresenta-se na listagem 6.10 o arquivo de implementação da classe CPb_VasquezBeggs.

Listing 6.10: Arquivo de implementação da classe CPb_VasquezBeggs.

```

311 #include <cmath>
312
313 #include "CPb_VasquezBeggs.h"
314 #include "CSimuladorBlackOil.h"
315
316 /**
317  @class CPb_VasquezBeggs
318  @file CPb_VasquezBeggs.cpp
319  */
320 using namespace std;
321
322 /// Metodo para calculo da pressão de bolha usando a correlação Vasquez
323 e Beggs
324 double CPb_VasquezBeggs::Pb(CFluidoBlackOil& fluido){
325     /// Seta os parametros da correlacao C3Parametros
326     if (fluido.api <= 30.0){
327         c1 = 0.0362;
328         c2 = 1.0937;
329         c3 = 25.724;
330     }
331     else {
332         c1 = 0.0178;
333         c2 = 1.187;
334         c3 = 23.931;
335     }
336     pb=pow((fluido.rsb/(c1*fluido.dgs* exp((c3*fluido.api)/(fluido.temp
337         +459.67))))),(1.0/c2));
338
339     return pb;
340 }

```

Apresenta-se na listagem 6.11 o arquivo com código da classe CRazaoSolubilidade.

Listing 6.11: Arquivo de cabeçalho da classe CRazaoSolubilidade.

```

340 #ifndef CRazaoSolubilidade_h
341 #define CRazaoSolubilidade_h
342
343 #include <vector>
344
345 #include "CFluidoBlackOil.h"
346 #include "CPressaoBolha.h"
347
348 /**
349  @brief Classe responsavel pelo calculo da Razao de Solubilidade
350  @class CRazaoSolubilidade
351  @file CRazaoSolubilidade.h
352  */
353 class CRazaoSolubilidade {
354
355     ///Atributo
356 public:
357     /// Vetor de Razao de Solubilidade
358     std::vector <double> vrs;
359
360     ///Metodos
361 public:
362     ///Construtor default
363     CRazaoSolubilidade () {};
364     ///Destrutor
365     ~CRazaoSolubilidade () {};
366
367     /// Metodos get e set
368     void Vrs(std::vector<double> _vrs) {vrs = _vrs;}
369     std::vector<double> Vrs()          {return vrs;}
370
371     /// Metodo para calculo da razao de solubilidade
372     std::vector<double> Rs(CFluidoBlackOil&, CPressaoBolha&) {return vrs
        ; }
373
374     /// Metodo para gerar grafico
375     void Plot(CFluidoBlackOil& fluido);
376
377 };
378 #endif // CRazaoSolubilidade_h

```

Apresenta-se na listagem 6.12 o arquivo de implementação da classe CRazaoSolubilidade.

Listing 6.12: Arquivo de implementação da classe CRazaoSolubilidade.

```

379 #include "CRazaoSolubilidade.h"
380 #include "CGnuplot.h"

```

```

381
382 /**
383 @class CRazaoSolubilidade
384 @file CRazaoSolubilidade.cpp
385 */
386
387 /// Metodo para gerar grafico
388 void CRazaoSolubilidade:: Plot(CFluidoBlackOil& fluido) {
389
390 static CGnuplot g2d;
391 g2d.Title("Pressao_x_Razao_de_Solubilidade");
392 g2d.XLabel("Pressao(psia)");
393 g2d.YLabel("Rs(scf/STB)");
394 g2d.Style("linespoints");
395 g2d.PlotVector(fluido.vpressao,vrs);
396 }

```

Apresenta-se na listagem 6.13 o arquivo com código da classe CRs_Standing.

Listing 6.13: Arquivo de cabeçalho da classe CRs_{standing}.

```

397 #ifndef CRs_Standing_h
398 #define CRs_Standing_h
399
400 #include "CRazaoSolubilidade.h"
401
402 /**
403 @brief Classe responsavel pelo calculo da Razao de Solubilidade pela
         correlacao Standing
404 @class CRs_Standing
405 @file CRs_Standing.h
406 */
407 class CRs_Standing : public CRazaoSolubilidade ///Herdeira da classe
         CRazaoSolubilidade
408 {
409
410 ///Metodos
411 public:
412     ///Construtor default
413     CRs_Standing () {};
414     ///Destrutor
415     ~CRs_Standing () {};
416
417     /// Metodo para calculo da razao de solubilidade pela correlacao de
         Standing
418     std::vector<double> Rs(CFluidoBlackOil& fluido, CPressaoBolha& cpb);
419 };
420
421 #endif // CRs_Standing_h

```

Apresenta-se na listagem 6.14 o arquivo de implementação da classe `CRs_Standing`.

Listing 6.14: Arquivo de implementação da classe `CRsstanding`.

```

422#include <vector>
423#include <cmath>
424
425#include "CRs_Standing.h"
426
427/**
428@class CRs_Standing
429@file CRs_Standing.cpp
430 */
431using namespace std;
432
433/// Metodo para calculo da razao de solubilidade pela correlacao de
    Standing
434vector<double> CRs_Standing::Rs(CFluidoBlackOil& fluido, CPressaoBolha&
    cpb){
435    vrs.resize(fluido.vpressao.size()); //mesmo tamanho
436
437/// Para pressoes abaixo do Pb a razao de solubilidade e calculada, para
    pressoes acima do Pb o Rs se mantem constante e igual ao Rsb
438for (int i=0; i<fluido.vpressao.size(); i++)
439    {
440        if (fluido.vpressao[i] <= cpb.pb){
441            vrs[i] = fluido.dg*(pow(((fluido.vpressao[i]/18.2+1.4)*(pow
                (10.0, (0.0125*fluido.api-0.00091*fluido.temp))))),
                1.2048));
442        }
443        else
444            vrs[i] = fluido.rsb;
445    }
446    return vrs;
447}

```

Apresenta-se na listagem 6.15 o arquivo com código da classe `CRs_VasquezBeggs`.

Listing 6.15: Arquivo de cabeçalho da classe `CRsVasquezBeggs`.

```

448#ifndef CRs_VasquezBeggs_h
449#define CRs_VasquezBeggs_h
450
451#include "CRazaoSolubilidade.h"
452#include "C3Parametros.h"
453
454/**
455@brief Classe responsavel pelo calculo da Razao de solubilidade pela
    correlacao Vasquez e Beggs
456@class CRs_VasquezBeggs
457@file CRs_VasquezBeggs.h

```

```

458 */
459 class CRs_VasquezBeggs: public CRazaoSolubilidade, C3Parametros ///
    Herdeira das classes CRazaoSolubilidade e C3Parametros
460 {
461
462 ///Metodos
463 public:
464     ///Construtor default
465     CRs_VasquezBeggs () {};
466     ///Destrutor
467     ~CRs_VasquezBeggs () {};
468
469     /// Metodo para calculo da razao de solubilidade pela correlacao de
        Vasquez e beggs
470     std::vector<double> Rs(CFluidoBlackOil& fluido, CPressaoBolha& cpb);
471
472 };
473
474 #endif // CRs_VasquezBeggs_h

```

Apresenta-se na listagem 6.16 o arquivo de implementação da classe CRs_VasquezBeggs.

Listing 6.16: Arquivo de implementação da classe CRs_VasquezBeggs.

```

475 #include <vector>
476 #include <cmath>
477
478 #include "CRs_VasquezBeggs.h"
479
480 /**
481 @class CRs_VasquezBeggs
482 @file CRs_VasquezBeggs.cpp
483 */
484 using namespace std;
485
486 /// Metodo para calculo da pressão de bolha usando a correlação Vasquez
    e Beggs
487 vector<double> CRs_VasquezBeggs::Rs(CFluidoBlackOil& fluido,
    CPressaoBolha& cpb){
488     /// Seta os parametros da correlacao C3Parametros
489     if (fluido.api <= 30.0){
490         c1 = 0.0362;
491         c2 = 1.0937;
492         c3 = 25.724;
493     }
494     else {
495         c1 = 0.0178;
496         c2 = 1.187;
497         c3 = 23.931;
498     }

```

```

499
500     vrs.resize(fluido.vpressao.size()); //mesmo tamanho
501
502 for (int i=0; i<fluido.vpressao.size(); i++){
503     /// Para pressoes abaixo do Pb a razao de solubilidade e
504     calculada, para pressoes acima do Pb o Rs se mantem constante
505     e igual ao Rsb
506     if (fluido.vpressao[i] <= cpb.pb){
507         vrs[i] = c1*fluido.dgs*pow(fluido.vpressao[i], c2)*exp((c3*
508             fluido.api)/(fluido.temp+459.67));
509     }
510     else
511         vrs[i] = fluido.rsb;
512 }
513 return vrs;
514 }

```

Apresenta-se na listagem 6.17 o arquivo com código da classe CBoFatorVolFormacaoOleo.

Listing 6.17: Arquivo de cabeçalho da classe CBoFatorVolFormacaoOleo.

```

513 #ifndef CBoFatorVolFormacaoOleo_h
514 #define CBoFatorVolFormacaoOleo_h
515
516 #include <vector>
517
518 #include "CFluidoBlackOil.h"
519 #include "CRazaoSolubilidade.h"
520 #include "CCompresOleo.h"
521 #include "CPressaoBolha.h"
522
523 /**
524  * @brief Classe responsavel pelo calculo do Fator volume de formacao do
525  *        oleo
526  * @class CBoFatorVolFormacaoOleo
527  * @file CBoFatorVolFormacaoOleo.h
528  */
529
530 class CBoFatorVolFormacaoOleo{
531
532 public:
533     /// Fator volume de formacao no ponto de bolha
534     double bob;
535     /// Vetor de Fator volume de formacao em outros pontos
536     std::vector<double> vbo;
537
538 /// Metodos

```

```

538 public:
539     ///Construtor default
540     CBoFatorVolFormacaoOleo (){};
541     ///Destrutor
542     ~CBoFatorVolFormacaoOleo (){};
543
544     /// Metodos get e set
545     void Bob(double _bob) {bob = _bob;}
546     double Bob()          {return bob;}
547
548     void Vbo(std::vector<double> _vbo) {vbo = _vbo;}
549     std::vector<double> Vbo()          {return vbo;}
550
551
552     /// Metodo para calculo do Bo no ponto de bolha
553     double Bob(CFluidoBlackOil&) { return bob; }
554
555     /// Metodo para calculo do Bo em varios pontos
556     std::vector<double> Bo(CFluidoBlackOil&, CPressaoBolha&,
557         CRazaoSolubilidade&, CCompresOleo&) { return vbo; }
558
559     /// Metodo para gerar grafico
560     void Plot(CFluidoBlackOil& fluido);
561
562 #endif // CBoFatorVolFormacaoOleo_h

```

Apresenta-se na listagem 6.18 o arquivo de implementação da classe CBoFatorVolFormacaoOleo.

Listing 6.18: Arquivo de implementação da classe CBoFatorVolFormacaoOleo.

```

563 #include "CBoFatorVolFormacaoOleo.h"
564 #include "CGnuplot.h"
565
566 /**
567 @class CBoFatorVolFormacaoOleo
568 @file CBoFatorVolFormacaoOleo.cpp
569 */
570
571 /// Metodo para gerar grafico
572 void CBoFatorVolFormacaoOleo :: Plot (CFluidoBlackOil& fluido) {
573
574     static CGnuplot g2d;
575     g2d.Title("Pressao_x_Fator_Volume_de_Formacao_do_Oleo");
576     g2d.XLabel("Pressao (psia)");
577     g2d.YLabel("Bo (bbl/STB)");
578     g2d.Style("linespoints");
579     g2d.PlotVector(fluido.vpressao, vbo);

```

```
580
581 }
```

Apresenta-se na listagem 6.19 o arquivo com código da classe CBo_Standing.

Listing 6.19: Arquivo de cabeçalho da classe CBo_{standing}.

```
582 #ifndef CBo_Standing_h
583 #define CBo_Standing_h
584
585 #include "CBoFatorVolFormacaoOleo.h"
586
587 /**
588  @brief Classe responsavel pelo calculo do Fator volume de formacao do
         oleo pela correlacao Standing
589  @class CBo_Standing
590  @file CBo_Standing.h
591  */
592 class CBo_Standing : public CBoFatorVolFormacaoOleo    ///Herdeira da
         classe CFatorVolumeFormacaoOleo
593 {
594     ///Metodos
595 public:
596     ///Construtor default
597     CBo_Standing (){};
598     ///Destructor
599     ~CBo_Standing(){};
600
601     /// Metodo para calculo do Bo no ponto de bolha pela correlacao
         Standing
602     double Bob(CFluidoBlackOil& fluido);
603 };
604
605 #endif // CBo_Standing_h
```

Apresenta-se na listagem 6.20 o arquivo de implementação da classe CBo_Standing.

Listing 6.20: Arquivo de implementação da classe CBo_{standing}.

```
606 #include <cmath>
607
608 #include "CBo_Standing.h"
609
610 /**
611  @class CBo_Standing
612  @file CBo_Standing.cpp
613  */
614 using namespace std;
615
616 /// Metodo para calculo do Bo no ponto de bolha pela correlacao Standing
617 double CBo_Standing::Bob(CFluidoBlackOil& fluido){
```



```

618
619     bob= 0.972 + 1.47*pow(10.0, -4.0)*pow((fluido.rs*(pow((fluido.dg/
        fluido.d_o), 0.5))+1.25*fluido.temp), 1.175);
620
621     return bob;
622 }

```

Apresenta-se na listagem 6.21 o arquivo com código da classe CBo_VasquezBeggs.

Listing 6.21: Arquivo de cabeçalho da classe CBo_VasquezBeggs.

```

624 #ifndef CBo_VasquezBeggs_h
625 #define CBo_VasquezBeggs_h
626
627 #include <vector>
628
629 #include "CBoFatorVolFormacaoOleo.h"
630 #include "C3Parametros.h"
631
632 /**
633  @brief Classe responsavel pelo calculo do Fator volume de formacao do
        oleo pela correlacao Vasquez e Beggs
634  @class CRs_VazquezBeggs
635  @file CRs_VasquezBeggs.h
636  */
637 class CBo_VasquezBeggs: public CBoFatorVolFormacaoOleo, C3Parametros ///
        Herdeira das classes CFatorVolumeFormacaoOleo e C3Parametros
638 {
639     ///Metodos
640 public:
641     ///Construtor default
642     CBo_VasquezBeggs () {} ;
643     ///Destrutor
644     ~CBo_VasquezBeggs () {} ;
645
646     /// Metodo para calculo do Bo pela correlacao Vasquez e Beggs
647     std::vector<double> Bo(CFluidoBlackOil& fluido, CPressaoBolha& cpb,
        CRazaoSolubilidade& crs, CCompresOleo& cco);
648
649 };
650 #endif // CBo_VasquezBeggs_h

```

Apresenta-se na listagem 6.22 o arquivo de implementação da classe CBo_VasquezBeggs.

Listing 6.22: Arquivo de implementação da classe CBo_VasquezBeggs.

```

651 #include <cmath>
652
653 #include "CBo_VasquezBeggs.h"
654
655 /**

```

```

656 @class CBo_VasquezBeggs
657 @file CBo_VasquezBeggs.cpp
658 */
659 using namespace std;
660
661 /// Metodo para calculo do Bo pela correlacao Vasquez e Beggs
662 vector<double> CBo_VasquezBeggs::Bo(CFluidoBlackOil& fluido,
    CPressaoBolha& cpb, CRazaoSolubilidade& crs, CCompresOleo& cco){
663     /// Seta os parametros da correlacao C3Parametros
664     if (fluido.api <= 30.0){
665         c1 = 0.0004677;
666         c2 = 0.00001751;
667         c3 = -0.00000001811;
668     }
669     else {
670         c1 = 0.0004677;
671         c2 = 0.000011;
672         c3 = -0.000000001377;
673     }
674
675     /// Calculo do Bo no ponto de bolha
676     bob = 1+c1*fluido.rsb+c2*(fluido.temp-60.0)*(fluido.api/fluido.
        dgs)+c3*fluido.rsb*(fluido.temp-60.0)*(fluido.api/fluido.dgs)
        ;
677
678     vbo.resize(fluido.vpressao.size()); //mesmo tamanho
679
680     /// Calculo do Bo acima do ponto de bolha e diferente do calculo
        para abaixo do ponto de bolha
681     for (int i=0; i<fluido.vpressao.size(); i++){
682
683         ///Verifica se a pressao esta acima ou abaixo do ponto de bolha
684         if (fluido.vpressao[i] <= cpb.pb)
685             vbo[i] = 1.0+c1*crs.vrs[i]+c2*(fluido.temp-60.0)*(fluido.api
                /fluido.dgs)+c3*crs.vrs[i]*(fluido.temp-60.0)*(fluido.api
                /fluido.dgs);
686
687         else
688             vbo[i] = bob*exp(cco.vco[i]*(cpb.pb - fluido.vpressao[i]));
689     }
690
691     return vbo;
692 }

```

Apresenta-se na listagem 6.23 o arquivo com código da classe CCompresOleo.

Listing 6.23: Arquivo de cabeçalho da classe CCompresOleo.

```

693 #ifndef CCompresOleo_h
694 #define CCompresOleo_h

```

```

695
696#include <iostream>
697#include <vector>
698
699#include "CFluidoBlackOil.h"
700#include "CRazaoSolubilidade.h"
701
702/**
703@brief Classe responsavel pelo calculo da Compressibilidade do Oleo
704@class CCompresOleo
705@file CCompresoleo.h
706 */
707class CCompresOleo {
708
709///Atributos
710public:
711    /// Vetor de compressibilidade do oleo
712    std::vector<double> vco;
713
714///Metodos
715public:
716    ///Construtor default
717    CCompresOleo (){};
718    ///Destrutor
719    ~CCompresOleo (){};
720
721    /// Metodos get e set
722    void Vco(std::vector<double> _vco) {vco = _vco;}
723    std::vector<double> Vco() {return vco;}
724
725    /// Metodo para calculo da compressibilidade do oleo
726    std::vector<double> Co(CFluidoBlackOil&, CRazaoSolubilidade&) {
        return vco; }
727
728    /// Metodo para gerar grafico
729    void Plot(CFluidoBlackOil& fluido);
730
731};
732#endif // CCompres_Oleo_h

```

Apresenta-se na listagem 6.24 o arquivo de implementação da classe CCompresOleo.

Listing 6.24: Arquivo de implementação da classe CCompresOleo.

```

733#include "CCompresOleo.h"
734#include "CGnuplot.h"
735
736/**
737@class CCompresOleo
738@file CCompresOleo.cpp

```

```

739 */
740
741 /// Metodo para gerar grafico
742 void CCompresOleo:: Plot(CFluidoBlackOil& fluido) {
743
744 static CGnuplot g2d;
745 g2d.Title("Pressao_x_Compressibilidade_do_Oleo");
746 g2d.XLabel("Pressao(psia)");
747 g2d.YLabel("Co(1/psi)");
748 g2d.Style("linespoints");
749 g2d.PlotVector(fluido.vpressao,vco);
750 }

```

Apresenta-se na listagem 6.25 o arquivo com código da classe CCo_VasquezBeggs.

Listing 6.25: Arquivo de cabeçalho da classe CCo_VasquezBeggs.

```

751 #ifndef CCo_VasquezBeggs_h
752 #define CCo_VasquezBeggs_h
753
754 #include <iostream>
755 #include <vector>
756
757 #include "CCompresOleo.h"
758
759 /**
760 @brief Classe responsavel pelo calculo da compressibilidade do oleo pela
       correlacao Vasquez e Beggs
761 @class CCo_VazquezBeggs
762 @file CCo_VasquezBeggs.h
763 */
764 class CCo_VasquezBeggs: public CCompresOleo {           ///Herdeira da
       classe CCompresOleo
765
766 ///Metodos
767 public:
768     ///Construtor default
769     CCo_VasquezBeggs () {};
770     ///Destrutor
771     ~CCo_VasquezBeggs () {};
772
773 /// Metodo para calculo do Co pela correlacao Vasquez e Beggs
774 std::vector<double> Co (CFluidoBlackOil& fluido,CRazaoSolubilidade& crs
       );
775
776 };
777 #endif // CCo_VasquezBeggs_h

```

Apresenta-se na listagem 6.26 o arquivo de implementação da classe CCo_VasquezBeggs.

Listing 6.26: Arquivo de implementação da classe CCo_{VasquezBeggs}.

```

778#include "CCo_VasquezBeggs.h"
779
780/**
781@class CBo_VasquezBeggs
782@file CBo_VasquezBeggs.cpp
783 */
784using namespace std;
785
786/// Metodo para calculo do Co pela correlacao Vasquez e Beggs
787vector<double> CCo_VasquezBeggs:: Co (CFluidoBlackOil& fluido,
      CRazaoSolubilidade& crs) {
788    vco.resize(fluido.vpressao.size()); //mesmo tamanho
789
790for (int i=0; i<fluido.vpressao.size(); i++){
791    vco[i]= ( -1433.0 + 5.0*crs.vrs[i] + 17.2*fluido.temp - 1180.0*
      fluido.dgs + 12.61*fluido.api) / (fluido.vpressao[i] *
      100000.0);
792    }
793    return vco;
794}

```

Apresenta-se na listagem 6.27 o arquivo com código da classe CFatorZComp.

Listing 6.27: Arquivo de cabeçalho da classe CFatorZComp.

```

795#ifndef CFatorZComp_h
796#define CFatorZComp_h
797
798
799#include <iostream>
800#include <vector>
801
802#include "CFluidoBlackOil.h"
803#include "CSutton.h"
804
805/**
806@brief Classe responsavel pelo calculo do Fator Z de compressibilidade
      do gas
807@class CFatorZComp
808@file CFatorZComp.h
809 */
810class CFatorZComp {
811///Atributos
812public:
813    ///Vetor de Fator Z
814    std::vector<double> vz;
815
816///Metodos

```

```

817 public:
818     ///Construtor default
819     CFatorZComp () {};
820     ///Destrutor
821     ~CFatorZComp () {};
822
823     /// Metodos get e set
824     void Vz(std::vector<double> _vz) {vz = _vz;}
825     std::vector<double> Vz()          {return vz;}
826
827     ///Metodo para o calculo do Fator de compressibilidade do gas Z
828     std::vector<double> Vz(CSutton&) {return vz;}
829
830     /// Metodo para gerar grafico
831     void Plot(CFluidoBlackOil&);
832
833 };
834
835 #endif // CFatorZComp_h

```

Apresenta-se na listagem 6.28 o arquivo de implementação da classe CFatorZComp.

Listing 6.28: Arquivo de implementação da classe CFatorZComp.

```

836 #include "CFatorZComp.h"
837 #include "CGnuplot.h"
838
839 /**
840 @class CFatorZComp
841 @file CFatorZComp.cpp
842 */
843
844 /// Metodo para gerar grafico
845 void CFatorZComp:: Plot(CFluidoBlackOil& fluido){
846
847     static CGnuplot g2d;
848     g2d.Title("Pressao_x_Fator_de_Compressibilidade_do_Gas");
849     g2d.XLabel("Pressao(psia)");
850     g2d.YLabel("Z");
851     g2d.Style("linespoints");
852     g2d.PlotVector(fluido.vpressao,vz);
853 }

```

Apresenta-se na listagem 6.29 o arquivo com código da classe CFatorZ_Papay.

Listing 6.29: Arquivo de cabeçalho da classe CFatorZ_Papay.

```

854 #ifndef CFatorZ_Papay_h
855 #define CFatorZ_Papay_h
856
857 #include "CFatorZComp.h"

```

```

858
859 /**
860 @brief Classe responsavel pelo calculo do Fator Z de compressibilidade
      do gas pela correlacao Papay
861 @class CFatorZ_Papay
862 @file CFatorZ_Papay.h
863 */
864 class CFatorZ_Papay: public CFatorZComp      ///Herdeira da classe
      CFatorZComp
865 {
866 ///Metodos
867 public:
868     ///Construtor default
869     CFatorZ_Papay () {};
870     ///Destrutor
871     ~CFatorZ_Papay () {};
872
873     ///Metodo para calculo do fator Z pela correlacao Papay
874     std::vector<double> Z (CSutton& prtr);
875
876 };
877 #endif // CFatorZ_Papay_h

```

Apresenta-se na listagem 6.30 o arquivo de implementação da classe CFatorZ.Papay.

Listing 6.30: Arquivo de implementação da classe CFatorZ_{Papay}.

```

878 #include <cmath>
879
880 #include "CFatorZ_Papay.h"
881
882 /**
883 @class CFatorZ_Papay
884 @file CFatorZ_Papay.cpp
885 */
886 using namespace std;
887
888 ///Metodo para calculo do fator Z pela correlacao Papay
889 std::vector<double> CFatorZ_Papay :: Z (CSutton& prtr){
890
891     vz.resize(prtr.vpr.size()); //mesmo tamanho
892
893     for (int i=0; i < prtr.vpr.size(); i++){
894         vz[i] = 1 - (3.52 * prtr.vpr[i])/( pow(10.0, (0.9813* prtr.tr))
            ) + (0.274 * prtr.vpr[i] * prtr.vpr[i]) / (pow(10.0,(0.8157*
            prtr.tr)));
895     }
896
897     return vz;
898 }

```

Apresenta-se na listagem 6.31 o arquivo com código da classe CFatorZ_BeggsBrill.

Listing 6.31: Arquivo de cabeçalho da classe CFatorZ_{BeggsBrill}.

```

899 #ifndef CFatorZ_BeggsBrill_h
900 #define CFatorZ_BeggsBrill_h
901
902 #include <iostream>
903 #include <vector>
904
905 #include "CFatorZComp.h"
906
907 /**
908  @brief Classe responsavel pelo calculo do Fator Z de compressibilidade
909         do gas pela correlacao Beggs e Brill
910  @class CFatorZ_BeggsBrill
911  @file CFatorZ_BeggsBrill.h
912  */
913 class CFatorZ_BeggsBrill : public CFatorZComp           ///Herdeira da
914                             classe CFatorZComp
915 {
916     ///Atributos
917 private:
918     ///Parametros da formula de Beggs e Brill( nao variam com a pressao)
919     double a, c, d;
920     /// Vetor de Parametro de Beggs e Brill (varia com a pressao)
921     std::vector<double> b;
922
923     ///Metodos
924 public:
925     ///Construtor default
926     CFatorZ_BeggsBrill () {};
927     ///Destrutor
928     ~CFatorZ_BeggsBrill () {};
929
930     ///Metodo para calculo do fator Z pela correlacao Beggs e Brill
931     std::vector<double> Z (CSutton& prtr);
932 };
933 #endif // CFatorZ_BeggsBrill_h

```

Apresenta-se na listagem 6.32 o arquivo de implementação da classe CFatorZ_BeggsBrill.

Listing 6.32: Arquivo de implementação da classe CFatorZ_{BeggsBrill}.

```

934 #include <iostream>
935 #include <cmath>
936
937 #include "CFatorZ_BeggsBrill.h"
938

```



```

939 /**
940 @class CFatorZ_BeggsBrill
941 @file CFatorZ_BeggsBrill.cpp
942 */
943 using namespace std;
944
945 ///Metodo para calculo do fator Z pela correlacao Beggs e Brill
946 vector<double> CFatorZ_BeggsBrill :: Z (CSutton& prtr) {
947     /// Calculo dos parametros independentes da pressao
948     a = 1.39 * (pow((prtr.tr - 0.92), 0.5)) - 0.36 * prtr.tr - 0.1;
949     c = 0.132 - 0.32 * log10(prtr.tr);
950     d = pow(10, (0.3106 - (0.49 * prtr.tr) + (0.1824 * prtr.tr *
        prtr.tr)));
951
952     b.resize(prtr.vpr.size()); //mesmo tamanho
953     vz.resize(prtr.vpr.size()); //mesmo tamanho
954     /// Calculo do paramentro dependente da pressao e do Fator Z
955     for (int i=0; i<prtr.vpr.size(); i++){
956         b[i] = ( 0.62 - (0.23 * prtr.tr)) * prtr.vpr[i] + ((0.066/(prtr.
            tr - 0.86)) - 0.037) * prtr.vpr[i] * prtr.vpr[i] + (0.32 / (
            pow(10, (9 * (prtr.tr - 1)))) * (pow(prtr.vpr[i], 6));
957
958         vz[i] = a + ((1-a)/ exp(b[i])) + (c * pow(prtr.vpr[i], d));
959
960     }
961     return vz;
962 }

```

Apresenta-se na listagem 6.33 o arquivo com código da classe CSutton.

Listing 6.33: Arquivo de cabeçalho da classe CSutton.

```

963 #ifndef CSutton_h
964 #define CSutton_h
965
966 #include <iostream>
967 #include <vector>
968
969 #include "CFluidoBlackOil.h"
970
971 /**
972 @brief Classe responsavel pelo calculo da Pressao e Temperatura pseudo-
        reduzida do gas pela correlacao Sutton
973 @class CSutton
974 @file CSutton.h
975 */
976 class CSutton {
977     ///Atributos
978 public:
979     /// Vetor de Pressao pseudo-reduzida

```

```

980     std::vector<double> vpr;
981     /// Temperatura pseudo-reduzida
982     double tr;
983
984 ///Metodos
985 public:
986     ///Construtor default
987     CSutton () {};
988     ///Destrutor
989     ~CSutton () {};
990
991     /// Metodos get e set
992     void Vpr(std::vector<double> _vpr) {vpr = _vpr;}
993     std::vector<double> Vpr()           {return vpr;}
994
995     void Tr(double _tr)                {tr = _tr;}
996     double Tr()                        {return tr;}
997
998     ///Metodo para o calculo da pressao pseudo-reduzida pela correlacao
999     Sutton
1000     std::vector<double> Pr (CFluidoBlackOil& fluido);
1001
1002     ///Metodo para o calculo da temperatura pseudo-reduzida pela
1003     correlacao Sutton
1004     double Tr (CFluidoBlackOil& fluido);
1005 };
1006 #endif // CSutton_h

```

Apresenta-se na listagem ?? o arquivo de implementação da classe CSutton.

Listing 6.34: Arquivo de implementação da classe CSutton.

```

1006 #include "CSutton.h"
1007
1008 /**
1009 @class CSutton
1010 @file CSutton.cpp
1011 */
1012 using namespace std;
1013
1014 ///Metodo para o calculo da pressao pseudo-reduzida pela correlacao
1015 Sutton
1016 vector<double> CSutton :: Pr (CFluidoBlackOil& fluido) {
1017     vpr.resize(fluido.vpressao.size()); //mesmo tamanho
1018     for (int i=0; i<fluido.vpressao.size(); i++){
1019         vpr[i] = fluido.vpressao[i]/(671.1 + (14.0 - 34.3*fluido.dg)*
1020             fluido.dg);
1021     }

```

```

1021     return vpr;
1022 }
1023
1024 ///Metodo para o calculo da temperatura pseudo-reduzida pela correlacao
Sutton
1025 double CSutton :: Tr(CFluidoBlackOil& fluido) {
1026
1027     tr = (fluido.temp + 459.67) / (120.1 + (429.0 - 62.9*fluido.dg)*
        fluido.dg);
1028     return tr;
1029 }

```

Apresenta-se na listagem 6.35 o arquivo com código da classe CBgFatorVolFormacaoGas.

Listing 6.35: Arquivo de cabeçalho da classe CBgFatorVolFormacaoGas.

```

1030 #ifndef CBgFatorVolFormacaoGas_h
1031 #define CBgFatorVolFormacaoGas_h
1032
1033 #include <vector>
1034
1035 #include "CFluidoBlackOil.h"
1036 #include "CFatorZComp.h"
1037
1038 /**
1039  @brief Classe responsavel pelo calculo do Fator volume de formacao do
gas
1040  @class CBgFatorVolFormacaoGas
1041  @file CBgFatorVolFormacaoGas.h
1042  */
1043
1044 class CBgFatorVolFormacaoGas {
1045 ///Atributos
1046 public:
1047     /// Vetor de Fator volume de formacao do gas Bg
1048     std::vector<double> vbg;
1049
1050 ///Metodos
1051 public:
1052     ///Construtor default
1053     CBgFatorVolFormacaoGas () {};
1054     ///Destrutor
1055     ~CBgFatorVolFormacaoGas () {};
1056
1057     /// Metodos get e set
1058     void Vbg(std::vector<double> _vbg) {vbg = _vbg;}
1059     std::vector<double> Vbg() {return vbg;}
1060

```

```

1061    /// Metodo para calcular o fator volume de formacao do gas
1062    std::vector<double> Bg (CFluidoBlackOil& fluido, CFatorZComp& fcz);
1063
1064    /// Metodo para gerar grafico
1065    void Plot(CFluidoBlackOil& fluido);
1066};
1067
1068#endif // CBgFatorVolFormacaoGas_h

```

Apresenta-se na listagem 6.36 o arquivo de implementação da classe CBgFatorVolFormacaoGas.

Listing 6.36: Arquivo de implementação da classe CBgFatorVolFormacaoGas.

```

1069#include "CBgFatorVolFormacaoGas.h"
1070#include "CGnuplot.h"
1071
1072/**
1073 @class CBgFatorVolFormacaoGas
1074 @file CBgFatorVolFormacaoGas.cpp
1075 */
1076using namespace std;
1077
1078/// Metodo para calcular o fator volume de formacao do gas
1079vector<double> CBgFatorVolFormacaoGas :: Bg(CFluidoBlackOil& fluido,
      CFatorZComp& fcz){
1080    vbg.resize(fluido.vpressao.size()); //mesmo tamanho
1081
1082    for (int i=0; i<fluido.vpressao.size(); i++){
1083        vbg[i] = (0.00504 * fcz.vz[i] * (fluido.temp + 459.67)) / fluido
            .vpressao[i];
1084    }
1085
1086    return vbg;
1087}
1088/// Metodo para gerar grafico
1089void CBgFatorVolFormacaoGas:: Plot(CFluidoBlackOil& fluido){
1090
1091    static CGnuplot g2d;
1092    g2d.Title("Pressao_x_Fator_Volume_de_Formacao_do_Gas");
1093    g2d.XLabel("Pressao(psia)");
1094    g2d.YLabel("Bg(bbl/scf)");
1095    g2d.Style("linespoints");
1096    g2d.PlotVector(fluido.vpressao,vbg);
1097
1098}

```

Apresenta-se na listagem 6.37 o arquivo com código da classe CSimuladorBlackOil.

Listing 6.37: Arquivo de cabeçalho da classe CSimuladorBlackOil.

```

1099#include <iostream>
1100#include <string>
1101#include <fstream>
1102#include <iomanip>
1103
1104#include "CFluidoBlackOil.h"
1105#include "CPressaoBolha.h"
1106#include "CPb_Standing.h"
1107#include "CPb_VasquezBeggs.h"
1108#include "CRazaoSolubilidade.h"
1109#include "CRs_Standing.h"
1110#include "CRs_VasquezBeggs.h"
1111#include "CBoFatorVolFormacaoOleo.h"
1112#include "CBo_Standing.h"
1113#include "CBo_VasquezBeggs.h"
1114#include "CCompresOleo.h"
1115#include "CCo_VasquezBeggs.h"
1116#include "CFatorZComp.h"
1117#include "CSutton.h"
1118#include "CFatorZ_Papay.h"
1119#include "CFatorZ_BeggsBrill.h"
1120#include "CBgFatorVolFormacaoGas.h"
1121
1122using namespace std;
1123
1124/**
1125 @brief Classe que simula o calculo de todos as propriedades do fluido
1126 blackoil
1127 @class CSimuladorBlackOil
1128 @file CSimuladorBlackOil.h
1129 */
1129class CSimuladorBlackOil {
1130 ///Atributos
1131public:
1132 ///Objeto da classe CFluidoBlackOil
1133 CFluidoBlackOil fblackoil;
1134 ///Objeto da classe CPb_Standing
1135 CPb_Standing standing_pb;
1136 ///Objeto da classe CPb_VasquezBeggs
1137 CPb_VasquezBeggs vb_pb;
1138 ///Objeto da classe CRs_Standing
1139 CRs_Standing standing_rs;
1140 ///Objeto da classe CRs_VasquezBeggs
1141 CRs_VasquezBeggs vb_rs;
1142 ///Objeto da classe CBo_Standing
1143 CBo_Standing standing_bo;
1144 ///Objeto da classe CBo_VasquezBeggs

```

```

1145     CBo_VasquezBeggs vb_bo;
1146     ///Objeto da classe CCo_VasquezBeggs
1147     CCo_VasquezBeggs vb_co;
1148     ///Objeto da classe CSutton
1149     CSutton sutton_prtr;
1150     ///Objeto da classe CFatorZ_Papay
1151     CFatorZ_Papay papay_z;
1152     ///Objeto da classe CFatorZ_BeggsBrill
1153     CFatorZ_BeggsBrill bb_z;
1154     ///Objeto da classe CFatorVolumeFormacaoGas
1155     CBgFatorVolFormacaoGas bg;
1156
1157 ///Metodos
1158 public:
1159     ///Construtor default
1160     CSimuladorBlackOil () {};
1161     ///Destrutor
1162     ~CSimuladorBlackOil() {};
1163
1164 ///Menu de execucao
1165 int Executar();
1166
1167 /// Metodo que solicita a entrada de dados pelo teclado ou pelo disco
1168 void EntradadeDados();
1169
1170 ///Metodo que calcula a pressao de bolha
1171 void CalculoPressaoBolha();
1172
1173 ///Metodo que calcula a razao de solubilidade
1174 void CalculoRazaodeSolubilidade();
1175
1176 ///Metodo que calcula o fator volume de formacao do oleo
1177 void CalculoFatorVolumeFormacaoOleo();
1178
1179 ///Metodo que calcula a compressibilidade do oleo
1180 void CalculoCompressibilidadeOleo();
1181
1182 ///Metodo que calcula o fator de compressibilidade do gas Z
1183 void CalculoFatordeCompressibilidadeZ();
1184
1185 ///Metodo que calcula o fator volume de formacao do gas
1186 void CalculoFatorVolumeFormacaoGas();
1187
1188 };

```

Apresenta-se na listagem 6.38 o arquivo de implementação da classe CSimuladorBlackOil.

Listing 6.38: Arquivo de implementação da classe CSimuladorBlackOil.

```

1189#include <iostream>
1190#include <string>
1191#include <fstream>
1192#include <iomanip>
1193
1194#include "CSimuladorBlackOil.h"
1195
1196/**
1197@class CSimuladorBlackOil
1198@file CSimuladorBlackOil.cpp
1199 */
1200using namespace std;
1201
1202 string linha = "
    -----
    \n";
1203
1204 ///Menu de execucao
1205 int CSimuladorBlackOil:: Executar()
1206 {
1207     cout << "
    -----
    " << endl;
1208     cout << "|UNIVERSIDADE_ESTADUAL_DO_NORTE_FLUMINENSE-
    DARCY_RIBEIRO|" << endl;
1209     cout << "|CENTRO_DE_CIENCIAS_E_TECNOLOGIA-
    CCT|" << endl;
1210     cout << "|LABORATORIO_DE_ENGENHARIA_E_EXPLORACAO_DE
    PETROLEO_LENEP|" << endl;
1211     cout << "|DISCIPLINA:PROPRAMACAO_PRATICA-
    PROJETO_C++|" << endl;
1212     cout << "|PROFESSOR:ANDRE_DUARTE_BUENO
    |" << endl;
1213     cout << "|ALUNOS:JOAO_BOSCO_MACIEL_FILHO
    |" << endl;
1214     cout << "|SUSANE_CHELLI_CAIRES_GOIS
    |" << endl;
1215     cout << "|PROGRAMA_PARA_CALCULO_DE_PARAMETROS_DE_RESERVATORIO
    USANDO_A_MODELAGEM_BLACK-OIL|" << endl;
1216     cout << "
    -----
    " << endl;
1217
1218 ///Solicita a entrada de dados
1219 EntradadeDados();
1220
1221 ///Gera o vetor de pressoes, seja de um ponto especifico ou de um

```

```

        intervalo
1222     fblackoil.Gera_vp(fblackoil.p, fblackoil.pmax, fblackoil.deltap);
1223
1224     int resp;
1225     do {
1226         do{
1227             cout << linha;
1228             cout << linha;
1229             cout << "Qual_parametro_calcular:" << endl;
1230             cout << "1_Pressao_de_bolha_Pb" << endl;
1231             cout << "2_Razao_de_Solubilidade_Rs" << endl;
1232             cout << "3_Fator_Volume_de_Formacao_do_Oleo_Bo" << endl;
1233             cout << "4_Compressibilidade_do_oleo_co" << endl;
1234             cout << "5_Fator_de_Compressibilidade_do_gas_Z" << endl;
1235             cout << "6_Fator_Volume_Formacao_do_Gas_Bg" << endl;
1236             cout << "0_Sair" << endl;
1237             cout << linha;
1238             cin >> resp; cin.get();
1239
1240             if((resp < 0) || (resp > 6))
1241                 {cout << "Opcao_Invalida!" << endl;};
1242
1243             }while ((resp < 0) && (resp > 6));
1244
1245         switch (resp) {
1246
1247         case 1: CalculoPressaoBolha();                break;
1248
1249         case 2: CalculoRazaodeSolubilidade();          break;
1250
1251         case 3: CalculoFatorVolumeFormacaoOleo();      break;
1252
1253         case 4: CalculoCompressibilidadeOleo();        break;
1254
1255         case 5: CalculoFatordeCompressibilidadeZ();    break;
1256
1257         case 6: CalculoFatorVolumeFormacaoGas();       break;
1258
1259         case 0 : return 0;
1260
1261         }
1262     } while (resp != 0);
1263 }
1264
1265 /// Metodo que solicita a entrada de dados pelo teclado ou pelo disco
1266 void CSimuladorBlackOil::EntradadeDados(){
1267     string nomearquivo;
1268     char opcao;

```



```

1269     do {
1270         cout << linha;
1271         cout << "Como_deseja_entrar_com_os_dados:\n";
1272         cout << "1-Teclado\n";
1273         cout << "2-Arquivo_de_disco\n";
1274         cout << linha;
1275         cin >> opcao; cin.get();
1276
1277         ///Entrada de dados pelo teclado
1278         if(opcao == '1'){
1279             cout << linha;
1280             cout << "Entrada_de_dados_pelo_teclado" << endl;
1281             cout << linha;
1282             cin >> fblackoil;}
1283         /// Entrada de dados pelo disco
1284         else if (opcao == '2'){
1285             cout << linha;
1286             cout << "Entrada_de_dados_por_arquivo_de_disco" << endl;
1287             cout << linha;
1288             cout << "Informe_o_nome_do_arquivo_com_a_extensao(ex:.txt,_.
                dat)" << endl;
1289             getline(cin, nomearquivo);
1290             ifstream fin;
1291             fin.open(nomearquivo.c_str());
1292             fin >> fblackoil;}
1293         /// Mensagem de erro se a escolha for errada
1294         if ((opcao != '1') && (opcao != '2'))
1295             {cout << "Opcao_Invalida" << endl;}
1296         }while ((opcao != '1') && (opcao != '2'));
1297
1298     }
1299
1300 ///Metodo que calcula a pressao de bolha
1301 void CSimuladorBlackOil:: CalculoPressaoBolha() {
1302     char resp1;
1303     do{
1304         cout << linha;
1305         cout << "1-Pressao_de_bolha-Pb" << endl;
1306         cout << linha;
1307         cout << "Qual_correlacao_deseja_usar:" << endl;
1308         cout << "1-Standing\n";
1309         cout << "2-Vasquez_e_Beggs" << endl;
1310         cout << linha;
1311         cin >> resp1; cin.get();
1312
1313         if((resp1 != '1') && (resp1 != '2'))
1314             {cout << "Opcao_Invalida!" << endl;};
1315

```

```

1316     }while((resp1!='1') && (resp1!='2'));
1317
1318     switch (resp1) {
1319         /// Calculo da pressao de bolha por Standing
1320         case '1':
1321
1322             cout << "Pb_=" << standing_pb.Pb(fblackoil) << "_psia" << endl;
1323
1324             break;
1325
1326         /// Calculo da pressao de bolha por Vasquez e Beggs
1327         case '2':
1328
1329             cout << "Pb_=" << vb_pb.Pb(fblackoil) << "_psia" << endl;
1330
1331             break;
1332     }
1333
1334 }
1335
1336 ///Metodo que calcula a razao de solubilidade
1337 void CSimuladorBlackOil:: CalculoRazaodeSolubilidade() {
1338     char resp2;
1339     do{
1340         cout << linha;
1341         cout << "_2_-Razao_de_Solubilidade_-Rs" << endl;
1342         cout << linha;
1343         cout << "Qual_correlacao_se_desejar_usar:" << endl;
1344         cout << "_1_-Standing\n";
1345         cout << "_2_-Vasquez_e_Beggs" << endl;
1346         cout << linha;
1347         cin >> resp2; cin.get();
1348
1349         if((resp2!='1') && (resp2!='2'))
1350             {cout << "Opcao_Invalida!" << endl;};
1351
1352     }while((resp2!='1') && (resp2!='2'));
1353
1354     switch (resp2) {
1355
1356         /// Calculo da Razao de solubilidade por Standing.
1357         /// Sera necessario o calculo de outras propriedades, estas tambem
1358         serao calculadas por Standing
1359         case '1':
1360             standing_pb.Pb(fblackoil);
1361             cout << "Pb_=" << standing_pb.pb << "_psia" << endl;
1362             standing_rs.Rs(fblackoil, standing_pb);

```

```

1363     cout << setw(20) << left << "Pressao_(psia)" << left << setw(15)
        << "Razao_Solubilidade_(scf/STB)" << endl;
1364     for(int i = 0; i<fblackoil.vpressao.size(); i++){
1365         cout << setw(20) << left << fblackoil.vpressao[i] << left <<
            setw(15) << standing_rs.vrs[i] << endl;}
1366         standing_rs.Plot(fblackoil);
1367
1368     break;
1369
1370     /// Calculo da Razao de solubilidade por Vasquez e Beggs.
1371     /// Sera necessario o calculo de outras propriedades, estas tambem
        serao calculadas por Vasquez e Beggs
1372     case '2':
1373
1374         cout << "Pb_=" << vb_pb.Pb(fblackoil) << "_psia" << endl;
1375         vb_rs.Rs(fblackoil, vb_pb);
1376
1377         cout << left << setw(20) << "Pressao_(psia)" << left << setw(15)
            << "Razao_Solubilidade_(scf/STB)" << endl;
1378         for(int i = 0; i<fblackoil.vpressao.size(); i++){
1379             cout << setw(20) << left << fblackoil.vpressao[i] << left <<
                setw(15) << vb_rs.vrs[i] << endl;}
1380         vb_rs.Plot(fblackoil);
1381
1382     break;
1383 }
1384 }
1385
1386 ///Metodo que calcula o fator volume de formacao do oleo
1387 void CSimuladorBlackOil::CalculoFatorVolumeFormacaoOleo() {
1388     char resp3;
1389     do{
1390         cout << linha;
1391         cout << "_3_Fator_Volume_de_Formacao_do_Oleo_Bo" << endl;
1392         cout << linha;
1393         cout << "Qual_correlacao_se_desejar_usar:" << endl;
1394         cout << "_1_Standing\n";
1395         cout << "_2_Vasquez_e_Beggs" << endl;
1396         cout << linha;
1397         cin >> resp3; cin.get();
1398
1399         if((resp3!='1') && (resp3!='2'))
1400             {cout << "Opcao_Invalida!" << endl;};
1401
1402     }while((resp3!='1') && (resp3!='2'));
1403
1404     switch (resp3) {
1405

```

```

1406      /// Calculo do Fator volume de formacao do oleo no ponto de bolha
1407      por Standing,
1408
1409      standing_bo.Bob(fblackoil);
1410      cout << "Fator_Volume_de_Formacao_do_Oleo_no_ponto_de_bolha" <<
1411      endl;
1412      cout << "Bob=" << standing_bo.bob << "bbl/STB" << endl;
1413
1414      break;
1415
1416      /// Calculo do Fator volume de formacao do oleo por Vasquez e Beggs.
1417      /// Sera necessario o calculo de outras propriedades, estas tambem
1418      serao calculadas por Vasquez e Beggs
1419
1420      case '2':
1421
1422      vb_pb.Pb(fblackoil);
1423      vb_rs.Rs(fblackoil, vb_pb);
1424      vb_co.Co(fblackoil, vb_rs);
1425      vb_bo.Bo(fblackoil, vb_pb, vb_rs, vb_co);
1426      cout << "Pressao_no_ponto_de_bolha" << endl;
1427      cout << "Pb=" << vb_pb.pb << "psia" << endl;
1428      cout << "Fator_Volume_de_Formacao_do_Oleo_no_ponto_de_bolha" <<
1429      endl;
1430      cout << "Bob=" << vb_bo.bob << "bbl/STB" << endl;
1431      cout << left << setw(20) << "Pressao(psia)" << left << setw(15)
1432      << "Fator_Volume_de_Formacao_do_Oleo(bbl/STB)" << endl;
1433      for(int i = 0; i<fblackoil.vpressao.size(); i++){
1434          cout << setw(20) << left << fblackoil.vpressao[i] << left <<
1435          setw(15) << vb_bo.vbo[i] << endl;}
1436      vb_bo.Plot(fblackoil);
1437
1438      break;
1439      }
1440  }
1441
1442  ///Metodo que calcula a compressibilidade do oleo
1443  void CSimuladorBlackOil::CalculoCompressibilidadeOleo(){
1444      cout << linha;
1445      cout << "4-Compressibilidade_do_oleo-co" << endl;
1446
1447      /// Calculo da Compressibilidade do oleo por Vasquez e Beggs.
1448      /// Sera necessario o calculo de outras propriedades, estas tambem
1449      serao calculadas por Vasquez e Beggs
1450
1451      vb_pb.Pb(fblackoil);
1452      vb_rs.Rs(fblackoil, vb_pb);
1453      vb_co.Co(fblackoil, vb_rs);
1454      cout << left << setw(20) << "Pressao(psia)" << left << setw(15) <<

```

```

        "Compressibilidade do Oleo (1/psia)" << endl;
1447 for(int i = 0; i < fblackoil.vpressao.size(); i++){
1448     cout << setw(20) << left << fblackoil.vpressao[i] << left << setw
        (15) << vb_co.vco[i] << endl;}
1449 vb_co.Plot(fblackoil);
1450
1451 }
1452
1453 ///Metodo que calcula o fator de compressibilidade do gas Z
1454 void CSimuladorBlackOil::CalculoFatordeCompressibilidadeZ() {
1455     char resp4;
1456     do{
1457         cout << linha;
1458         cout << "5- Fator de Compressibilidade do gas Z" << endl;
1459         cout << linha;
1460         cout << "Qual correlacao se deseja usar:" << endl;
1461         cout << "1- Papay\n";
1462         cout << "2- Beggs e Brill" << endl;
1463         cout << linha;
1464         cin >> resp4; cin.get();
1465
1466         if((resp4 != '1') && (resp4 != '2'))
1467             {cout << "Opcao Invalida!" << endl;};
1468
1469     }while((resp4 != '1') && (resp4 != '2'));
1470
1471     /// Calculo da Pressao e Temperatura pseudo-reduzida por Sutton
1472     sutton_prtr.Pr(fblackoil);
1473     sutton_prtr.Tr(fblackoil);
1474
1475     switch (resp4) {
1476
1477     /// Calculo do Fator Z de compressibilidade do gas por Papay
1478     case '1':
1479
1480         papay_z.Z(sutton_prtr);
1481         cout << left << setw(20) << "Pressao (psia)" << left << setw(15) <<
            "Fator de Compressibilidade do Gas" << endl;
1482         for(int i = 0; i < fblackoil.vpressao.size(); i++){
1483             cout << setw(20) << left << fblackoil.vpressao[i] << left << setw
                (15) << papay_z.vz[i] << endl;}
1484         papay_z.Plot(fblackoil);
1485
1486         break;
1487
1488     /// Calculo do Fator Z de compressibilidade do gas por Beggs e Brill
1489     case '2':
1490

```

```

1491     bb_z.Z(sutton_prtr);
1492     cout << left << setw(20) << "Pressao_(psia)" << left << setw(15) <<
        "Fator_de_Compressibilidade_do_Gas" << endl;
1493     for(int i = 0; i<fblackoil.vpressao.size(); i++){
1494         cout << setw(20) << left << fblackoil.vpressao[i] << left << setw
            (15) << bb_z.vz[i] << endl;}
1495     bb_z.Plot(fblackoil);
1496
1497         break;
1498     }
1499
1500 }
1501
1502 ///Metodo que calcula o fator volume de formacao do gas
1503 void CSimuladorBlackOil::CalculoFatorVolumeFormacaoGas() {
1504
1505     cout << linha;
1506     cout << "6_Fator_Volume_Formacao_do_Gas-Bg" << endl;
1507     cout << linha;
1508     cout << "Necessario_o_calculo_de_Z,_qual_correlacao_deseja_usar:"
        << endl;
1509     char resp4;
1510     do{
1511         cout << linha;
1512         cout << "Fator_de_Compressibilidade_do_gas-Z" << endl;
1513         cout << "1-Papay\n";
1514         cout << "2-Beggs_e_Brill" << endl;
1515         cout << linha;
1516         cin >> resp4; cin.get();
1517
1518         if((resp4!='1') && (resp4!='2'))
1519             {cout << "Opcao Invalida!" << endl;};
1520
1521     }while((resp4!='1') && (resp4!='2'));
1522
1523     /// Calculo da Pressao e Temperatura pseudo-reduzida por Sutton
1524     sutton_prtr.Pr(fblackoil);
1525     sutton_prtr.Tr(fblackoil);
1526
1527     switch (resp4) {
1528
1529     /// Calculo do Fator volume de formacao do gas usando o Fator Z
        calculado por Papay
1530     case '1':
1531
1532         papay_z.Z(sutton_prtr);
1533         bg.Bg(fblackoil, papay_z);
1534         cout << left << setw(20) << "Pressao_(psia)" << left << setw(15) <<

```

```

        "Fator_Volume_de_Formacao_do_Gas_(bbl/scf)" << endl;
1535     for(int i = 0; i<fblackoil.vpressao.size(); i++){
1536         cout << setw(20) << left << fblackoil.vpressao[i] << left << setw
            (15) << bg.vbg[i] << endl;}
1537     bg.Plot(fblackoil);
1538
1539         break;
1540
1541     /// Calculo do Fator volume de formacao do gas usando o Fator Z
        calculado por Beggs e Brill
1542     case '2':
1543
1544         bb_z.Z(sutton_prtr);
1545         bg.Bg(fblackoil, bb_z);
1546         cout << left << setw(20) << "Pressao_(psia)" << left << setw(15) <<
            "Fator_Volume_de_Formacao_do_Gas_(bbl/scf)" << endl;
1547         for(int i = 0; i<fblackoil.vpressao.size(); i++){
1548             cout << setw(20) << left << fblackoil.vpressao[i] << left << setw
                (15) << bg.vbg[i] << endl;}
1549         bg.Plot(fblackoil);
1550
1551         break;
1552     }
1553 }

```

Apresenta-se na listagem 6.39 o programa que usa a classe main.

Listing 6.39: Arquivo de implementação da função main().

```

1558 #include "CSimuladorBlackOil.h"
1559
1560 using namespace std;
1561
1562 int main()
1563 {
1564     /// Objeto da classe CSimuladorBlackOil
1565     CSimuladorBlackOil simulador;
1566     /// Executa o simulador
1567     simulador.Executar();
1568     return 0;
1569 }

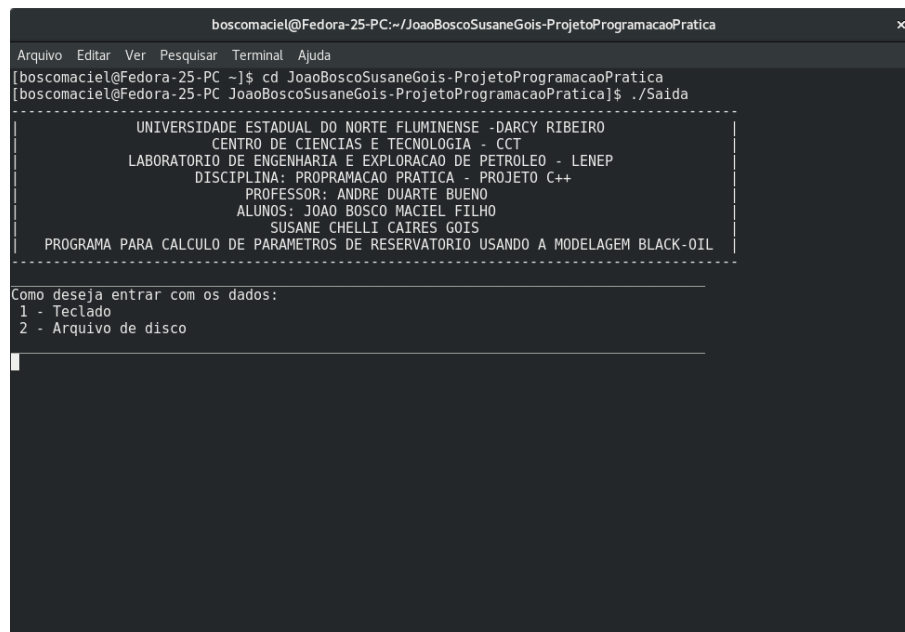
```

Capítulo 7

Teste

7.1 Teste 1: Entrada de dados pelo teclado

Neste primeiro teste, foi escolhido a entrada de dados pelo teclado (Veja Figura 7.1) , onde o usuário deverá digitar os dados solicitados na ordem correta (Veja Figura 7.2). Foi testada a opção de calcular o Ponto de bolha deste reservatório na correlação de Vasquez e Beggs (Veja Figura 7.3) e em seguida o Fator volume de formação do óleo no ponto de bolha utilizando a correlação de Standing (Veja Figura 7.4).



```
boscomaciel@Fedora-25-PC:~/JoaoBoscoSusaneGois-ProjetoProgramacaoPratica
Arquivo Editar Ver Pesquisar Terminal Ajuda
[boscomaciel@Fedora-25-PC ~]$ cd JoaoBoscoSusaneGois-ProjetoProgramacaoPratica
[boscomaciel@Fedora-25-PC JoaoBoscoSusaneGois-ProjetoProgramacaoPratica]$ ./Saida
-----
UNIVERSIDADE ESTADUAL DO NORTE FLUMINENSE -DARCY RIBEIRO
CENTRO DE CIENCIAS E TECNOLOGIA - CCT
LABORATORIO DE ENGENHARIA E EXPLORACAO DE PETROLEO - LENEP
DISCIPLINA: PROGRAMACAO PRATICA - PROJETO C++
PROFESSOR: ANDRE DUARTE BUENO
ALUNOS: JOAO BOSCO MACIEL FILHO
SUSANE CHELLI CAIRES GOIS
PROGRAMA PARA CALCULO DE PARAMETROS DE RESERVATORIO USANDO A MODELAGEM BLACK-OIL
-----

Como deseja entrar com os dados:
1 - Teclado
2 - Arquivo de disco
```

Figura 7.1: Tela do programa mostrando o menu de escolha da forma de entrada dos dados


```

boscomaciel@Fedora-25-PC:~/JoaoBoscoSusaneGois-ProjetoProgramacaoPratica
Arquivo Editar Ver Pesquisar Terminal Ajuda
Como deseja entrar com os dados:
1 - Teclado
2 - Arquivo de disco

1

Entrada de dados pelo teclado

Entre com os valores das variaveis conforme sequencia:
1 - Grau API
20
2 - Temperatura no Reservatorio (em Fahrenheit)
180
3 - Razao de Solubilidade no ponto de bolha = Rsb (em scf/STB)
225
4 - Pressao do ponto desejado (em Psia) (caso queira calcular de varios pontos digitar 0 para essa variavel)
1500
5 - Pressao maxima do reservatorio (caso queira calcular em um unico ponto digitar 0 para essa variavel)
0
6 - Variacao de pressao desejada para acrescimo de pressao
0
7 - Densidade do gas no separador
0.8088
8 - Densidade do gas
0.8
9 - Densidade do oleo
0.95

```

Figura 7.2: Inserção dos dados pelo usuário utilizando o teclado

```

boscomaciel@Fedora-25-PC:~/JoaoBoscoSusaneGois-ProjetoProgramacaoPratica
Arquivo Editar Ver Pesquisar Terminal Ajuda
9 - Densidade do oleo
0.95

Qual parametro calcular:
1 - Pressao de bolha - Pb
2 - Razao de Solubilidade - Rs
3 - Fator Volume de Formacao do Oleo - Bo
4 - Compressibilidade do oleo - co
5 - Fator de Compressibilidade do gas - Z
6 - Fator Volume Formacao do Gas - Bg
0 - Sair

1

1 - Pressao de bolha - Pb

Qual correlacao deseja usar:
1 - Standing
2 - Vasquez e Beggs

2

Pb = 1711.46 psia

Qual parametro calcular:
1 - Pressao de bolha - Pb
2 - Razao de Solubilidade - Rs
3 - Fator Volume de Formacao do Oleo - Bo
4 - Compressibilidade do oleo - co
5 - Fator de Compressibilidade do gas - Z
6 - Fator Volume Formacao do Gas - Bg
0 - Sair

```

Figura 7.3: Cálculo da pressão de bolha pela correlação de Vasquez e Beggs

```

boscomaciel@Fedora-25-PC:~/JoaoBoscoSusaneGois-ProjetoProgramacaoPratica
Arquivo Editar Ver Pesquisar Terminal Ajuda
Pb = 1711.46 psia

Qual parametro calcular:
1 - Pressao de bolha - Pb
2 - Razao de Solubilidade - Rs
3 - Fator Volume de Formacao do Oleo - Bo
4 - Compressibilidade do oleo - co
5 - Fator de Compressibilidade do gas - Z
6 - Fator Volume Formacao do Gas - Bg
0 - Sair

3

3 - Fator Volume de Formacao do Oleo - Bo

Qual correlacao se desejar usar:
1 - Standing
2 - Vasquez e Beggs

1
Fator Volume de Formacao do Oleo no ponto de bolha
Bob = 1.1554 bbl/STB

Qual parametro calcular:
1 - Pressao de bolha - Pb
2 - Razao de Solubilidade - Rs
3 - Fator Volume de Formacao do Oleo - Bo
4 - Compressibilidade do oleo - co
5 - Fator de Compressibilidade do gas - Z
6 - Fator Volume Formacao do Gas - Bg
0 - Sair

```

Figura 7.4: Cálculo do fator volume de formação do óleo no ponto de bolha pela correlação de Standing

7.2 Teste 2: Estrada de dados por arquivo de disco

Neste segundo teste, foi escolhido a entrada de dados por arquivo de disco, onde o usuário deve informar o nome do arquivo juntamente com sua extensão que contém os dados (Veja Figura 7.2), também foi testada a função de cálculo dos parâmetros em diversos pontos de acordo com a variação de pressão, lembrando que esta função não é exclusiva da entrada de dados por arquivo de disco, como também o cálculo de parâmetros em um único ponto não é exclusivo da entrada de dados pelo teclado, isto será melhor abordado no próximo capítulo.

Após a inserir o nome do arquivo foi testado o cálculo do Fator Z de compressibilidade do gás pela correlação de Papay (Veja Figura 7.5). Os resultados além de serem mostrados na tela (Veja Figura 7.6), também são plotados pelo programa Gnuplot (Veja Figura 7.7). Em seguida foi testado o cálculo da Razão de Solubilidade do reservatório pela correlação de Vasquez e Beggs (Veja Figura 7.8), e simililarmente ao exemplo anterior, os resultados foram mostrados na tela (Veja Figura 7.8) e plotados pelo Gnuplot (Veja Figura 7.9).

```

boscomaciel@Fedora-25-PC:~/JoaoBoscoSusaneGois-ProjetoProgramacaoPratica
Arquivo Editar Ver Pesquisar Terminal Ajuda
boscomaciel@Fedora-25-PC:~/JoaoBoscoSusaneGois-ProjetoProgramacaoPratica$ ./saida
-----
UNIVERSIDADE ESTADUAL DO NORTE FLUMINENSE -DARCY RIBEIRO
CENTRO DE CIENCIAS E TECNOLOGIA - CCT
LABORATORIO DE ENGENHARIA E EXPLORACAO DE PETROLEO - LENEP
DISCIPLINA: PROGRAMACAO PRATICA - PROJETO C++
PROFESSOR: ANDRE DUARTE BUENO
ALUNOS: JOAO BOSCO MACIEL FILHO
SUSANE CHELLI CAIRES GOIS
PROGRAMA PARA CALCULO DE PARAMETROS DE RESERVATORIO USANDO A MODELAGEM BLACK-OIL
-----

Como deseja entrar com os dados:
1 - Teclado
2 - Arquivo de disco
2

Entrada de dados por arquivo de disco

Informe o nome do arquivo com a extensao (ex: .txt, .dat)
Dadosreservatorio.txt

Qual parametro calcular:
1 - Pressao de bolha - Pb
2 - Razao de Solubilidade - Rs
3 - Fator Volume de Formacao do Oleo - Bo
4 - Compressibilidade do oleo - co
5 - Fator de Compressibilidade do gas - Z
6 - Fator Volume Formacao do Gas - Bg
0 - Sair
5

5 - Fator de Compressibilidade do gas - Z

Qual correlacao se desejar usar:
1 - Papay
2 - Beggs e Brill
1

Pressao (psia)      Fator de Compressibilidade do Gas
14.7                0.99738
214.7               0.963344
414.7               0.922207

```

Figura 7.5: Inserção de dados por arquivo de disco e escolha de calcular Z pela correlação Papay

```

boscomaciel@Fedora-25-PC:~/JoaoBoscoSusaneGois-ProjetoProgramacaoPratica
Arquivo Editar Ver Pesquisar Terminal Ajuda

5 - Fator de Compressibilidade do gas - Z

Qual correlacao se deseja usar:
1 - Papay
2 - Beggs e Brill

1
Pressao (psia)      Fator de Compressibilidade do Gas
14.7                0.99738
214.7               0.963344
414.7               0.932297
614.7               0.90424
814.7               0.879172
1014.7              0.857094
1214.7              0.838006
1414.7              0.821907
1614.7              0.808798
1814.7              0.798678
2014.7              0.791548
2214.7              0.787408
2414.7              0.786257
2614.7              0.788096
2814.7              0.792925
3014.7              0.800743
3214.7              0.811551
3414.7              0.825348
3614.7              0.842135
3814.7              0.861912
4014.7              0.884678
4214.7              0.910434
4414.7              0.93918
4614.7              0.970915
4814.7              1.00564

Qual parametro calcular:
1 - Pressao de bolha - Pb
2 - Razao de Solubilidade - Rs
3 - Fator Volume de Formacao do Oleo - Bo
4 - Compressibilidade do oleo - co
5 - Fator de Compressibilidade do gas - Z
6 - Fator Volume Formacao do Gas - Bg
0 - Sair

```

Figura 7.6: Valores do Fator Z para variados valores de pressão neste reservatório

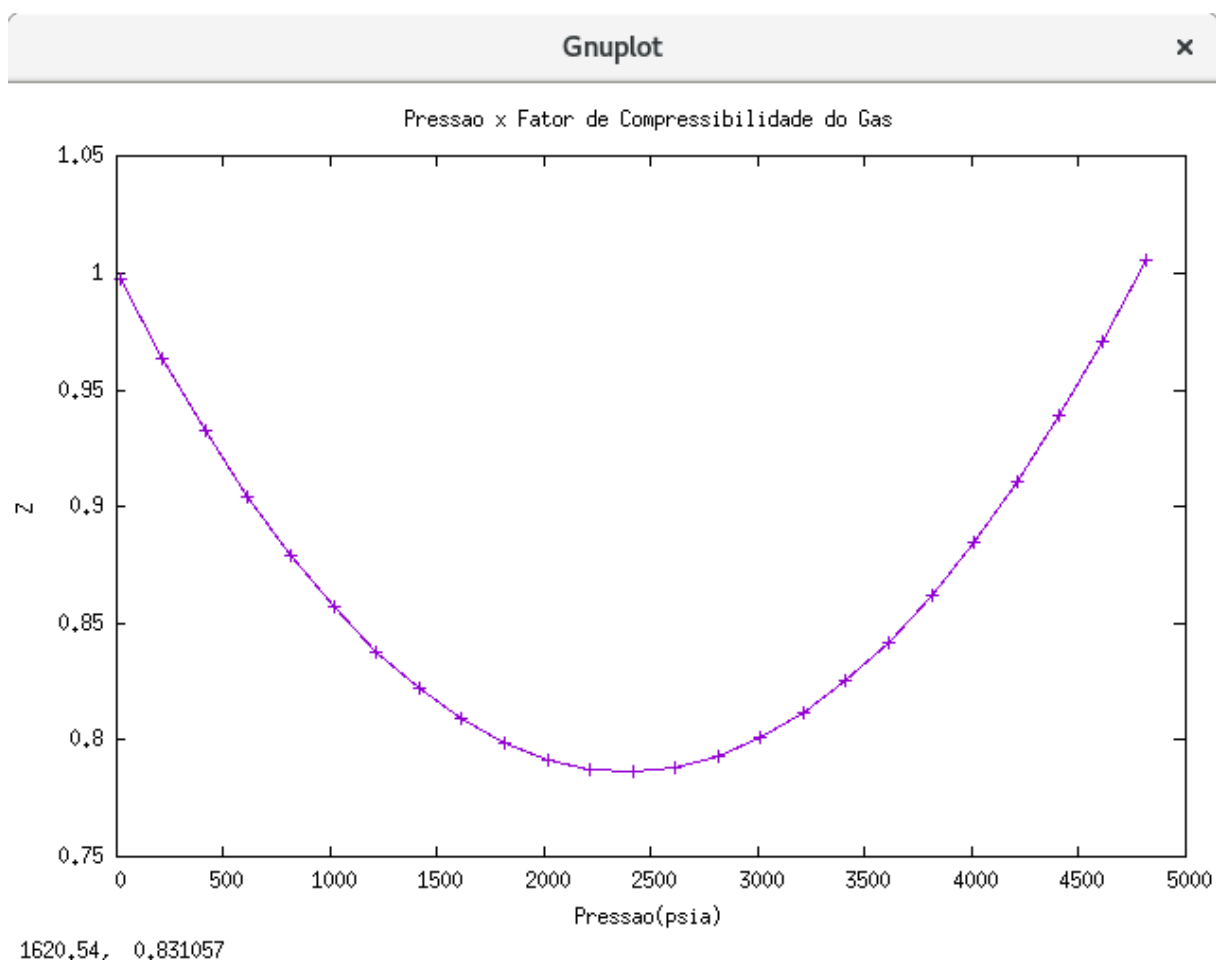


Figura 7.7: Gráfico do Fator Z versus Pressão gerado pelo Gnuplot

```

boscomaciel@Fedora-25-PC:~/JoaoBoscoSusaneGois-ProjetoProgramacaoPratica
Arquivo Editar Ver Pesquisar Terminal Ajuda

Qual parametro calcular:
1 - Pressao de bolha - Pb
2 - Razao de Solubilidade - Rs
3 - Fator Volume de Formacao do Oleo - Bo
4 - Compressibilidade do oleo - co
5 - Fator de Compressibilidade do gas - Z
6 - Fator Volume Formacao do Gas - Bg
0 - Sair

2

2 - Razao de Solubilidade - Rs

Qual correlacao se deseja usar:
1 - Standing
2 - Vasquez e Beggs

2
Pb = 1117.37 psia
Pressao (psia)      Razao Solubilidade (scf/STB)
14.7                1.30651
214.7               31.5058
414.7               68.8265
614.7               109.812
814.7               153.412
1014.7              199.08
1214.7              223.211
1414.7              223.211
1614.7              223.211
1814.7              223.211
2014.7              223.211
2214.7              223.211
2414.7              223.211
2614.7              223.211
2814.7              223.211
3014.7              223.211
3214.7              223.211
3414.7              223.211
3614.7              223.211
3814.7              223.211
4014.7              223.211
4214.7              223.211
4414.7              223.211
4614.7              223.211
4814.7              223.211

```

Figura 7.8: Cálculo da Razão de Solubilidade pela correlação de Vasquez e Beggs

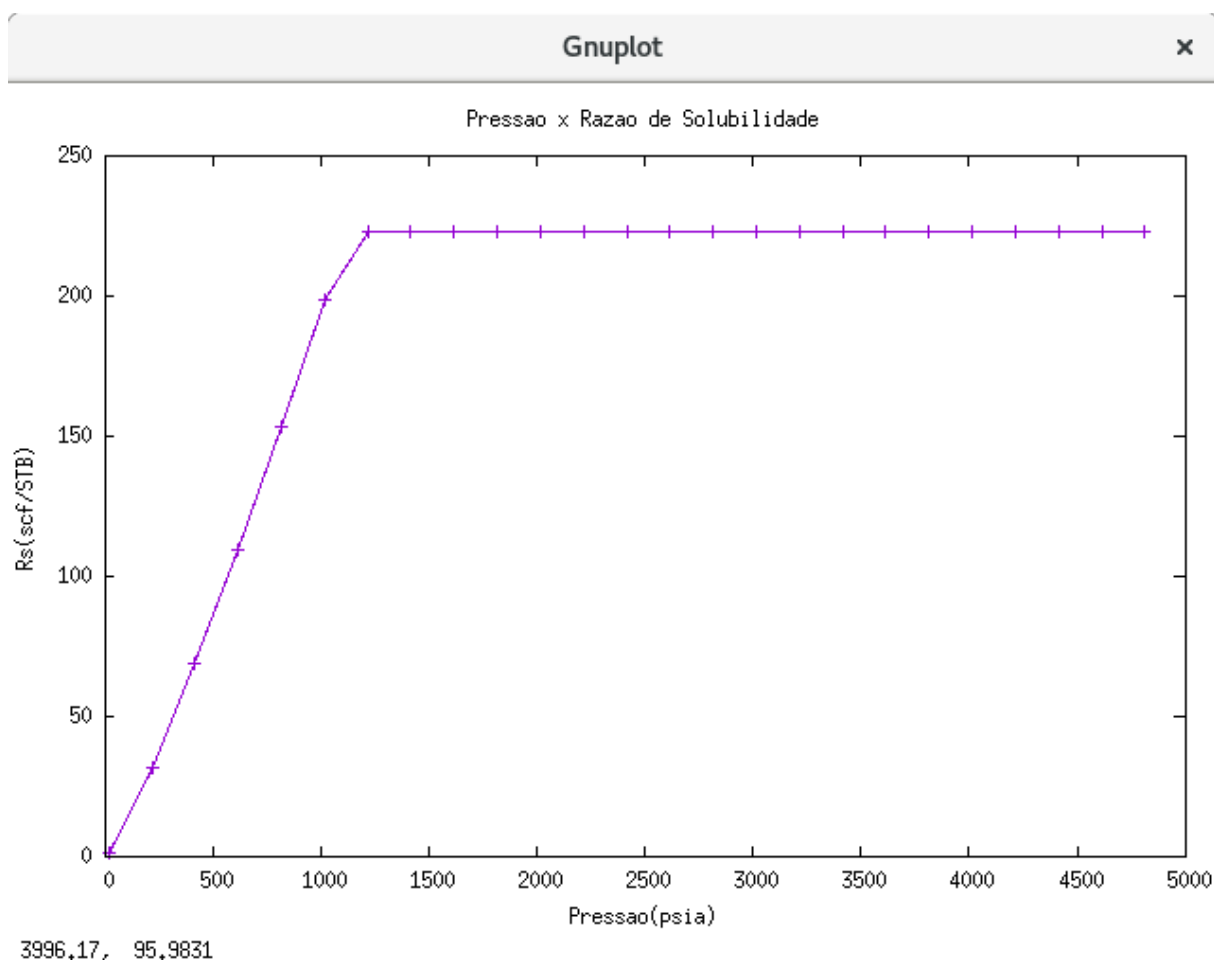


Figura 7.9: Gráfico da Razão de Solubilidade versus Pressão gerado pelo Gnuplot

Capítulo 8

Documentação

Apresenta-se neste capítulo a documentação de uso do software CCPRMB. Esta documentação tem o formato de uma apostila que explica passo a passo como usar o software.

8.1 Documentação do usuário

A seguir encontra-se o manual do usuário, um guia que explica, passo a passo a forma de instalação e uso do software desenvolvido.

8.1.1 Como rodar o software

Abra o terminal, vá para o diretório onde está o código, compile o programa e, depois, execute. Logo após executar, siga os seguintes passos:

1. No menu inicial escolha o tipo de entrada de dados:
 - (a) Opção 1 - Teclado
 - (b) Opção 2 - Arquivo de disco
2. Caso entre com a opção 1 o programa executará o menu de preenchimento pelo teclado e as instruções para ordem de entrada serão dadas, caso escolha a opção 2 referente a leitura de dados de disco o programa irá perguntar o nome do arquivo onde os dados estão salvos. O nome do arquivo de disco deve ser digitado juntamente com o formato da extensão (ex: .txt, .dat).
3. Após o carregamento dos dados, aparecerá o menu com as opções dos parâmetros a serem calculados como segue:
 - (a) Opção 1 - Pressao de bolha - Pb
 - (b) Opção 2 - Razao de Solubilidade - Rs
 - (c) Opção 3 - Fator Volume de Formacao do Oleo - Bo

- (d) Opção 4 - Compressibilidade do óleo - c_o
 - (e) Opção 5 - Fator de Compressibilidade do gás - Z
 - (f) Opção 6 - Fator Volume Formação do Gás - B_g
 - (g) Opção 0 - Sair do programa
4. Caso entre com a opção 1, escolha a opção referente a correlação que deseja utilizar para realizar o cálculo da pressão de bolha conforme segue. Após o cálculo conforme a opção escolhida, o valor da pressão de bolha será mostrado na tela.
- (a) Opção 1 - Standing
 - (b) Opção 2 - Vasquez e Beggs
5. Caso entre com a opção 2, escolha a opção referente a correlação que deseja utilizar para realizar o cálculo da razão de solubilidade. Após o cálculo conforme a opção escolhida, os valores do vetor da razão de solubilidade serão mostrados na tela e será gerado um gráfico da razão de solubilidade em função da pressão utilizando o software externo Gnuplot.
- (a) Opção 1 - Standing
 - (b) Opção 2 - Vasquez e Beggs
6. Caso entre com a opção 3, escolha a opção referente a correlação que deseja utilizar para realizar o cálculo do fator volume de formação do óleo. Após o cálculo conforme a opção escolhida, o vetor com os valores do fator volume de formação do óleo serão mostrados na tela e será gerado um gráfico de B_o em função da pressão utilizando o software externo Gnuplot.
- (a) Opção 1 - Standing
 - (b) Opção 2 - Vasquez e Beggs
7. Caso entre com a opção 4, a correlação de Vasquez e Beggs será utilizada para calcular a compressibilidade do óleo. Após o cálculo o vetor com os valores de compressibilidade será mostrado na tela e gerado um gráfico de C_o em função da pressão utilizando o software externo Gnuplot.
8. Caso entre com a opção 5, escolha a opção referente a correlação que deseja utilizar para realizar o cálculo do fator de compressibilidade do gás. A correlação de Sutton é utilizada para calcular os parâmetros pseudo-críticos e reduzidos de temperatura e pressão necessários tanto para opção 1 e 2. Após o cálculo conforme a opção escolhida, os valores do vetor Z serão mostrados na tela e será gerado um gráfico do fator Z em função da pressão utilizando o software externo Gnuplot.

- (a) Opção 1 - Papay
 - (b) Opção 2 - Beggs e Brill
9. Caso entre com a opção 6, o software irá realizar o cálculo do fator volume de formação do gás, para isso é preciso calcular o fator Z e a escolha da opção referente a correlação que deseja utilizar para este cálculo aparece como segue. Após o cálculo conforme a opção escolhida, o vetor com os valores do fator volume de formação do gás serão mostrados na tela e será gerado um gráfico de B_g em função da pressão utilizando o software externo Gnuplot.
- (a) Opção 1 - Papay
 - (b) Opção 2 - Beggs e Brill
10. Caso entre com a opção 0 o software será encerrado.
11. Caso o usuário digite uma opção que não corresponda as sugeridas em quaisquer que sejam os menus a mensagem “Opção Inválida!” aparecerá na tela e ele terá a opção de refazer a escolha.

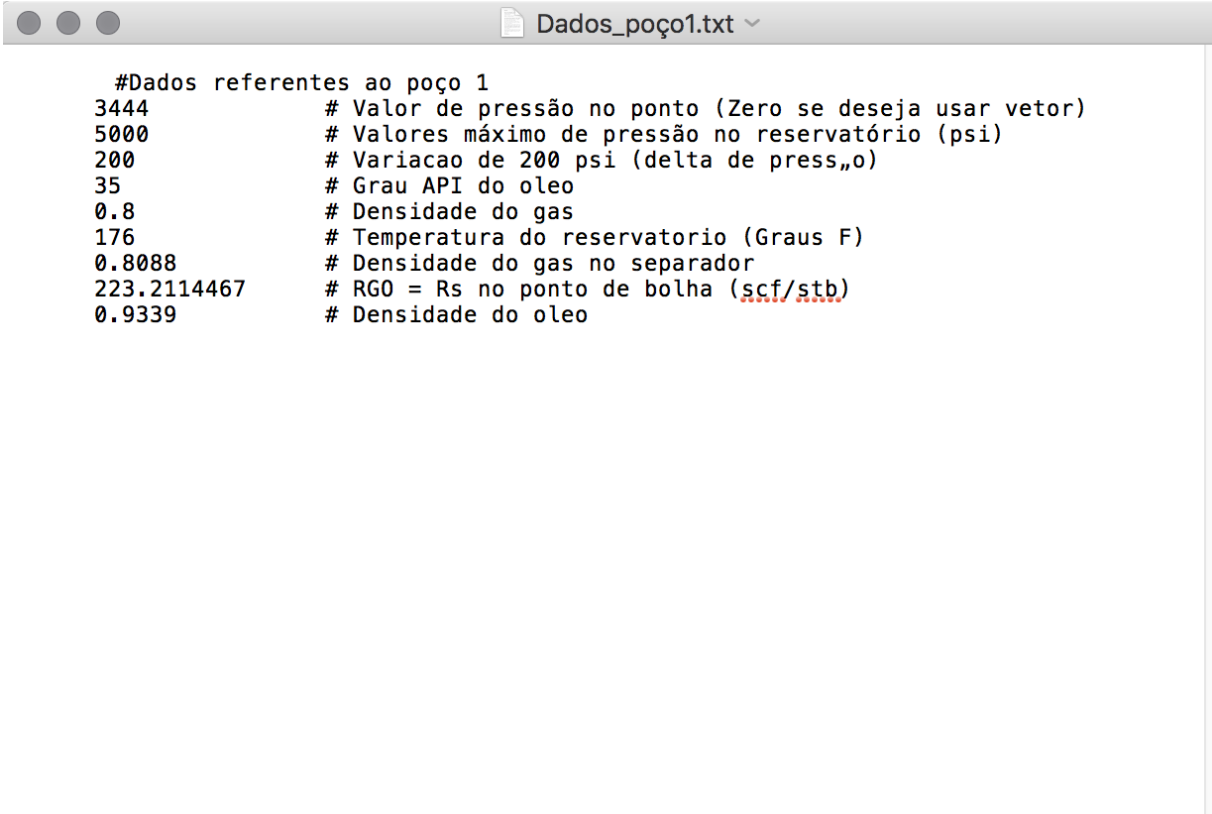
8.2 Documentação para desenvolvedor

Apresenta-se nesta seção a documentação para o desenvolvedor, isto é, informações para usuários que queiram modificar, aperfeiçoar ou ampliar este software.

8.2.1 Dependências

Para compilar o software é necessário atender as seguintes dependências:

- No sistema operacional GNU/Linux:
 - Instalar o compilador `g++` da GNU disponível em [http : //gcc.gnu.org](http://gcc.gnu.org).
 - Para instalar no GNU/Linux use o comando: `yum install gcc`.
- No sistema operacional Windows:
 - Instalar um compilador apropriado.
 - Recomenda-se o Dev C++ disponível em [http : //dev – c.softonic.com.br/](http://dev-c.softonic.com.br/).
- O software gnuplot, deve estar instalado.
 - Gnuplot está disponível no endereço [http : //www.gnuplot.info/](http://www.gnuplot.info/).
 - É possível que haja necessidade de setar o caminho para execução do gnuplot.
- O programa depende da existência de um arquivo de dados (formato `.dat` ou `.txt`), Figura 8.1, para preencher os vetores dos perfis.



```
#Dados referentes ao poço 1
3444      # Valor de pressão no ponto (Zero se deseja usar vetor)
5000      # Valores máximo de pressão no reservatório (psi)
200       # Variacao de 200 psi (delta de press,,o)
35        # Grau API do oleo
0.8       # Densidade do gas
176       # Temperatura do reservatorio (Graus F)
0.8088    # Densidade do gas no separador
223.2114467 # RGO = Rs no ponto de bolha (scf/stb)
0.9339    # Densidade do oleo
```

Figura 8.1: Formato arquivo de dados armazenados no disco

8.2.2 Como gerar a documentação usando doxygen

A documentação do código do software deve ser feita usando o padrão JAVADOC, conforme apresentada no Capítulo - Documentação, do livro texto da disciplina. Depois de documentar o código, use o software `doxygen` para gerar a documentação do desenvolvedor no formato html. O software `doxygen` lê os arquivos com os códigos (*.h e *.cpp) e gera uma documentação muito útil e de fácil navegação no formato html.

- Veja informações sobre uso do formato JAVADOC em:
 - <http://www.stack.nl/~dimitri/doxygen/manual/docblocks.html>
- Veja informações sobre o software `doxygen` em
 - <http://www.stack.nl/~dimitri/doxygen/>

Passos para gerar a documentação usando o `doxygen`.

- Documente o código usando o formato JAVADOC. Um bom exemplo de código documentado é apresentado nos arquivos da biblioteca CGnuplot, abra os arquivos `CGnuplot.h` e `CGnuplot.cpp` no editor de texto e veja como o código foi documentado.

- Abra um terminal.
- Vá para o diretório onde está o código.

```
cd /caminho/para/seu/codigo
```

- Peça para o **doxygen** gerar o arquivo de definições (arquivo que diz para o doxygen como deve ser a documentação).

```
doxygen -g
```

- Peça para o **doxygen** gerar a documentação.

```
doxygen
```

- Verifique a documentação gerada abrindo o arquivo `html/index.html`.

```
firefox html/index.html
```

ou

```
chrome html/index.html
```

Capítulo 9

Sugestões para Trabalhos Futuros

Para um melhoramento do projeto, sugerimos que sejam incorporadas as seguintes expansões:

- Implementar sobrecarga de métodos com utilização de ponteiros.
- Aumentar o número de parâmetros a serem calculados. Ex: Massa específica, viscosidade do óleo.
- Incluir novas correlações.

Referências Bibliográficas

- [Banzer, 1996] Banzer, C. (1996). Correlaciones numericas pvt. *INPELUZ, Maracaibo, Venezuela*. 10
- [Leite, 2016] Leite, R. (2016). Notas de aula: Aula modelagem fluido black-oil. elevacao e escoamento. *Laboratorio de Engenharia e Exploracao de Petroleo LENEP*. 9, 10, 11, 12
- [MACHADO, 2002] MACHADO, J. (2002). Reologia e escoamento de fluidos: enfase na industria do petroleo. editora interciencia. *Petrobras, Rio de Janeiro*. 10
- [Thomas, 2001] Thomas, J. E. (2001). *Fundamentos de engenharia de petroleo*. Interciência. 9
- [Velarde, 1997] Velarde, J. J. (1997). Correlation of black oil properties at pressures below bubble point pressure - a new approach. 10
- [Vieira,] Vieira, R. Notas de aula: Curso de escoamento multifasico. 3