



Universidade Federal
do Rio de Janeiro
Escola Politécnica

SIMULADOR COMPUTACIONAL DE PLANEJAMENTO E PERFURAÇÃO EM TEMPO REAL DE POÇOS VERTICAIS

Victor Costa da Silva

Projeto de Graduação apresentado ao Curso de Engenharia do Petróleo da Escola Politécnica, Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Engenheiro de Petróleo.

Orientador: Paulo Couto, Dr.Eng

RIO DE JANEIRO, RJ – BRASIL
Janeiro de 2012

SIMULADOR COMPUTACIONAL DE PLANEJAMENTO E PERFURAÇÃO EM TEMPO REAL DE POÇOS VERTICAIS

Victor Costa da Silva

PROJETO DE GRADUAÇÃO SUBMETIDO AO CORPO DOCENTE DO CURSO DE ENGENHARIA DO PETRÓLEO DA ESCOLA POLITÉCNICA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE ENGENHEIRO DO PETRÓLEO.

Examinada por:

Paulo Couto, Dr. Eng.

Ibson Paranhos Pasqualino, D. Sc.

Shiniti Ohara, Ph. D.

Gérson Gomes Cunha, D. Sc.

RIO DE JANEIRO, RJ – BRASIL
Janeiro de 2012

Dedicatória

À minha família, meus membros

a Deus, minha mente

a Natalie, meu coração

Agradecimentos

Agradeço acima de tudo a Deus, não só pela vida, mas por estar presente em cada minuto, guiando meus passos e iluminando meu caminho.

Aos meus pais, Robson e Sonia, por cada palavra positiva ou negativa, e pela sólida base que deram. *“Nem os prédios mais fortes resistem com um solo fraco, mas qualquer casebre fica de pé sobre uma rocha”* (anônimo). Muito obrigado meus pais.

Aos meus avós Déa e Wilson. Dizem por aí que avós são pais duas vezes. Esses aí me mostraram que podem ser bem mais que isso. Muito obrigado por tudo.

À minha irmã Luciana, mas um risco certo de Deus no meu caminho.

À Natalie, que apesar de não me acompanhar desde o início foi a minha maior companheira ao longo dos turbulentos anos de minha formação. Por cada palavra de incentivo, por cada acalanto e por não me deixar esmorecer, dedico a você um dos meus mais especiais agradecimentos.

Aos meus amigos, que desde o primeiro dia estiveram do meu lado. Passei por dificuldades, provações e desafios, mas na companhia de vocês foi tudo muito mais prazeroso. Nomes definitivamente não são necessários.

A todos os professores que me ajudaram a compreender o mundo do petróleo, em especial os professores Paulo Couto, meu orientador, por toda a ajuda prestada, incentivo conferido e conhecimento dividido; a Alexandre Leiras, a quem muito perturbei e a quem mais devo ao fim de todo esse tempo; e a Su Jian, a quem devo a possibilidade de estar aqui agora.

Aos profissionais que conheci e me ensinaram muito mais que o teórico e que tenho o grande prazer de chamar de amigos. Marcelo Frydman, Jorge Pastor, Rodrigo Portugal, Shiniti Ohara e tantos outros, obrigado por tudo. Vocês foram fundamentais na modelagem do profissional que hoje me tornei.

A todos os órgãos fomentadores que em algum momento me financiaram, em especial a ANP, através do PRH 02, mas também o CNPq, a CAPES, e a própria UFRJ.

À todos que um dia contribuíram com um gesto, uma palavra, uma oportunidade ou mesmo um sorriso, não só para o desenvolvimento deste trabalho, mas para toda minha jornada. A vocês, um muito obrigado especial.

Resumo do Projeto de Graduação apresentado a Escola Politécnica/UFRJ como parte dos requisitos recomendados para obtenção do grau de Engenheiro de Petróleo.

Proposta de um Simulador Computacional de Planejamento e Perfuração em Tempo Real de Poços Verticais

Victor Costa da Silva

Janeiro de 2012

Orientador: Prof. Paulo Couto, D. Eng.

Curso: Engenharia de Petróleo

Esse trabalho tem por objetivo apresentar um simulador de perfuração desenvolvido em linguagem C++ focando tanto no planejamento do poço quanto na simulação dinâmica do poço, permitindo atuação direta do usuário sobre o caso de perfuração simulado. Além disso será apresentada a modelagem teórica a partir da qual este foi desenvolvido, além dos modelos computacionais implementados que regem o funcionamento do programa, focando nos conceitos relevantes a engenharia de petróleo. Por fim foi realizado um estudo de caso da perfuração de um poço sintético utilizando as ferramentas fornecidas pelo simulador, com seus respectivos resultados apresentados e discutidos, comparando o desempenho da perfuração simulada com a planejada.

Palavras-chave: simulação em tempo real, planejamento de poço, ferramenta de otimização da perfuração, perfil de hidráulica, modelo de perfuração.

Abstract of Undergraduate Project presented to POLI/UFRJ as a partial fulfillment of the requirements for the degree of Engineer

Proposal of a Computational Simulator of Well Planning and Real Time Drilling for Vertical Well

Victor Costa da Silva

January 2012

Advisor: Prof. Paulo Couto, D. Eng.

Major: Engenharia de Petróleo

This project aims to present a drilling simulator developed in C++, focusing on well planning and real time simulation, allowing the interaction of the user over the simulated case. Thus it shall be presented the theoretical approach and concepts that work as the base of the simulator, in addition to the computational models used, that rules the execution of the program, focusing on the relevant concepts to petroleum engineering. Finally was accomplished a synthetic drilling case study using the tools provided by the simulator, with its respective results being presented and discussed, comparing the performance of the simulated drilling well to the planned one.

Keywords: real time drilling, well planning, drilling optimization tool, hydraulic profile, drilling model.

Índice

1. Introdução	1
1.1. Delineação entre Simuladores e a Perfuração	1
1.2. Objetivo do Trabalho	3
1.3. Estruturação do Trabalho	5
2. Abordagem Teórica do Módulo de Planejamento da Perfuração	7
2.2. Projeto das Geopressões.....	8
2.2.1. Aquisição de Dados	10
2.2.2. Massa Específica das Formações.....	11
2.2.4. Gradiente de Pressão de Poros.....	18
2.2.5. Gradiente de Pressão de Fraturas.....	30
2.3. Projeto dos Revestimentos	34
2.3.1. Tipos de revestimento.....	35
2.3.2. Dimensionamento dos Revestimentos	38
2.4. Projeto dos Equipamentos de Segurança - BOP	44
2.5. Projeto de Poço.....	47
2.5.1. Assentamento da Sapata dos Revestimentos	47
2.5.2. Projetos Típicos de Poços	52
3. Abordagem Teórica do Módulo de Perfuração em Tempo Real	55
3.1. Introdução.....	55
3.2. Modelagem da Hidráulica de Poço	55
3.2.1. Hidráulica de poço em condições estáticas.....	56
3.2.2. Hidráulica de poço em condições dinâmicas	57
3.3. Modelagem do Comportamento de Brocas.....	75
3.3.1. Introdução	75
3.3.2. Mecanismos de Fraturamento durante a Perfuração.....	76
3.3.3. Modelo de Perfuração – Estimação do <i>ROP</i>	78
4. Metodologia aplicada no Módulo de Planejamento da Perfuração	86
4.1. Introdução.....	86

4.2.	Características Gerais do Simulador	88
4.3.	Carregamento e Processamento dos Dados.....	90
4.3.1.	Perfil Sísmico e outros Dados Básicos para Planejamento.....	90
4.3.2.	Estimação e Extrapolação das Massas Específicas.....	94
4.3.3.	Estimação das Geopressões	97
4.3.4.	Assentamento das Sapatas	103
4.3.5.	Definição dos Diâmetros de Projeto	105
4.3.6.	Dimensionamento dos Revestimentos	108
4.3.7.	Dimensionamento do BOP	112
4.3.8.	Dimensionamento da coluna de cimento	112
4.3.9.	Modelo de Perfuração e Estimação do Tempo de Perfuração	114
4.3.10.	Estimação do Custo da Perfuração	125
4.3.11.	Resultados Fornecidos pelo Módulo.....	128
5.	Metodologia aplicada no Módulo de Perfuração em Tempo Real.....	130
5.1.	Introdução.....	130
5.2.	Inicializando o módulo e carregamento de dados	130
5.2.1.	Dados do fluido de perfuração	132
5.2.2.	Dados da coluna de perfuração	134
5.2.3.	Dados da broca.....	135
5.3.	Inter-relação e Manipulação das Variáveis	136
5.4.	Blocos Internos do Módulo de Simulação	141
5.4.1.	Bloco de Hidráulica	141
5.4.2.	Bloco de Profundidade por Tempo	150
5.4.3.	Bloco de Estabilidade Poço	157
5.5.	Atuação do Usuário na Simulação	162
5.6.	Resultados Fornecidos pelo Módulo	164
6.	Estudo de Caso, Resultados e Discussões utilizando o Simulador Proposto.....	168
6.1.	Cenário Sintético <i>Proposto</i> para Perfuração.....	168
6.1.1.	Informações Básicas	168
6.1.2.	Testes no Poço de Correlação	170
6.1.3.	Dados do Poço de Correlação	171

6.1.4.	Dados Discretizados por Litologia.....	173
6.1.5.	Custos Associados a Perfuração	174
6.2.	Projeto e Planejamento do Poço PEGASUS	175
6.2.1.	Tratamento Inicial dos Dados	175
6.2.2.	Resultados quanto o cálculo da densidade do poço	177
6.2.3.	Extrapolção do Perfis Sísmico e da Massa Específica.....	178
6.2.4.	Estimação das Geopressões	181
6.2.5.	Assentamento dos Revestimentos.....	185
6.2.6.	Geometria dos Revestimentos.....	186
6.2.7.	Desempenho da Perfuração.....	187
6.2.8.	Estimação do Custo da Perfuração	194
6.2.9.	Demais Resultados do Planejamento	195
6.3.	Simulação em Tempo Real da Perfuração do Poço PEGASUS	198
6.3.1.	Regras para Cenarizar a Simulação	198
6.3.2.	Resultados da Simulação em Tempo Real da Perfuração.....	199
7.	Conclusões e Recomendações Finais.....	216
8.	Referências Bibliográficas	218

Lista de Figuras

Figura 1 – Custo médio de perfuração por pé em poços nos EUA (elaboração própria baseada nos dados do US EIA).....	1
Figura 2 – Simulador de Perfuração da Aker Solutions® (www.AkerSolutions.com)	2
Figura 3 – Interface típica de um simulador de controle de poço (www.WellControl.com.br)	3
Figura 4 – Fluxo de trabalho para cálculo das geopressões ao longo do poço (Rocha <i>et al.</i> , 2008, adaptado).....	9
Figura 5 – Representação esquemática de uma janela operacional de perfuração (elaboração própria).....	10
Figura 6 – Discretização dos trechos para determinação das massas específicas (ROCHA <i>et al.</i> , 2008)	12
Figura 7 – Análise de sensibilidade do método de Gardner frente a modificações nos parâmetros empíricos <i>a</i> e <i>b</i> . (elaboração própria)	14
Figura 8 – Aplicação da correlação de extrapolação das densidades para profundidades rasas (elaboração própria)	16
Figura 9 – Comportamento típico de tensão e de gradiente de sobrecarga ao longo do poço (elaboração própria)	18
Figura 10 – Imagem de uma plataforma de petróleo após sofrer <i>blowout</i> (Frydman, 2010)	19
Figura 11 – Representação esquemática ilustrando zonas de pressão de poros anormalmente altas (elaboração própria)	21
Figura 12 – Perfil de tempo de trânsito ao longo do poço e sua respectiva curva de tendência normal (elaboração própria).....	25
Figura 13 – Perfil de velocidade sísmica para zonas de comportamento perfeitamente elástico, plástico e perfeitamente plástico (elaboração própria)	27
Figura 14 – Comparação entre os resultados dados pelos métodos de Eaton e Bowers (elaboração própria).....	30
Figura 15 – Fratura no entrono de um poço vertical (Fjaer <i>et al.</i> , 2009).....	31
Figura 16 – Curva de κ , calibrada pelos LOTs (elaboração própria)	34
Figura 17 – Tubos para revestimento de um poço (Thomas <i>et al.</i> , 2002)	35
Figura 18 – Exemplos de programas de poço (Bourgoyne <i>et al.</i> , 1986)	36

Figura 19 – Balanço de forças axiais no revestimento (Bourgoyne <i>et al.</i> 1986, adaptado).....	39
Figura 20 – Desenho esquemático de um BOP submarino (Thomas <i>et al.</i> 2002).....	44
Figura 21 – Critério de assentamento de sapatas baseado apenas na janela operacional (elaboração própria).....	50
Figura 22 – Critério de assentamento de sapatas segundo a margem de segurança do riser (elaboração própria).....	52
Figura 23 – Fluxo para escolha da combinação entre diâmetros de poços e de revestimentos, partindo-se do princípio do drift do revestimento (Rocha <i>et al.</i> 2006, adaptado)	53
Figura 24 – Projetos típicos usualmente aplicados em poços rasos ou zonas normalmente pressurizadas (Rocha <i>et al.</i> 2006, adaptado).....	54
Figura 25 – Perfil de pressão hidrostática em um poço com um fluido de 10,5 <i>ppg</i> até 950m e um de 14 <i>ppg</i> daí até o fundo (elaboração própria).	57
Figura 26 – Diagrama ilustrativo de um sistema de circulação de fluidos (Bourgoyne <i>et al.</i> 1986)	59
Figura 27 – Modelos Reológicos (Machado, 2002, adaptado).	64
Figura 28 – Diagrama de Stanton (Bourgoyne <i>et al.</i> 1986, adaptado)	68
Figura 29 – Número de Reynolds crítico para fluidos plásticos de Bingham (Bourgoyne <i>et al.</i> 1986, adaptado).....	72
Figura 30 – Número de Reynolds crítico e fator de fricção para modelo de lei de potências (Bourgoyne <i>et al.</i> 1986, adaptado).....	74
Figura 31 – Brocas tricônicas com insertos de tungstênio e aço, respectivamente (Thomas <i>et al.</i> , 2002)	76
Figura 32 – Mecanismo de perfuração em brocas com partes móveis (Bourgoyne <i>et al.</i> 1986, adaptado).....	77
Figura 33 – Sensibilidade do <i>ROP</i> aos diversos parâmetros da perfuração, no modelo de Bourgoyne (elaboração própria)	80
Figura 34 – Tela inicial do simulador	86
Figura 35 – Fluxograma básico do módulo de planejamento de poço (elaboração própria).....	87
Figura 36 – Tela inicial do módulo de planejamento de poço do simulador <i>proposto</i>	88
Figura 37 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT.	91
Figura 38 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTDT.TXT	92

Figura 39 – Fluxograma dos blocos 1 do módulo planejamento do poço (elaboração própria)...	93
Figura 40 – Fluxograma do bloco 2 – interpolação do perfil de <i>DT</i> (elaboração própria).....	93
Figura 41 – Fluxograma do Bloco 3 do módulo de planejamento da perfuração (elaboração própria).....	95
Figura 42 – Fluxograma para Bloco 4 do módulo de planejamento de poço, focando na extrapolação da massa específica das formações (elaboração própria).....	96
Figura 43 – Fluxograma para Bloco 4 do módulo de planejamento de poço, focando na extrapolação do perfil sísmico (elaboração própria).....	96
Figura 44 – Fluxograma para o cálculo do gradiente de sobrecarga (elaboração própria).....	97
Figura 45 – Tela de escolha de como obter os dados do gradiente de poros.....	98
Figura 46 – Tela de escolha do método a ser aplicado.....	98
Figura 47 – Tela de escolha dos pontos da curva de tendência no método de Eaton.....	99
Figura 48 – Fluxograma para estimação da pressão de poros, segundo Eaton (elaboração própria).....	99
Figura 49 – Fluxograma para definição da curva de tendência para o modelo de Eaton (elaboração própria).....	100
Figura 50 – Tela de opção de entrada de parâmetros no método de Bowers.....	100
Figura 51 – Fluxograma para estimação da pressão de poros pelo método de Bowers (elaboração própria).....	101
Figura 52 – Fluxograma para estimação do gradiente de fratura (elaboração própria).....	102
Figura 53 – Arquivo UFRJDRILLINGSIMULATOR/DEBUG/DEBUGGF.TXT.....	102
Figura 54 – Fluxograma para o critério de assentamento de sapatas baseado na janela operacional (elaboração própria).....	104
Figura 55 – Fluxograma para o critério de assentamento de sapatas baseado na margem de riser (elaboração própria).....	105
Figura 56 – Tela inicial da escolha da forma de carregar os arranjos dos diâmetros.....	105
Figura 57 – Tela do carregamento de diâmetros manualmente.....	106
Figura 58 – Arquivo de entrada UFRJDRILLINGSIMULATOR/INPUTS/INPUTSTAGECONFIG.TXT.....	107
Figura 59 – Tela do carregamento de diâmetros de forma automatizada.....	108
Figura 60 – Fluxograma para o dimensionamento do revestimento frente ao colapso (elaboração própria).....	109

Figura 61 – Fluxograma para dimensionamento do revestimento frente a explosão (elaboração própria).....	110
Figura 62 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTFCOEFICIENT.TXT	111
Figura 63 – Fluxograma para dimensionamento do revestimento frente ao colapso (elaboração própria).....	112
Figura 64 – Fluxograma para o dimensionamento da coluna e do volume de cimento (elaboração própria).....	114
Figura 65 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTBOURGOYNEPARAMETERS.TXT.....	116
Figura 66 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTBITRECORD.TXT.....	117
Figura 67 – Fluxograma genérico para estimação do PT	121
Figura 68 – Fluxograma para atuação do simulador em caso de manobra (elaboração própria)	123
Figura 69 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTCASINGPLACEMENT.TXT.....	124
Figura 70 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTFIXEDCOSTS.TXT	125
Figura 71 – Conjunto de custos variáveis carregados no simulador.....	126
Figura 72 – Tela inicial do módulo de simulação em tempo real.....	131
Figura 73 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTREALTIME_MAIN.TXT	132
Figura 74 – Tela do carregamento de dados de fluido.....	133
Figura 75 – Tela de carregamento dos parâmetros da coluna.....	135
Figura 76 – Tela de carregamento dos parâmetros da broca	136
Figura 77 – Fluxograma básico do funcionamento do módulo de simulação em tempo real (elaboração própria).....	140
Figura 78 – Esquema da divisão das seções para cálculo da perda de carga (elaboração própria).....	142
Figura 79 – Fluxograma para cálculo da perda de carga em fluido newtonianos (elaboração própria).....	144
Figura 80 – Fluxograma para cálculo da perda carga para fluidos binghamianos (elaboração própria).....	146
Figura 81 – Ajuste linear do Re crítico em função do parâmetro n (elaboração própria)	147
Figura 82 – Fluxograma para cálculo da perda de carga para fluidos de lei de potências (elaboração própria).....	148

Figura 83 – Fluxograma para manobra no módulo de simulação em tempo real (elaboração própria).....	155
Figura 84 – Esquema de distribuição do tempo no bloco de assentamento dos revestimentos (elaboração própria)	156
Figura 85 – Tela de carregamento dos dados do revestimento	157
Figura 86 – Diagrama esquemático da zona fraturada para o simulador (elaboração própria) ..	159
Figura 87 – Tela da simulação, com diversos parâmetros de acompanhamento do poço	167
Figura 88 – Diagrama esquemático do poço, contendo as informações originais para projeto (elaboração própria)	169
Figura 89 – Tela do simulador indicando que os dados foram carregados.....	175
Figura 90 – Tela do simulador indicando que os perfil foi interpolado.....	176
Figura 91 – Diagrama do perfil sísmico original e após a interpolação (elaboração própria, baseado nos resultados do simulador proposto).....	176
Figura 92 – Perfil de densidade estimada para $a = 0,23$ e $b = 0,25$ (resultado do simulador proposto)	177
Figura 93 – Densidade estimada para $a = 0,23$ e $b = 0,245$ (resultado do simulador proposto). 178	
Figura 94 – Tela do simulador informando os parâmetros ajustados da curva de extrapolação de densidades	179
Figure 95 – Perfil da massa específica extrapolada (resultado do simulador <i>proposto</i>).....	179
Figura 96 – Tela do simulador para carregamento dos parâmetros da equação invertida de Gardner	180
Figura 97 – Extrapolação do perfil sísmico através do modelo de Gardner (resultado do simulador proposto)	180
Figura 98 – Perfis de tensão e gradiente de sobrecarga ao longo do poço (elaboração própria baseada nos resultado do simulador proposto)	181
Figura 99 – Tela do simulador contendo pontos que definirão a curva de tendência.....	182
Figura 100 – Perfil sísmico do poço PEGASUS com a curva de tendência definida pelo simulador (resultados do simulador proposto).....	183
Figura 101 – Coeficientes de ponderação para suavização da curva de pressão de poros	183
Figura 102 – Perfil de pressão e gradiente de poros calculados pelo simulador (elaboração própria baseada nos resultado do simulador proposto).....	184

Figura 103 – Perfil de pressão e gradiente de fratura calculado pelo simulador (elaboração própria baseada nos resultado do simulador proposto).....	184
Figura 104 – Diagrama de geopressões e profundidade de assentamento dos revestimentos do poço estudado (resultados do simulador proposto).....	185
Figura 105 – Dados do arranjo de diâmetros e parâmetros do aço do revestimento	187
Figura 106 – Diagrama tempo por profundidade estimado para o poço estudado (resultado do simulador)	190
Figura 107 – <i>ROP</i> ao longo do poço, marcando a influência da litologia (resultados do simulador proposto, adaptado).....	192
Figura 108 – <i>ROP</i> ao longo do poço, marcando a influência da troca de brocas (resultados do simulador proposto, adaptado).....	193
Figura 109 – <i>ROP</i> ao longo do poço, marcando a influência do gradiente de poros (resultados do simulador proposto, adaptado).....	194
Figura 110 – Diagrama custo por profundidade estimado para o poço estudado (resultados do simulador proposto)	195
Figura 111 – Vista superior do poço projetado (resultados do simulador <i>proposto</i>).....	197
Figura 112 – Vista frontal do poço projetado (resultados do simulador <i>proposto</i>)	197
Figura 113 – Perfil de desgaste das 21 brocas utilizadas no poço (elaboração própria baseada nos resultados do simulador <i>proposto</i>)	201
Figura 114 – Diagrama tempo por profundidade do poço simulado frente ao poço projetado (resultados do simulador <i>proposto</i>).....	203
Figura 115 – Perfil do <i>ROP</i> ao longo do tempo de perfuração (elaboração própria baseada nos resultados do simulador <i>proposto</i>)	204
Figura 116 – Diagrama da densidade e da <i>ECD</i> frente a janela operacional do poço (resultado do simulador <i>proposto</i>)	205
Figura 117 – Eventos de perdas de circulação (elaboração própria baseada nos resultados do simulador <i>proposto</i>)	206
Figura 118 – Eventos de perdas de circulação, ao longo do tempo (elaboração própria baseada nos resultados do simulador <i>proposto</i>)	207
Figura 119 – Resultados do simulador em um evento de perdas de kick (elaboração própria baseada nos resultados do simulador <i>proposto</i>).....	208

Figura 120 – Resultados do simulador em um evento de kick, em função do tempo (elaboração própria baseada nos resultados do simulador proposto)	209
Figura 121 – Perfil hidráulico do poço estudado (elaboração própria baseada nos resultados do simulador proposto)	210
Figura 122 – Perfil de <i>BHP</i> ao longo do tempo de perfuração (elaboração própria baseada nos resultados do simulador proposto)	211
Figura 123 – Perfil de vazão de bombeio definida para o poço (elaboração própria baseada nos resultados do simulador proposto)	212
Figura 124 – Perfil de pressão e bombeio ao longo do tempo de perfuração (elaboração própria baseada nos resultados do simulador proposto)	212
Figura 125 – Perfil de potência da bomba ao longo da perfuração (elaboração própria baseada nos resultados do simulador proposto)	213
Figura 126 – Perfil do <i>WOH</i> ao longo do tempo de perfuração (elaboração própria baseada nos resultados do simulador proposto)	214
Figura 127 – Enfoque do comportamento do <i>WOH</i> durante uma manobra (elaboração própria baseada nos resultados do simulador proposto)	215

Lista de Tabelas

Tabela 1 – Classificação dos Gradientes da Pressão de Poros, (Rocha <i>et al.</i> 2008, adaptado). ...	20
Tabela 2 – Valores típicos do coeficiente de Eaton	24
Tabela 3 – Graus de aços reconhecidos pela API (Bourgoyne et al.. 1986, adaptado)	38
Tabela 4 – Arquivos que devem estar presente na pasta UFRJDRILLINGSIMULATOR/INPUTS	89
Tabela 5 – Esquemas de diâmetros padrão do simulador	107
Tabela 6 – Tabela de custos considerados e como foram vistos e manipulados pelo simulador	127
Tabela 7 – Tabela mostrando as relações entre as variáveis do problema.....	138
Tabela 8 – Características das seções do poço para cálculo da perda de carga	142
Tabela 9 – Origem da variáveis relevantes para o modelo de Bourgoyne-Young	152
Tabela 10 – Caminhos para atuação do usuário durante a simulação.....	164
Tabela 11 – Dados de RFT obtidos no poço de correlação	171
Tabela 12 – Dados de LOT obtidos no poço de correlação	171
Tabela 13 – Dados referentes ao assentamento dos revestimentos no poço de correlação	172
Tabela 14 – Dados relativos a cada litologia da coluna estratigráfica	174
Tabela 15 – Dados de custos utilizados no cenário de perfuração	175
Tabela 16 – Profundidade de assentamento e densidade para cada fase do poço estudado (elaboração própria baseada nos resultados do simulador proposto).....	186
Tabela 17 – Resultados referentes a Geometria das fases de perfuração do poço estudado (elaboração própria baseada nos resultados do simulador proposto).....	187
Tabela 18 – Brocas utilizadas na perfuração do poço estudado (elaboração própria baseada nos resultados do simulador proposto)	188
Tabela 19 – Tempo total da perfuração do poço estudado (elaboração própria baseada nos resultados do simulador proposto)	189
Tabela 20 – Tempo de perfuração de cada fase do poço estudado (elaboração própria baseada nos resultados do simulador proposto)	190
Table 21 – Dimensionamento da coluna de cimento para as fases do poço estudado (elaboração própria baseada nos resultados do simulador proposto)	196
Tabela 22 – Características gerais das fases do poço simulado (elaboração própria baseada nos resultados do simulador proposto)	200

Tabela 23 – Comparação de desempenho dos poços simulado e planejado (elaboração própria baseada nos resultados do simulador proposto)..... 202

Simbologia – Símbolos Latinos

a	Parâmetro de calibração	[]
A	Área	[in^2]
b	Parâmetro de calibração	[]
BHP	Pressão de fundo de poço	[psi]
BHA	<i>Bottom Hole Assmby</i>	
BOP	<i>Blowout Preventer</i>	
c	Parâmetro de calibração	[]
D	Diâmetro	[in]
DT	Tempo de trânsito compressional	[$\mu s/ft$]
ECD	Densidade equivalente de circulação	[ppg]
f	Fator de atrito	[]
f	Relações funcionais modelo de Bourgoyne-Young	[]
FS	Fator de segurança	[]
F	Força	[lbf]
G	Gradiente	[ppg]
h	Desgaste da broca	[]
h	Net-pay do reservatório	[m]
He	Número de Hedstron	[]
H_1	Parâmetro da broca	[]
H_2	Parâmetro da broca	[]
ID	Diâmetro interno	[in]
J_2	Parâmetro de desgaste	[]

k	Permeabilidade da formação	[<i>md</i>]
K	Índice de consistência do fluido de lei de potências	[<i>eq. cp</i>]
L	Comprimento das seções	[<i>m</i>]
LDA	Lâmina d'água	[<i>m</i>]
LOT	Leak-of Test	
m	Coefficiente angular a curva de tendência de Eaton	[]
n	Índice de comportamento do fluido de lei de potências	[]
NPT	Non-productive time	[<i>h</i>]
OD	Diâmetro externo	[<i>in</i>]
P	Pressão	[<i>psi</i>]
PT	Productive time	[<i>h</i>]
POT	Potência elétrica	[<i>HP</i>]
q	Vazão	[<i>gal/min</i>]
Re	Número de Reynolds	[]
r_e	Raio do reservatório	[<i>in</i>]
r_w	Raio do poço	[<i>in</i>]
ROP	Taxa de penetração	[<i>m/h</i>]
t	Tempo	[<i>h</i>],[<i>s</i>]
U	Parâmetro de descarregamento de Bowers	[]
V	Volume	[<i>bbbl</i>]
V_p	Velocidade sísmica	[<i>ft/s</i>]
\bar{v}	Velocidade média de área	[<i>m/s</i>]
w	Peso específico	[<i>lbf/ft</i>]
WOB	Peso sobre a broca	[<i>lbf</i>]
WOH	Peso sobre o gancho	[<i>lbf</i>]

Z

Profundidade

[*m*],[*ft*]

Simbologia – Símbolos Gregos

α	Fator de conversão de unidades	[]
β	Fator de empuxo	[]
γ	Taxa de cisalhamento	[s^{-1}]
δ	Espessura	[<i>in</i>]
Δ	Varição	
ϵ	Rugosidade	[<i>in</i>]
θ	Leitura do viscosímetro rotacional	[]
κ	Fator de proporcionalidade	[]
λ	Indicador da pressão de poros	
μ	Viscosidade	[<i>cp</i>]
ρ	Massa específica	[<i>ppg</i>]
σ	Tensão	[<i>psi</i>]
τ_0	Ponto de cisalhamento	[<i>psi</i>]
τ	Tensão de cisalhamento	[<i>psi</i>]
τ_H	Abrasividade	[<i>h</i>]
φ	Coefficiente de Eaton	[]

Simbologia – Subscritos

() _r	Rocha
() _{ov}	Sobrecarga
() _{pr}	Poros
() _{fr}	Fratura
() _p	Bomba
() _b	Broca
() _f	Fluido
() _{BOP}	BOP
() _{cim}	Cimento
() _{max}	Máximo
() _{min}	Mínimo
() _s	Sedimentos
() _{br}	Explosão
() _{el}	Elástico
() _p	Plástico
() _a	Aparente
() _j	Jatos
() _{well}	Poço
() _w	Água
() _{cr}	Crítico

() <i>eq</i>	Equivalente
() <i>y</i>	Escoamento
() <i>h</i>	Hidrostática
() <i>MR</i>	Margem de Riser
() <i>n</i>	Normal
() <i>rev</i>	Revestimento
() <i>t</i>	Manobra
() <i>lam</i>	Laminar
() <i>turb</i>	Turbulento
() <i>gas</i>	Gás
() <i>aço</i>	Aço
() <i>dp</i>	<i>Drill pipe</i>
() <i>hwdp</i>	<i>Heavy-weight drill pipe</i>
() <i>dc</i>	<i>Drill collar</i>
() <i>coluna</i>	Coluna
() <i>anular</i>	Anular

1. Introdução

1.1. Delineação entre Simuladores e a Perfuração

O início da atividade de perfuração se confunde com o início da própria atividade de exploração de petróleo pelo mundo. Desde 1860, quando foi perfurado o primeiro poço com objetivo de encontrar petróleo, a engenharia de poços evoluiu proporcionalmente aos desafios da exploração como um todo, onde a otimização da perfuração se tornou um dos grandes objetivos da indústria. A prática é altamente complexa e envolve diversas áreas de conhecimento, dispostas em um fluxograma amplo e denso, objetivando potencializar a atividade como um todo, reduzindo o tempo total de exploração ou desenvolvimento, os custos e, assim, garantindo a segurança e a viabilidade econômica de toda a atividade.

Segundo o portal de economia EconomyWatch.com, os custos com perfuração nos dias de hoje equivalem a aproximadamente 65% do total de investimentos anteriores ao início da extração do petróleo. Além disso, estatísticas disponibilizadas pelo U.S. *Energy Information Administration* (EIA) mostram a tendência crescente do custo médio da perfuração ao longo dos anos, tomando como base os dados de poços perfurados em território americano, como podemos observar na Figura 1.

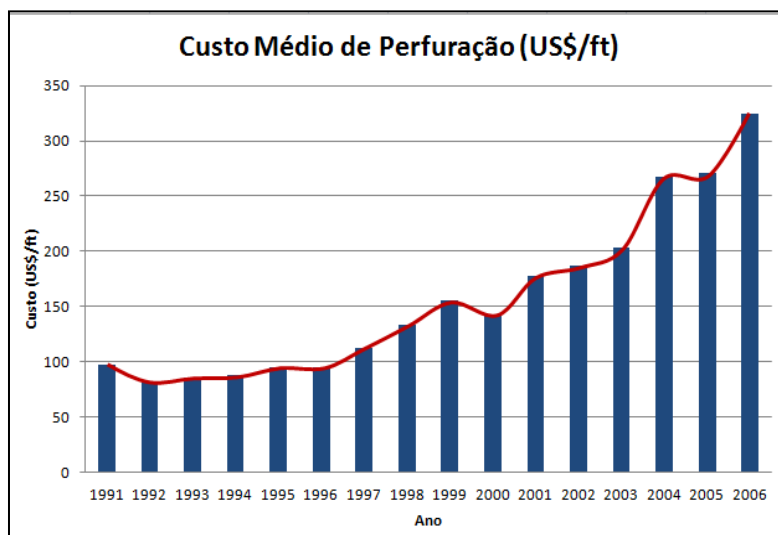


Figura 1 – Custo médio de perfuração por pé em poços nos EUA (elaboração própria baseada nos dados do US EIA)

Sendo assim a necessidade de se aperfeiçoar a atividade de perfuração como forma a reduzir o tempo total, bem como os custos associados a ela, se torna imperativa para indústria nos dias de hoje, particularmente em um contexto onde o horizonte apresentado é cada vez mais desafiador, o que conduz a expectativa de que se ultrapassem as barreiras tecnológicas atuais, deslocando o estado da arte da indústria para níveis consideravelmente mais amplos.

E é sob essa ótica que os simuladores de perfuração ganham importância. Atualmente existem alguns simuladores de perfuração disponíveis no mercado, porém como diferentes objetivos. Existem simuladores cujo foco está na simulação do funcionamento de determinados equipamentos associados à perfuração, como top-drive, BOP, broca, entre outros. Existem também simuladores voltados para o treinamento de sondadores, onde o foco está na simulação de um ambiente similar ao de uma plataforma, onde telas de 180 e 270° simulam as práticas realizadas na perfuração, como manobras, descida de equipamentos, etc., como podemos ver na Figura 2.

Outro grupo de simuladores são os simuladores de projeto e física de poço, onde através de interfaces amigáveis o usuário é capaz de projetar um poço de maneira simples e, a partir daí, implementar um processo dinâmico da perfuração desse poço projetado. Nesse grupo os usuários lidam com todos os parâmetros relevantes na perfuração, podendo avaliar, de maneira direta, como cada decisão tomada pode afetar a perfuração daquele poço, permitindo que uma nova e mais apurada estratégia de perfuração seja adotada de forma a otimizar o tempo e a segurança de todo o processo, como podemos ver na Figura 3. E é nesse grupo que o simulador proposto nesse se insere.

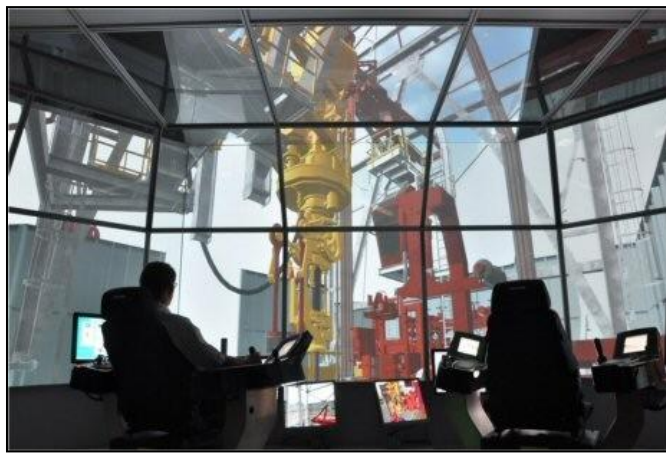


Figura 2 – Simulador de Perfuração da Aker Solutions® (www.AkerSolutions.com)

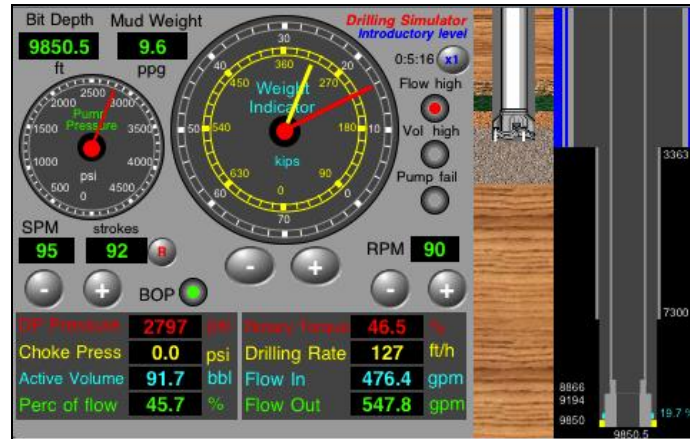


Figura 3 – Interface típica de um simulador de controle de poço (www.WellControl.com.br)

1.2. Objetivo do Trabalho

Seguindo a motivação descrita anteriormente o trabalho em questão tem como objetivo o desenvolvimento de um simulador computacional de perfuração para poços verticais, tanto em terra quanto no mar, utilizando a linguagem C++, através do compilador DevC++®. Toda a parte gráfica do simulador – diagramas, figuras e gráficos relevantes para o fluxograma proposto – será desenvolvido utilizando a biblioteca GD_Paul®, que foi escrita pelo professor M.Sc. Paul Merchant do Departamento de Engenharia da Washington and Lee University, Lexington, VA, sendo gentilmente cedida para o desenvolvimento deste trabalho. A interface utilizada será a DOS, que é produto direto da compilação do código.

O simulador citado será composto por dois módulos dependentes: planejamento de poço e simulador em dinâmica *real-time*. O primeiro será capaz de absorver uma série de dados de entrada – que neste trabalho serão tratados como *inputs* – e tratá-los segundo um fluxograma, que será apresentado posteriormente no capítulo 4, e os interesses do usuário, objetivando compor, de maneira automatizada, o projeto do poço fornecendo as seguintes informações:

- i) Cálculo das geopressões ao longo do poço;
- ii) Dimensionamento de cada revestimento, a partir de um grau de aço pré-especificado;
- iii) Dimensionamento do BOP que melhor atenderá o poço em questão;

- iv) Profundidade de assentamento das sapatas de cada fase, segundo um critério de assentamento adotado pelo usuário;
- v) Peso de lama, volume de cimento e quantidade de baritina necessária para aumento de densidade em cada fase;
- vi) *Bit record* de cada broca utilizada na perfuração;
- vii) *ROP* estimado ao longo de todo poço;
- viii) Tempo de perfuração discretizado entre: tempo total de manobra, tempo total de revestimento, tempo total produtivo e tempo de deslocamento;
- ix) Diagrama de custos acumulados ao longo da perfuração do poço.

Já o segundo módulo permitirá ao usuário o criar um cenário de perfuração do mesmo poço a partir da escolha de fluidos, brocas e tubos, e avaliar, ao longo dos intervalos de tempo pré-definidos, como variam os seguintes parâmetros relevantes à perfuração:

- i) Pressão de fundo de poço (*BHP*);
- ii) Pressão de bombeio (ΔP_b);
- iii) Taxa de penetração (*ROP*);
- iv) Volume de lama no poço (V_m);
- v) Volume de lama nos tanques (V_{pit});
- vi) Peso sobre a broca (*WOB*);
- vii) Peso sobre o gancho (*WOH*);
- viii) Densidade equivalente de circulação (*ECD*);
- ix) Desgaste da broca (*h*);

Além do acompanhamento em tempo real desses o usuário terá a flexibilidade para alterar quaisquer dos parâmetros de entrada (*inputs*) e assim observar o comportamento dos parâmetros de saída listados acima (*outputs*), bem como acionar processos como troca de broca e assentamento de revestimento. Esse módulo também permitirá ao usuário que este veja o diagrama tempo-profundidade do poço simulado em qualquer tempo desejado, podendo comparar os resultados do cenário proposto aos resultados do projeto de poço. Da mesma forma será disponibilizado ao usuário um gráfico das geopressões e das densidades estática e dinâmica da lama utilizada, de forma que esse possa garantir a segurança da perfuração, evitando influxos ou perdas.

1.3. Estruturação do Trabalho

O trabalho em questão tem como foco apresentar, no enfoque da engenharia de petróleo, as abordagens desenvolvidas e modelos utilizados ao longo do simulador, em detrimento a programação, algoritmos e rotinas, propriamente ditos, que também compõe parte fundamental desse trabalho. Assim sendo este será dividido em sete capítulos distribuídos da seguinte forma:

- i) Capítulo 1: aborda a necessidade de se desenvolver simuladores de perfuração, focando na necessidade econômica que motiva a otimização da perfuração, bem como a aplicação dos simuladores com esse objetivo;
- ii) Capítulo 2: descreve a abordagem teórica por trás do módulo de planejamento da perfuração (atividade estática), que abrange toda a teoria de geomecânica para estimações das geopressões, a teoria de mecânica dos sólidos para dimensionamento dos revestimentos, os diferentes critérios de assentamento das sapatas, dimensionamento do BOP, etc.;
- iii) Capítulo 3: descreve a abordagem teórica por trás do módulo de perfuração em tempo real (atividade dinâmica), que abrange a teoria de brocas tricônicas, que rege o desgaste e a taxa de penetração ao longo da perfuração. Além disso, será apresentado o modelo de hidráulica utilizado, para três tipos distintos de fluido: newtonianos, binghamianos e de lei de potências;
- iv) Capítulo 4: descreve toda a metodologia do módulo de planejamento de poço, o fluxograma adotado, como o usuário pode influenciar na atuação do simulador, as limitações do mesmo, além da base necessária para que um terceiro utilize a ferramenta
- v) Capítulo 5: descreve toda a metodologia abordada no módulo de simulação em tempo real, apresentando os fluxos de trabalho adotados, os modelos considerados, as simplificações assumidas, a forma de atuação do usuário de maneira dinâmica sobre a perfuração, entre outras;
- vi) Capítulo 6: apresenta um estudo de caso, onde o projeto e a simulação de um poço baseado em dados sintéticos serão aplicados desde o início, avaliando estratégias

distintas de perfuração e discussão de como essas estratégias afetariam o desempenho daquela perfuração;

- vii) Capítulo 7: conterà as conclusões do trabalho, bem como as recomendações para trabalhos futuros, visando o aperfeiçoamento do simulador desenvolvido.

2. Abordagem Teórica do Módulo de Planejamento da Perfuração

2.1. Introdução

Como dito anteriormente o simulador supracitado será composto de dois módulos: um de planejamento de perfuração e outro de perfuração em tempo real. Será utilizada a denominação Planejamento de Poço para o conjunto de práticas adotadas previamente a perfuração propriamente dita, que fornecerá o detalhamento necessário para avaliação técnica e econômica da mesma, bem como as diretrizes e informações a serem utilizadas no processo decisório de compra de equipamentos, de logística e etc.

Nesse sentido o conjunto de técnicas, conceitos, correlações e modelos que regem os fenômenos que visamos caracterizar nessa fase e que foram implementados no simulador de perfuração em questão, serão descritos no decorrer deste capítulo, tais como:

- i. Projeto das geopressões
 - a. Densidade das formações;
 - b. Gradiente de sobrecarga;
 - c. Gradiente de poros;
 - d. Gradiente de fratura

- ii. Projeto dos revestimentos;
 - a. Dimensionamento quanto à resistência a carga axial;
 - b. Dimensionamento quanto à resistência a pressão interna;
 - c. Dimensionamento quanto à resistência a colapso;

- iii. Projeto de Equipamentos de segurança de poço - BOP;
 - a. Dimensionamento baseado na pressão de poros;
 - b. Dimensionamento considerando efeito da lâmina d'água;
 - c. Dimensionamento baseado no limite de fraturamento da última sapata;

- d. Dimensionamento baseado na resistência do revestimento
-
- iv. Projeto de Poço;
 - a. Critérios baseado apenas na janela operacional;
 - b. Critério baseado na margem de riser;
 - c. Projetos típicos

É importante observarmos que, apesar dos modelos de estimação do *ROP* e de desgaste de broca terem sido aplicados neste módulo, estes não serão abordando nesse capítulo. Os modelos teóricos que regem tais fenômenos serão abordados apenas no capítulo seguinte, que visa à caracterização teórica do módulo de simulação em tempo real.

2.2. Projeto das Geopressões

Segundo Fjaer *et al.* (2008) geopressões podem ser entendidas como toda e qualquer pressão ou tensão exercida sobre uma formação geológica, seja em escala macro ou microscópica. Dessa forma podemos caracterizar como geopressões conceitos relevantes para o desenvolvimento de qualquer projeto de poço como pressão de poros, tensão de sobrecarga, tensões horizontais totais, tensões efetivas, pressão de fratura, entre outros.

Para o bom desenvolvimento de um estudo geomecânico voltado para o projeto de um poço é necessário um conjunto grande de dados, compostos por dados sísmicos, perfis de poços de correlação, registros de perfuração e perfil composto de poços de correlação, estudo geológico-estratigráfico aprofundado da região (como a caracterização das falhas presentes, das litologias etc.) entre diversos outros.

Usualmente tais estudos são desenvolvidos por equipes inteiras, que avaliam mecanicamente o poço, levantando propriedades das rochas, direções e magnitudes de tensões verticais e horizontais ao longo da trajetória do poço, modelo de geopressões calibrados com análise de estabilidade da parede do poço e, eventualmente, uma análise tridimensional do modelo, aplicando métodos numéricos para avaliar com maior precisão fatores inerentes a perfuração, como influxos e perdas. Tal análise é consideravelmente

complexa e subjetiva, de forma que o presente trabalho aplicará uma proposta simplificada do mesmo.

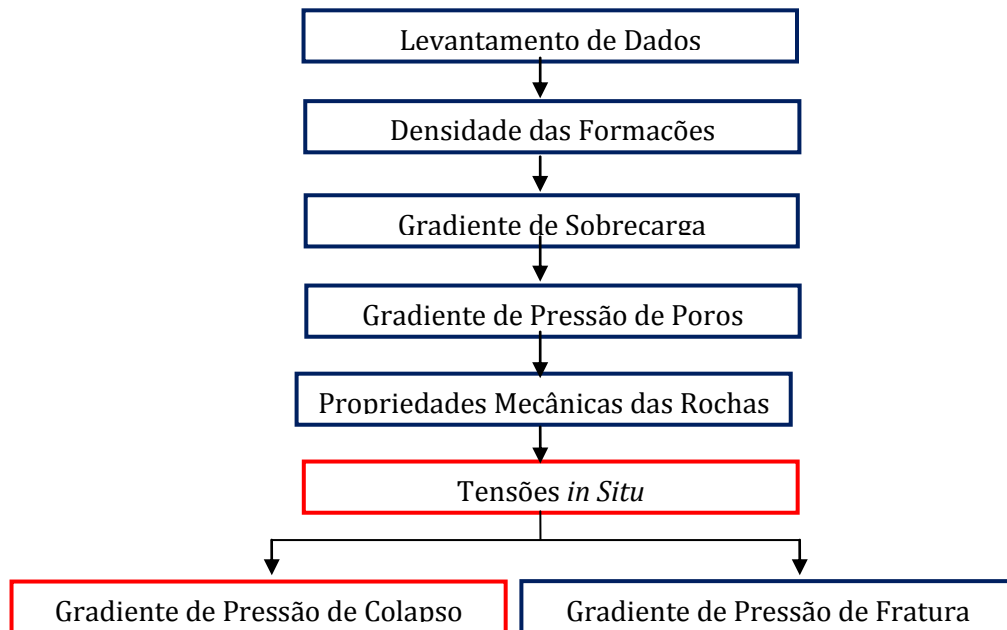


Figura 4 – Fluxo de trabalho para cálculo das geopressões ao longo do poço (Rocha *et al.*, 2008, adaptado)

Podemos observar que os quadros em azul da Figura 4 representam os procedimentos implementados no simulador, enquanto que os quadros em vermelho representam conceitos que não foram abordados no simulador neste momento.

Sendo assim a janela operacional, definida por Rocha *et al.* (2009) como sendo a variação permissível para a pressão exercida pelo fluido de perfuração dentro do poço como forma a manter a integridade desde, não levará em conta a pressão de colapso. Em geral a janela operacional é limitada por valores dos gradientes de poros ou de colapso, como limitante inferior, e pelo gradiente de fratura, como limitante superior.

A pressão de colapso representa a pressão abaixo da qual ocorrerá falha por cisalhamento na formação, levando a redução do diâmetro do poço ou ao desmoronamento do mesmo. Tais eventos podem levar ao aprisionamento de ferramentas, formação de *breakouts*, entre outras.

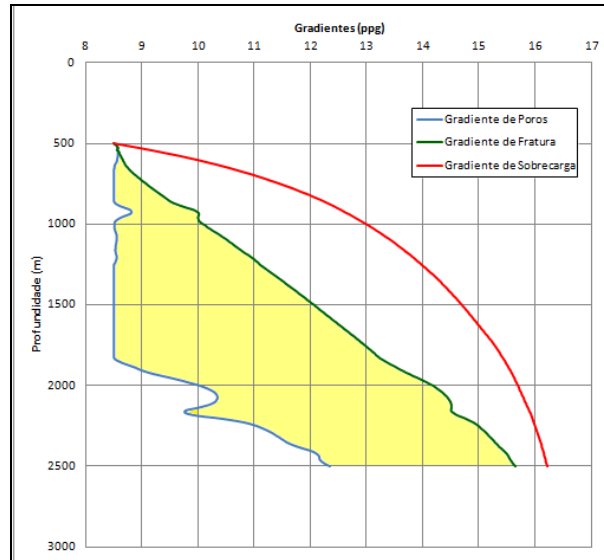


Figura 5 – Representação esquemática de uma janela operacional de perfuração (elaboração própria)

Na Figura 5 podemos observar uma representação esquemática de uma janela operacional de perfuração, limitada apenas pelo gradiente de poros e pelo gradiente de fratura, assim como a janela operacional que será modelada e utilizada pelo simulador, objeto deste trabalho. A janela operacional é definida como sendo um intervalo de pressão de fundo poço que o mesmo possa experimentar sem acarretar em eventos indesejados.

2.2.1. Aquisição de Dados

Atualmente na indústria do petróleo a aquisição de dados, como dados sísmicos, perfis de poço etc., é um fator de grande importância para diversos estudos e análises ao longo da cadeia produtiva, e em especial durante a exploração, devido às grandes incertezas associadas aos processos. É quase intuitivo pensarmos que quanto maior a qualidade e a quantidade dos dados adquiridos maior também será a certeza nas respostas obtidas a partir de tais dados.

A obtenção desses dados, entretanto, normalmente advém de processos longos e caros. Dados sísmicos podem ser obtidos por um navio de aquisição (que apresenta uma cobertura de área maior, apesar de uma menor resolução) ou por ferramenta composta por um transmissor e um receptor de sinal sísmico (que garante maior resolução, apesar de fornecer dados localizados) descida no poço. Dados de resistividade, densidade, raio gama e tantos

outros também são obtidos ao longo da trajetória do poço por uma ferramenta própria, cujo custo, muitas vezes, é evitado em poços marítimos, de custo elevado. Sendo assim o volume de dados obtidos tem relação direta com a qualidade do estudo realizado, entretanto apresenta relação inversa com a viabilidade econômica da campanha exploratória.

Dados de teste de formação e outros são também utilizados e normalmente aplicados como dados para calibração dos modelos de correlação utilizados, como RFTs¹, FPWDs², LOTS³ entre outros.

Usualmente, para poços exploratórios, utilizam-se dados advindos de aquisição sísmica de superfície, que são processados, avaliados e sofrem o processo chamado de inversão sísmica. Como resultado, temos um cubo sísmico de baixa resolução (entre 25 – 100m) que forneceriam dados de *DT* (tempo de retorno da onda sísmica compressional) a serem aplicados para o desenvolvimento de um projeto de poço.

Para poços de desenvolvimento, no entanto, como já existem outros poços para correlação, é possível perfis ao longo da trajetória do novo poço sejam estimados. Perfis como de densidade e resistividade apresentam aplicação em modelos da literatura para estimação de geopressões, com a vantagem de apresentarem maior resolução se comparado com os dados advindos do cubo sísmico previamente citado. Perfis sísmicos também podem ser obtidos utilizando poços de correlação.

2.2.2. Massa Específica das Formações

A massa específica das formações é um parâmetro importante em um projeto de poço uma vez que este constitui a base para as estimações do gradiente de sobrecarga, como veremos na sequência da seção.

¹ Repeated Formation Test: teste para medição da pressão de poros. É realizada através de descida de um equipamento de medição por cabos após a perfuração de uma seção

² Formation Pressure While Drilling: teste para medir a pressão de poros. É realizada por meio de equipamentos descidos na coluna de perfuração que medem a pressão da formação durante a perfuração.

³ Leak-of-Test: teste para medir a pressão de absorção de uma formação aberta. Normalmente realizada após o assentamento de um revestimento.

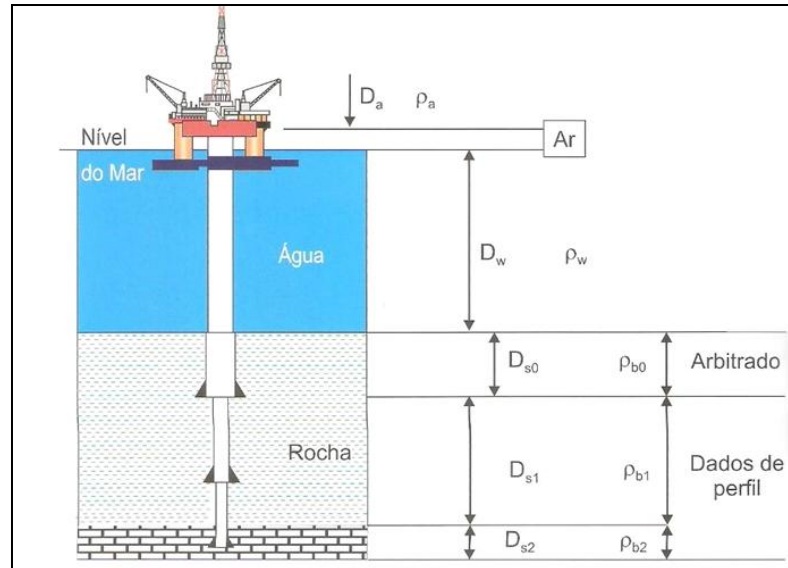


Figura 6 – Discretização dos trechos para determinação das massas específicas (ROCHA *et al.* 2008)

Na Figura 6 podemos observar a discretização das massas específicas que deverão ser obtidas de alguma forma com o objetivo de se calcular a pressão de sobrecarga. A massa específica da água do mar (trecho D_w) deve ser previamente definida. Em geral $\rho_w = 1,03 \text{ g/cm}^3$. No trecho representado por D_{s0} temos a massa específica de um trecho superficial no qual, usualmente, não temos se tem dados de perfis, nem de densidade tampouco de qualquer outro que possamos utilizar como base para correlações. Nesse trecho aplicam-se outras correlações da literatura que calibram a massa específica desse trecho superficial com a das formações imediatamente abaixo, como forma a apresentar um perfil calibrado e aproximadamente contínuo ao longo de toda a trajetória do poço.

Para o último trecho ($D_{s1} + D_{s2}$) pode-se obter a massa específica das formações por meio de medições *in situ* (perfis) ou através de outras correlações matemáticas, como Gardner (1958), utilizando-se outros parâmetros como base de cálculo, como o próprio DT .

2.2.2.1. Perfil de Massa Específica

É utilizado com o objetivo de se obter diretamente essas propriedades ao longo da trajetória do poço, como uma elevada resolução. Tal forma de aquisição de dados apresenta a vantagem de ser obtida diretamente, o que minimiza o erro inerente a calibração do modelo

matemático. Os dados diretos podem ser aplicados como dados para calibração dos modelos ou mesmo aplicados como resultados.

2.2.2.2. Método de Gardner

Gardner *et al.* (1958) propuseram uma das correlações para determinação da massa específica de formação mais aplicada na indústria de petróleo até os dias de hoje devido a sua simplicidade e confiabilidade (Rocha *et al.* 2008). Assim como a correlação proposta por Belloti *et al.* (1962), outra correlação usualmente aplicada na indústria, esta faz uso de dados de DT , entretanto correlação de Gardner não apresenta distinção entre litologias, como ocorre com a correlação de Belloti. A qualidade dos resultados da mesma só depende de uma boa calibração.

$$\rho_r = aV_p^b = a\left(\frac{10^6}{DT}\right)^b \quad (1)$$

Nessa formulação temos que:

- ρ_r massa específica da formação, em g/cm^3 ;
- V_p velocidade compressional, em $ft/\mu s$;
- DT tempo de trânsito compressional, em $\mu s/ft$;
- a constante empírica (usualmente igual a 0,23 para Golfo do México, passível de calibração com o cenário em questão);
- b expoente empírico (usualmente igual a 0,25 para o Golfo do México, passível de calibração com o cenário em questão).

A Figura 7 apresenta o ajuste do perfil de densidade a partir da alteração dos parâmetros empíricos a e b .

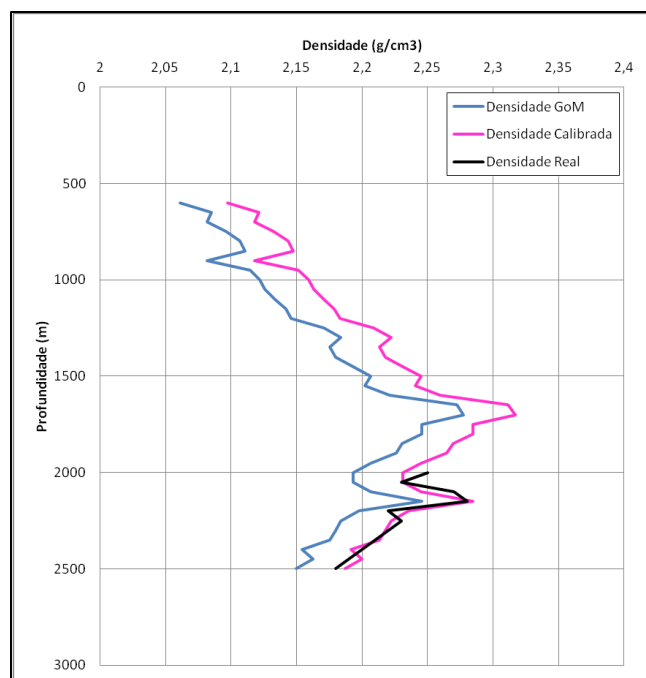


Figura 7 – Análise de sensibilidade do método de Gardner frente a modificações nos parâmetros empíricos a e b . (elaboração própria)

2.2.2.3. Correlação de Extrapolação da Massa Específica

Como já foi dito anteriormente nas zonas mais rasas do poço normalmente há uma ausência de dados que impossibilita a estimativa da densidade dessas formações por métodos diretos. Dessa forma correlações empíricas devem ser aplicadas de forma a garantir uma boa concordância e continuidade entre a densidade das formações acima e abaixo do limite de aquisição de dados (profundidade $D_w + D_{s0}$, considerando a Figura 6 como referência).

Nesse sentido modelos matemáticos foram propostos. Correlações como as de Miller e de Bourgoyne se utilizam de informações e perfis de porosidade ao longo da trajetória do poço. Entretanto, como neste trabalho visamos o planejamento de poço de forma automatizada e sem que exija um grande volume de dados a serem processados, tais correlações não serão implementadas. Utilizaremos então uma correlação exponencial da massa específica em função da profundidade dos sedimentos, proposta por Traugott (1997) *apud* Fjaer (2008), que ajusta bem o perfil de densidade em formações mais rasas, além de ser mais simples que as demais.

$$\rho_r = a(Z_s)^b + c \quad (2)$$

Nessa formulação temos que:

- Z_s profundidade dos sedimentos, em m ;
- a constante empírica que varia com o perfil;
- b constante empírica que varia com o perfil;
- c constante empírica que deve representar a massa específica das formações no leito marinho, sendo, normalmente, igual $1,8 \text{ g/cm}^3$.

Podemos observar que nessa equação os parâmetros a , b e c correspondem a coeficientes de calibração que deverão garantir que a massa específica estimada seja coerente com o modelo já calculado.

Podemos observar nos gráficos apresentados na Figura 8 a boa concordância entre os resultados estimados pelo modelo de extrapolação analisado. Podemos notar que a continuidade entre os valores previstos e aqueles calibrados pelo método de Gardner é satisfeita, corroborando os bons resultados fornecidos por esse modelo, dada uma boa calibração com os parâmetros a , b e c .

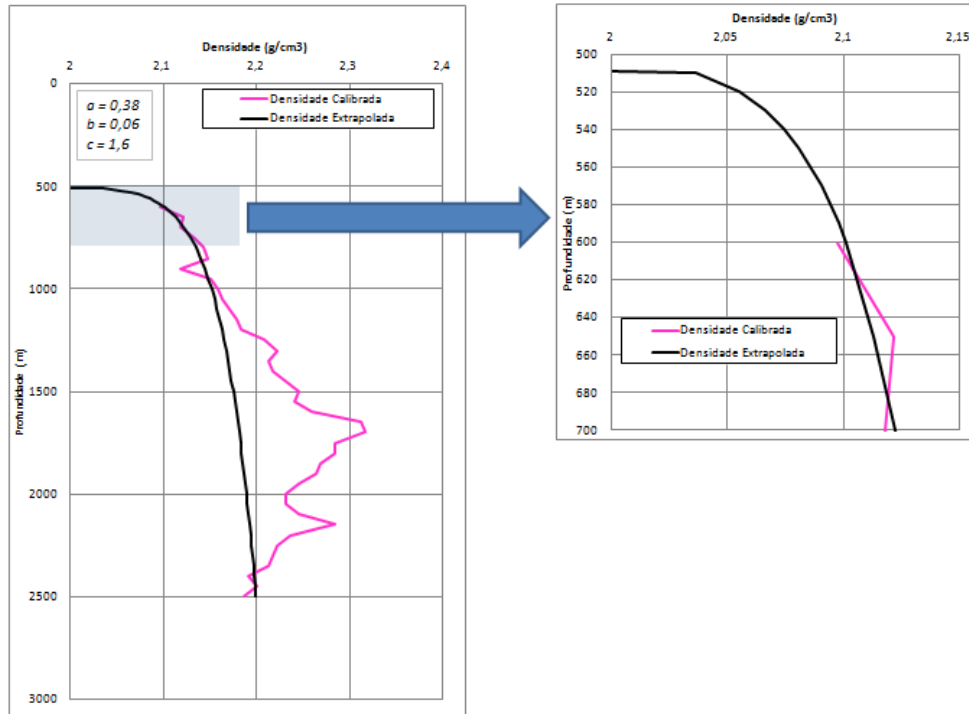


Figura 8 – Aplicação da correlação de extrapolação das densidades para profundidades rasas (elaboração própria)

2.2.3. Tensão e Gradiente de Sobrecarga

A tensão de sobrecarga corresponde à tensão vertical exercida sobre um determinado elemento de volume a uma profundidade arbitrária Z . Simplificadamente esse representa o peso das camadas de rocha sobrepostas a este elemento de volume. A maneira formal para calcularmos a tensão de sobrecarga é dada por:

$$\sigma_{ov} = \int_0^Z \rho(Z)gdZ \quad (3)$$

Nessa formulação temos que:

- σ_{ov} tensão de sobrecarga, em Pa , no Sistema Internacional de Unidades (SI);
- $\rho = \rho(Z)$ massa específica da formação em função da profundidade, em kg/m^3 , no Sistema Internacional de Unidades;

- Z profundidade vertical a partir da mesa rotativa, desprezando o air gap;
- g aceleração da gravidade, aproximadamente $9,8 \text{ m/s}^2$, no Sistema Internacional de Unidades.

Entretanto, como pudemos ver na seção 2.2.2, a determinação da massa específica das formações não é feita de forma contínua, mas o que se tem é uma sucessão de dados calculados para diversos pontos ao longo da trajetória do poço de maneira discreta. Sendo assim a aplicação da expressão formal (Eq. 3) não se faz coerente, levando a necessidade de uma expressão que descreva a tensão de sobrecarga ao longo da trajetória do poço utilizando um perfil discreto de densidade das formações. Teremos então que a tensão de sobrecarga também poderá ser dada pelo somatório das densidades discretizadas, como vemos abaixo.

$$\sigma_{ov} = \alpha \left(\rho_w LDA + \sum_0^n \rho_{r_i} \Delta Z_i \right) \quad (4)$$

Nessa formulação temos que:

- σ_{ov} tensão de sobrecarga, em *psi*;
- LDA lâmina de água, em *m*;
- ρ_{r_i} massa específica da formação em uma determinada profundidade, em g/cm^3 ;
- ΔZ_i diferença de profundidade entre a densidade ρ_{r_i} e a densidade $\rho_{r_{i-1}}$, em *m*;
- α fator de conversão de unidades, igual a 1,422 para as unidades citadas.

A partir da estimação da tensão de sobrecarga podemos calcular o conceito de gradiente de sobrecarga, que nada mais é do que derivada da tensão com a profundidade. Simplificadamente, como trabalharemos com resultados discretos, temos que:

$$G_{ov} = \frac{\sigma_{ov}}{\alpha Z} \quad (5)$$

Nessa formulação temos que:

- G_{ov} gradiente de sobrecarga, em *ppg*;
- α fator de conversão de unidades, igual a 0,1704 para as unidades utilizadas aqui

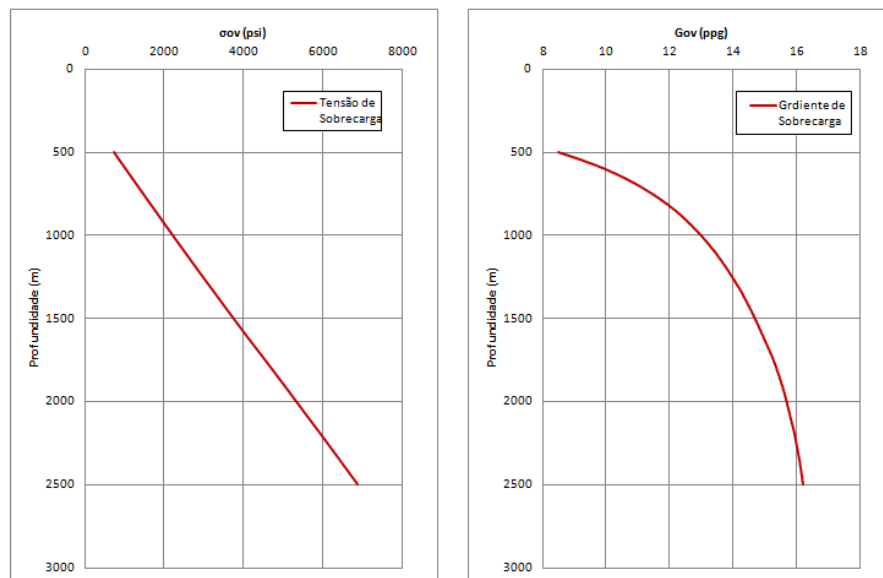


Figura 9 – Comportamento típico de tensão e de gradiente de sobrecarga ao longo do poço (elaboração própria)

2.2.4. Gradiente de Pressão de Poros

Segundo Rocha *et al.* (2008), pressão de poros pode ser definida como a pressão na qual o fluido que preenche o volume poroso da rocha está confinado. Esta apresenta papel fundamental no desenvolvimento de um projeto de poço, uma vez que eventos indesejados como *kicks* (influxos de gás, água, óleo, etc, que avançam para dentro do poço devido a um diferencial de pressão negativo no fundo) são governados basicamente pelo balanço de pressão entre a pressão no fundo do poço e pressão de poros. Tais *kicks* podem, eventualmente, atingir a plataforma como um grande de volume de gás, levando a um cenário extremo de *blow-out*, como podemos ver na Figura 10.

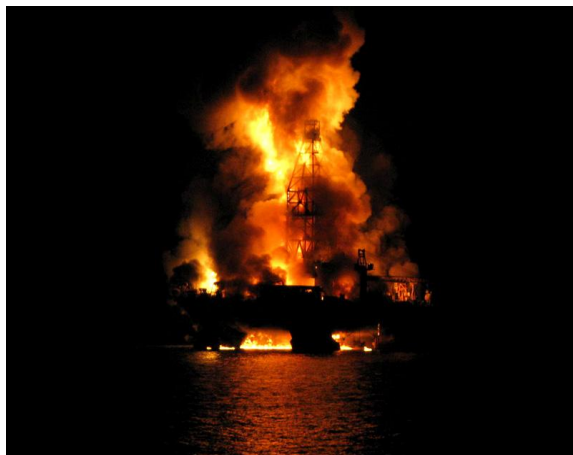


Figura 10 – Imagem de uma plataforma de petróleo após sofrer *blowout* (Frydman, 2010)

Entretanto é válido ter em mente que preocupação com influxos só fazem sentido quando estamos perfurando formações permeáveis e com chances de presença de hidrocarbonetos. Caso contrário mesmo havendo um diferencial de pressão entre a formação e o poço não haverá condições geológicas necessárias para ocorrer o influxo.

Um problema potencial que pode acontecer durante a perfuração devido ao diferencial de pressão baixo entre formações permeáveis e a do interior do poço é o aprisionamento da coluna. Este ocorre quando o diferencial de pressão se inverte, ou seja, a pressão no interior do poço se torna muito maior que a pressão de poros da formação. A força necessária para se movimentar a coluna deve ser capaz de vencer esse diferencial, portanto, nesse caso, teríamos um acréscimo muito acentuado nessa força, acarretando no aprisionamento da coluna.

Dessa forma é fundamental uma boa estimativa da pressão de poros para garantir a segurança da operação e otimizar a perfuração, minimizando a ocorrência de eventos indesejáveis e da necessidade de procedimentos corretivos. Assim minimiza-se o tempo não produtivo (*NPT*) e, conseqüentemente, o custo total da atividade. Entretanto tais estimações nem sempre são triviais. Segundo Pastor (2010) o gradiente de poros é o parâmetro de maior incerteza dentro de um estudo geomecânico, sendo classificado pelo mesmo como um mundo a parte dentro da mecânica de rochas, tamanha a dificuldade de estimações precisas. Assim

sendo aplicações acuradas dos métodos e uma boa calibração são fundamentais para resultados coerentes e com incerteza reduzida.

2.2.4.1. Classificação dos Gradientes de Poros

Segundo Rocha *et al.* (2008), a pressão de poros é dita normal para uma dada profundidade quando seu valor é igual à pressão que seria exercida por uma coluna hidrostática de fluido da formação. Isso acontece quando há um equilíbrio entre o aumento na pressão de compactação, redução do espaço poroso e a conseqüente expulsão da água que saturava a rocha. Qualquer fenômeno físico ocorrido durante a compactação daquela rocha que viesse a alterar esse equilíbrio levaria a pressão de poros e um estado dito “anormal”, ou seja, diferente a pressão hidrostática do fluido presente na formação.

Sendo assim os gradientes de poros podem ser classificados em quatro diferentes grupos, segundo a Tabela 1.

Perfil de Pressão de Poros	Limites Definidos
Anormalmente Baixa	$G_{pr} < 8,5 \text{ ppg}$
Normal	$8,5 \text{ ppg} < G_{pr} < 9,0 \text{ ppg}$
Anormalmente Alta (Sobrepessão)	$9,1 \text{ ppg} < G_{pr} < 90\% \text{ Gov}$
Alta Sobrepessão	$G_{pr} > 90\% \text{ Gov}$

Tabela 1 – Classificação dos Gradientes da Pressão de Poros, (Rocha *et al.* 2008, adaptado).

Para o planejamento de poços devemos ter cuidado na estimação das pressões de poros principalmente para zonas de sobrepressão e alta sobrepressão. A perfuração dessas zonas eleva o risco de ocorrência de influxos, que podem evoluir para um *blowout*. Zonas de pressão anormalmente baixas (reservatórios depletados por exemplo) são candidatas a ocorrência de aprisionamento de coluna, devido ao diferencial de pressão negativo entre pressão da formação e a pressão do fundo do poço. Na Figura 11 observamos uma zona de pressão de poros anormalmente alta.

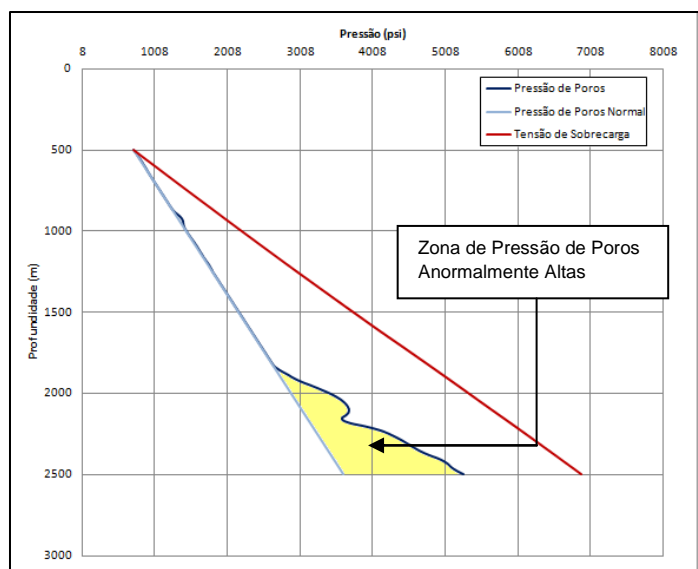


Figura 11 – Representação esquemática ilustrando zonas de pressão de poros anormalmente altas (elaboração própria)

Para uma boa estimativa do gradiente de poros é necessário, entretanto, que se saibam as origens da sobrepressão. Como veremos a seguir os modelos disponíveis na literatura para previsão da pressão de poros são, em sua grande maioria, de aplicação restrita apenas para determinadas causas de sobrepressão. Dessa forma é importante que tais conceitos estejam bem sedimentados e uma investigação seja realizada, como forma de avaliar tal causa antes que o modelo geomecânico seja desenvolvido.

2.2.4.2. Origem das Sobrepressões

Segundo Bowers (1995) existem duas causas principais que geram zonas de pressão de poros anormalmente altas: subcompactação e expansão de fluidos. Efeitos como de diferença de densidade e transferência de pressão também poderão levar a zonas de sobrepressão. No entanto tais efeitos não serão abordados no escopo desse trabalho.

A) Mecanismo de Subcompactação

Segundo Fjaer *et al.* (2008) a subcompactação é o causa mais comum de pressões de poros anormalmente altas nas bacias mundiais. Esta ocorre quando há um desequilíbrio entre o acréscimo da pressão de compactação e a expulsão dos fluidos contidos no espaço poroso

da formação devido à baixa permeabilidade da mesma, característica típica de rochas como folhelhos e argilas. Nesse caso a sobrepressão é resultado do trapeamento do fluido nessas formações, ou mesmo em camadas finas de arenitos permeáveis, de forma a sustentar parte da pressão de sobrecarga. Portanto pode-se observar que, nesse caso, a pressão de poros é função da tensão de sobrecarga e da tensão efetiva.

B) Mecanismos de Expansão de Fluidos

Segundo Bowers a expansão de fluidos na bacia ocorre devido à ocorrência de determinados fenômenos físicos, tais como fluxo térmico proveniente de tectonismo e outras fontes, maturação de hidrocarbonetos, diagênese entre outros. Nesse caso a sobrepressão é oriunda do confinamento imposto pela matriz da rocha quando o fluido contido no seu espaço poroso – que não tem seu volume alterado – tenta expandir-se. Ao contrário do caso anterior mecanismos de expansão de fluidos podem levar a um acréscimo na pressão de poros a uma taxa superior ao acréscimo da tensão de sobrecarga, não sendo, portanto, função da mesma.

2.2.4.3. Métodos de Estimação da Pressão de Poros

Na literatura existem diversos métodos para estimação de pressão de poros, cada um com sua respectiva janela de aplicação e restrições. Alguns métodos fornecem resultados coerentes quando a causa da sobrepressão é o efeito de subcompactação, entretanto são completamente falhos quando temos efeito de expansão de fluidos. Portanto é importante um bom diagnóstico das origens das sobrepressões objetivando uma estimação correta da pressão de poros.

O primeiro método proposto para estimação da pressão de poros foi de Hottman e Johnson (1965) *apud* Bowers (1998), e desde então a literatura sobre o tema só aumentou. Basicamente todos os métodos em questão se baseiam na premissa de que a pressão de poros tem influência sobre as propriedades dos folhelhos que são dependentes de compactação, como porosidade, densidade, resistividade e velocidade sônica. A partir daí tais métodos foram divididos dois grandes grupos: métodos diretos e métodos de tensão efetiva (Bowers, 1998).

A) Métodos Diretos

Os métodos de estimação direta fundamentam-se, basicamente, em quanto o indicador da pressão de poros – perfil de porosidade, velocidade sônica etc. – divergem da curva de tendência normal. Existem dois desses métodos na literatura: Hottman e Johnson (1965) e Pennebaker (1968) *apud* Bowers (1998).

B) Métodos da Tensão Efetiva

Já os métodos da tensão efetiva são baseados na 1ª Lei dos Esforços Efetivos, proposta por Terzaghi (1943) *apud* Fjaer *et al.* (2008), que afirma que compactação das rochas é controlada pela diferença entre a tensão confinante total e a pressão de poros. A essa diferença dá-se o nome de tensão efetiva.

De uma forma global todos os métodos de estimação baseados na lei de Terzaghi seguem o seguinte padrão:

$$P_{pr} = \sigma_{ov} - \sigma' \quad (6)$$

A diferença em cada técnica esta apenas na forma de estimação da tensão efetiva σ' , a partir dos indicadores da pressão de poros.

Diversos métodos já foram propostos para se estimar a pressão de poros baseado na primeira lei dos esforços efetivos de Terzaghi. Entre elas podemos citar o método de Foster e Whalen (1966), Lane e Macpherson (1976), Belloti e Giacca (1978) e os mais comumente aplicados na indústria: Eaton (1975) e Bowers (1995), entre vários outros. Nesse trabalho abordaremos apenas os dois últimos devido sua maior aplicabilidade nos projetos geomecânicos para perfuração.

i) Método de Eaton

Eaton (1975) postulou que a pressão de poros em argilas poderia ser estimada a partir da Lei de Terzaghi, onde a tensão efetiva seria modelada através de perfis sônicos ou de resistividade, dada por:

$$\sigma' = \sigma'_n \left(\frac{\lambda}{\lambda_n} \right)^\varphi \quad (7)$$

Nessa formulação temos que:

- σ' tensão efetiva sobre a formação;
- σ'_n tensão efetiva para estado de compactação normal;
- λ indicador da pressão de poros efetivamente observado: velocidade sísmica ou resistividade;
- λ_n indicador da pressão de poros para o estado normal de compactação: velocidade sísmica ou resistividade;
- φ coeficiente de Eaton, que varia para diferentes indicadores e diferentes regiões.

Como dito anteriormente o indicador para estimação da pressão de poros pode ser registros de porosidade, densidade, resistividade ou velocidade sônica. Para cada indicador o modelo é o mesmo, só devendo ser ajustado o parâmetro φ , conhecido como coeficiente de Eaton. Na tabela a seguir podemos observar as faixas de φ usualmente aplicados na indústria para diferentes indicadores.

Indicador da Pressão de Poros	φ
Resistividade	1 - 1,5
Velocidade Sônica	2 - 3

Tabela 2 – Valores típicos do coeficiente de Eaton

Já os parâmetros σ'_n e λ_n correspondem a valores esperados quando o estado de compactação normal for observado. Portanto estes devem ser obtidos a partir de uma curva de tendência da compactação normal. Essa curva normalmente é considerada como sendo uma reta em escala loglog, a partir da qual teremos uma melhor calibração da pressão de

poros estimada com a observada por medições diretas, a partir de dados de RFTs e/ou FPWD. Na Figura 12 podemos observar um perfil de tempo de trânsito (DT) e a curva de tendência a compactação normal.

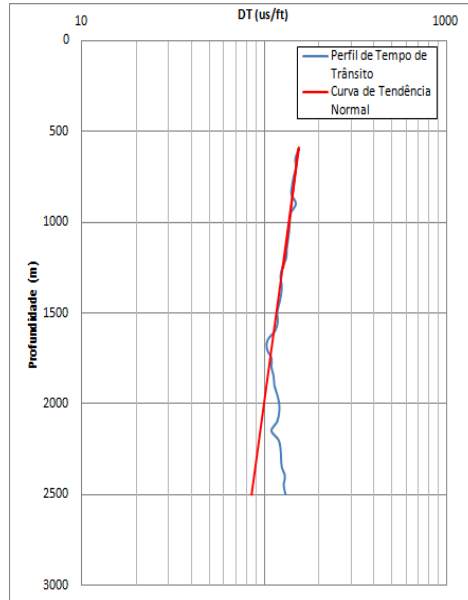


Figura 12 – Perfil de tempo de trânsito ao longo do poço e sua respectiva curva de tendência normal (elaboração própria)

O coeficiente angular da curva de tendência semilogarítmica (m), representada acima, será obtido pela relação abaixo, onde os índices i e j representam pontos quaisquer dentro da zona de compactação normal, que pertencerão a curva de tendência.

$$m = \frac{\log \lambda_i - \log \lambda_j}{Z_i - Z_j} \quad (8)$$

A partir daí poderemos estimar o indicador para cenário de compactação normal em qualquer profundidade, a partir de inversa:

$$\lambda_n = \lambda_i 10^{m(Z-Z_i)} \quad (9)$$

Como, para um cenário de compactação normal, a pressão de poros será igual à hidrostática de uma coluna de fluido da formação podemos representar a tensão efetiva normal por:

$$\sigma'_n = \sigma_{ov} - P_{pn} = \sigma_{ov} - \rho_w g Z \quad (10)$$

Sendo assim podemos enfim estimar o gradiente de poros ao longo da trajetória do poço pela seguinte expressão:

$$G_{pr} = G_{ov} - (G_{ov} - G_n) \left(\frac{\lambda}{\lambda_n} \right)^\gamma \quad (11)$$

Esse método fornece uma ferramenta útil para estimação da pressão e do gradiente de poros, em particular devido a sua simplicidade e facilidade de implementação, além de fornecer resultados com razoável precisão quando bem calibrado. Entretanto, de acordo com Bowers (1995) o modelo de Eaton considera que a formação se comporta de maneira perfeitamente elástica e, portanto, sua aplicação só seria razoável quando a sobrepressão fosse causada por mecanismos de subcompactação. Em casos de mecanismos de expansão de fluidos teríamos uma subestimação de tais valores ao aplicar o método de Eaton. Para solucionar tal questão Bowers propôs o modelo a seguir.

ii) *Método de Bowers*

Com o objetivo de desenvolver um método para estimativa da pressão de poros que não fosse limitado a origem por subcompactação, Bowers (1995) desenvolveu um modelo sensível a origem do efeito de sobrepressão, de forma que zonas anormalmente pressurizadas devido a atuação de mecanismos de expansão de fluido possam ser modeladas corretamente, fato que não ocorria com a aplicação do método de Eaton.

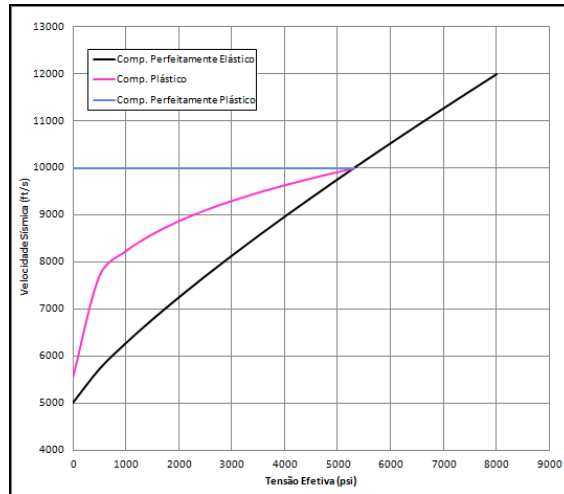


Figura 13 – Perfil de velocidade sísmica para zonas de comportamento perfeitamente elástico, plástico e perfeitamente plástico (elaboração própria)

Segundo Fjaer *et al.* (2008) a origem da sobrepressão pode ser observada em um gráfico como o apresentado na Figura 13. Quando o comportamento da velocidade sísmica apresenta um perfil próximo a da curva em preto, representativa de uma rocha com comportamento perfeitamente elástico podemos afirmar que a sobrepressão teria origem na subcompactação. A essa curva dá-se o nome de curva virgem. O perfil de tensão efetiva do modelo de Eaton se aproxima da tendência apresentada por essa curva.

Já quando o perfil da velocidade sísmica diverge dessa curva, se aproximando da curva em azul, representativa de um perfil perfeitamente plástico, temos um comportamento típico de sobrepressão devido à expansão de fluidos. A essa curva (em magenta) dá-se o nome de curva de descarregamento.

Sendo assim o método de Bowers visa à aplicação das relações que regem as tendências das curvas citadas objetivando modelar zonas de sobrepressão devido a subcompactação e a mecanismos de expansão de fluidos, utilizando como indicador apenas a velocidade sísmica.

Para zonas de compactação normal e aquelas cuja sobrepressão segue a curva virgem (efeito da subcompactação) o método prevê que tensão efetiva será dada por:

$$\sigma' = \left(\frac{V_p - V_{p_0}}{A} \right)^{1/B} \quad (12)$$

Nessa formulação temos que:

- V_{p_0} parâmetro de calibração que varia entre 4000 e 6000 *ft/s*. Vale 5000 *ft/s* no Golfo do México;
- A parâmetro de calibração que varia entre 1 e 20 normalmente. Vale 4,465 no Golfo do México;
- B parâmetro de calibração que varia entre 0 e 1. Vale 0,8168 no Golfo do México (Bowers, 1995);

Já para zonas de compactação anormalmente altas, onde a sobrepressão é oriunda de mecanismos de expansão de fluidos, foi *proposto* que a tensão efetiva deverá ser dada por:

$$\sigma' = \sigma_{max} \left[\frac{1}{\sigma_{max}} \left(\frac{V_p - V_{p_0}}{A} \right)^{1/B} \right]^U \quad (13)$$

$$\sigma_{max} = \left(\frac{V_{p_{max}} - V_{p_0}}{A} \right)^{1/B} \quad (14)$$

$$V_{p_{max}} = 10^6 / DT_{min} \quad (15)$$

Nessas formulações temos que:

- $V_{p_{max}}$ máxima velocidade sísmica observada no perfil sísmico utilizado, em *ft/s*;
- σ_{max} máxima tensão efetiva observada, que corresponde ao ponto máximo da curva de descarregamento, em *psi*;

- U parâmetro de descarregamento que mede o quão plasticamente a rocha se comporta. Varia normalmente entre 3 e 8, valendo 3,13 no Golfo do México (Bowers, 1995).

Dessa forma a estimação do gradiente de poros aplicando o método de Bowers será dado pelas equações a seguir, onde a primeira equação apresenta estimação para rochas com comportamento perfeitamente elástico e a segunda para rochas com comportamento plástico. Segundo Bowers zonas sobre a curva de descarregamento apresentarão maiores estimativas da pressão de poros.

$$G_{pr} = G_{ov} - (G_{ov} - G_n) \left(\frac{V_p - V_{p_0}}{A} \right)^{\frac{1}{B}} \quad (16)$$

$$G_{pr} = G_{ov} - (G_{ov} - G_n) \left(\frac{V_{p_{max}} - V_{p_0}}{A} \right)^{1/B} \left[\frac{1}{\left(\frac{V_{p_{max}} - V_{p_0}}{A} \right)^{1/B} \left(\frac{V_p - V_{p_0}}{A} \right)^{1/B}} \right]^U \quad (17)$$

Vale observar que, a partir do momento que sabemos que o poço está em uma zona de sobrepressão, a decisão sobre qual das equações devem ser aplicada normalmente é realizada com o auxílio de dados de poços de correlação.

Na Figura 14 podemos observar o impacto do método na pressão de poros estimada, em particular a sobrestimação do gradiente quando consideramos o efeito de mecanismos de expansão de fluido como gerador da sobrepressão.

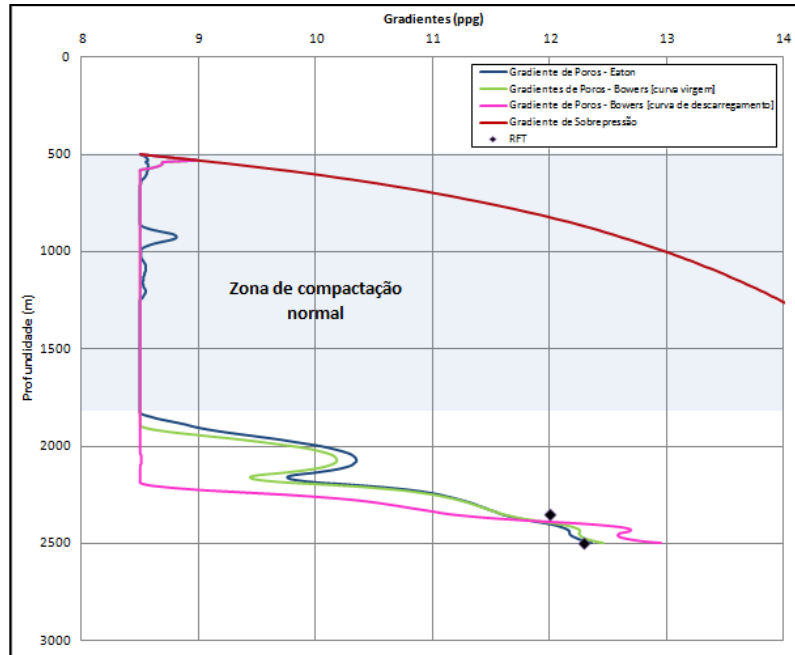


Figura 14 – Comparação entre os resultados dados pelos métodos de Eaton e Bowers (elaboração própria)

2.2.5. Gradiente de Pressão de Fraturas

Segundo Fjaer *et al.* (2008) o fraturamento hidráulico da formação ocorre quando a pressão de poros da rocha excede a tensão horizontal mínima somada a resistência a tração da mesma. Isso equivaleria a dizer que o fraturamento ocorrerá quando a pressão no interior do poço é grande o bastante para alterar o estado de tensões na parede do poço de compressivas para trativas, atingindo a resistência a tração da mesma. Também podemos chamar tais fraturas por *breakdowns* ou fraturas superiores.

A ocorrência de *breakdowns* durante a perfuração gera zonas de escoamento dos fluidos de perfuração para dentro da parede do poço, de forma a ocasionar a dita perda de circulação. Esse se torna um evento de grande risco para perfuração uma vez que a perda de fluido para a formação reduz o seu volume no poço e, conseqüentemente, o seu nível. Sendo assim a pressão no fundo do poço também reduz, podendo atingir níveis inferiores a pressão de poros de formação. Nesse caso poderíamos ter um influxo seja permeável. Portanto a boa estimativa dos gradientes de fratura ao longo da trajetória do poço é importante para um projeto seguro.

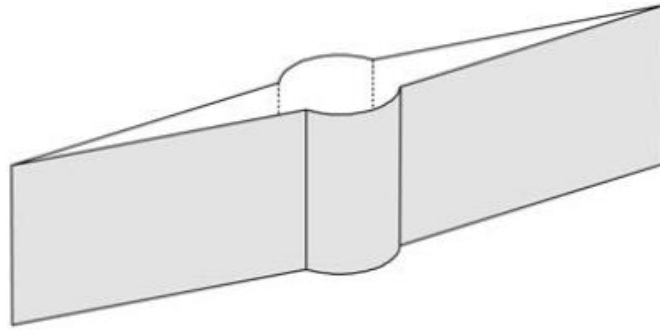


Figura 15 – Fratura no entrono de um poço vertical (Fjaer *et al.*, 2009)

2.2.5.1. Estimativas do Gradiente de Fratura

Segundo Rocha *et al.* (2008), existem basicamente duas formas de estimação da pressão de propagação de fratura ao longo de um poço: através de ferramentas de medição ou de testes de poço que permitem a aquisição direta da pressão de fratura, comumente denominados de métodos diretos; e através de modelos teóricos e calibrações com as medições diretas, denominados métodos indiretos. Nesse ponto é importante diferenciarmos pressão de surgimento de fratura e pressão de propagação de fratura, uma vez que a primeira é função da trajetória do poço, e a segunda não. Nesse contexto estaremos interessados na pressão de propagação de fratura, doravante denominada simplesmente de pressão de fratura.

A) Métodos Diretos

Tipicamente a aquisição direta da pressão de fratura é obtida através de testes de poço onde o pressuriza-se e despressuriza-se o mesmo, seguindo uma sequência controlada, e sensores de pressão no fundo do poço acompanham a resposta do desse poço. Assim, podem-se inferir informações como pressão de fratura, tensão horizontal mínima entre outras. Entre tais testes podemos elencar: Teste de Absorção (LOT), Teste de Absorção Estendido, Teste de Integridade da Formação (FIT) e Teste de Microfraturamento. No entanto já existem algumas ferramentas no mercado que se propõem a medir as tensões da formação *in situ*.

A aplicação de métodos diretos, apesar de reduzir a incerteza nas estimações dos parâmetros, não é capaz de abranger todo poço. Para tal devemos lançar mão dos métodos indiretos, utilizando os dados obtidos pelos métodos diretos apenas para a melhor calibração dos nossos modelos.

B) Métodos Indiretos

Como dito acima os métodos indiretos permitem uma estimativa do gradiente de fratura ao longo da trajetória do poço. Entretanto os modelos teóricos nos quais tais métodos se baseiam dependem de diversas informações que nem sempre estão disponíveis, o que acarreta em diversas simplificações nessa abordagem, levando a incertezas consistentes no modelo. Dentre os diversos métodos para estimação indireta da pressão de fratura podemos citar:

- Método das Tensões Tangenciais (*Hoop Stress*)
- Método da Tensão Mínima
- Correlações Específicas

O primeiro método é baseado na solução de Kirch para tensões tangenciais na parede do poço. Tal solução apresenta originalmente cinco graus de liberdade ($\sigma_H, \sigma_h, T_o, \theta, P_p$).

Já o método da tensão mínima apresenta originalmente apenas três graus de liberdade, portanto este será o método aplicado para estimação do gradiente de fratura ao longo do poço no simulador objeto de estudo.

i) Método da Tensão Mínima

Esse método é usualmente aplicado na indústria do petróleo para a estimação indireta do gradiente de fratura devido sua simplicidade, exigindo menos informação que o método das tensões tangenciais. Esse método assume como hipóteses que a rocha não possui resistência a tração e que o início da propagação da fratura ocorre quando a pressão de fundo atinge a tensão horizontal mínima. Como forma de simplificarmos o modelo para utilizarmos um pacote menor de informações substitui-se a tensão horizontal mínima pela pressão de absorção obtida nos métodos de medição direta nos poços de correlação.

$$P_{wf} = \sigma_h = P_{fr} \quad (18)$$

$$\sigma_h = \sigma_h' + P_{pr} \quad (19)$$

Como existe uma relação proporcional entre a tensão horizontal mínima efetiva e a tensão vertical efetiva, dada pelo coeficiente κ , podemos representar a pressão de fundo através da pressão de poros e da tensão de sobrecarga, ambos os parâmetros já conhecidos no nosso simulador.

$$\kappa = \frac{\sigma_h'}{\sigma_{ov}'} = \frac{\sigma_h - P_{pr}}{\sigma_{ov} - P_{pr}} = \frac{P_{fr} - P_{pr}}{\sigma_{ov} - P_{pr}} \quad (20)$$

$$P_{fr} = \sigma_h = P_{pr} + \kappa (\sigma_{ov} - P_{pr}) \quad (21)$$

O coeficiente κ deverá ser obtido para diferentes profundidades ao longo do poço. Normalmente calcula-se tal coeficiente através uma curva logarítmica com a profundidade onde os coeficientes da curva são calculados por meio de uma regressão linear com as informações dos testes de medição direta, substituindo a tensão horizontal mínima pela pressão de absorção. A curva em questão seria dada por:

$$\kappa = a + b \ln(Z_s) \quad (22)$$

$$a = \frac{\sum \ln(Z_s)^2 \sum \kappa - \sum \ln(Z_s) \kappa \sum \ln(Z_s)}{n \sum \ln(Z_s)^2 - (\sum \ln(Z_s))^2} \quad (23)$$

$$b = \frac{n \sum \ln(Z_s) \kappa - \sum \ln(Z_s) \sum \kappa}{n \sum \ln(Z_s)^2 - (\sum \ln(Z_s))^2} \quad (24)$$

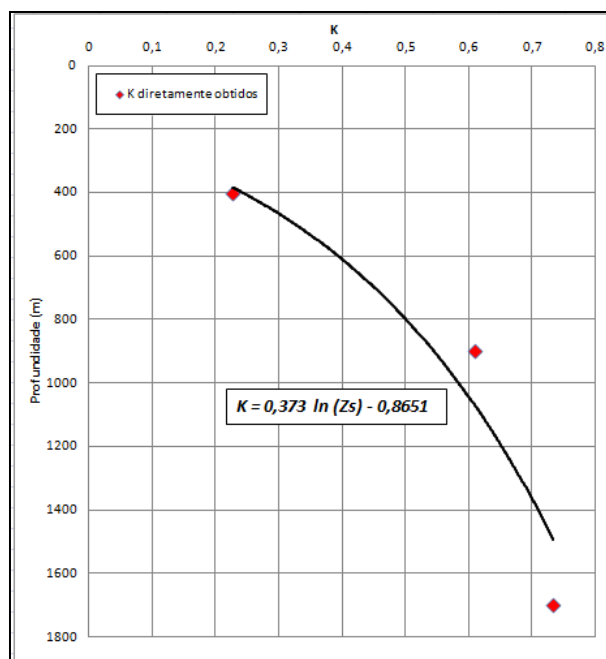


Figura 16 – Curva de K , calibrada pelos LOTS (elaboração própria)

2.3. Projeto dos Revestimentos

Revestimentos de poços de petróleo apresentam diversas funções. Estes são responsáveis por prevenir o colapso e fraturamento da parede do poço acima da sapata e isolar determinadas formações como forma a evitar contaminação tanto do ambiente (zonas de aquífero) quanto do poço (argilas reativas, formações salinas, zonas contendo gases ácidos, etc.). Também atuam em conjunto com o BOP permitindo o controle seguro do poço e funcionam como canal para o escoamento dos fluidos de perfuração retornando a superfície, entre outros e protegem zonas de menores gradientes de fratura, permitindo um aumento na pressão de fundo sem acarretar em perdas de circulação nessas zonas

Segundo Bourgoyne *et al.* (1986) os revestimentos tem sido uma das partes mais caras dentro de um programa de perfuração onde estudos mostraram que o custo com a compra de revestimentos para a perfuração atingia 18% do custo total de um poço offshore completado já na década de 1980. Em poços terrestres o peso dos revestimentos é ainda maior, da ordem de 50%. Daí a necessidade de um estudo aprofundado para a definição dos revestimentos adequados para compor um poço, levando-se em conta parâmetros como diâmetro,

espessura, grau do aço, tipo de aço, entre outros. Na Figura 17 podemos observar exemplos tubos de revestimento aplicados na indústria do petróleo.



Figura 17 – Tubos para revestimento de um poço (Thomas *et al.* 2002)

2.3.1. Tipos de revestimento

Segundo o American Petroleum Institute (API) o revestimento corresponde a um tubo de aço com diâmetro externo variando entre 4,5 e 20 polegadas, alojado e cimentado no poço de forma a promover as características citadas acima. O conjunto de revestimentos é, no entanto, subdividido em cinco diferentes grupos: revestimentos condutores, revestimentos de superfície, revestimentos intermediários, revestimentos de produção e *liners*. Na Figura 18 podemos observar três programas típicos de poço, bem como seus revestimentos.

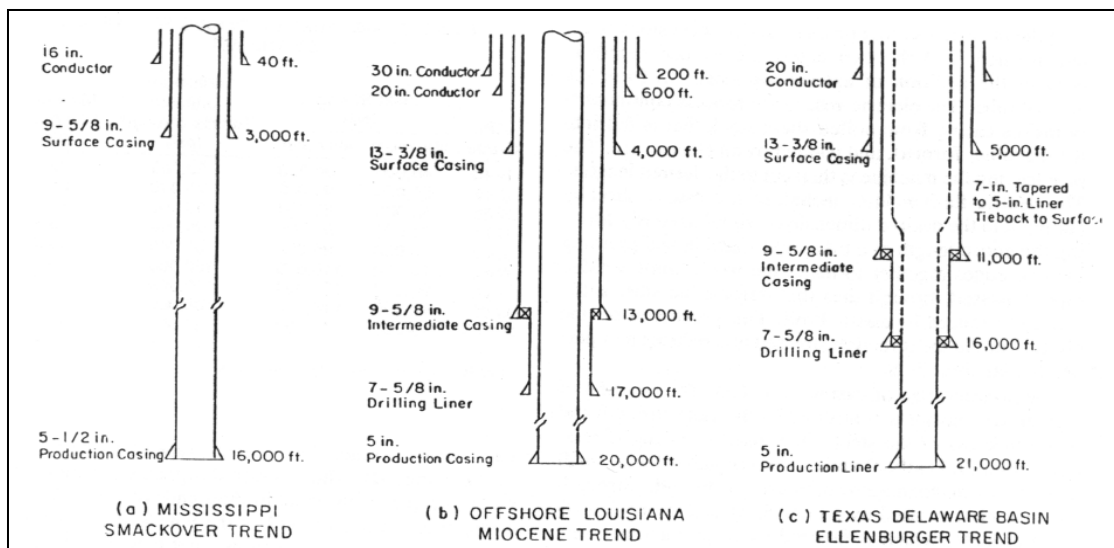


Figura 18 – Exemplos de programas de poço (Bourgoyne *et al.* 1986)

A) Revestimentos condutores

O revestimento condutor representa o primeiro revestimento a ser assentado no poço, e é a partir dele que a perfuração tem início. Esse revestimento tem, dentre outros, o objetivo de servir como base para equipamentos da cabeça do poço, além de proteger o revestimento de superfície de processos erosivos. Usualmente, em poços *offshore*, esses revestimentos são compostos por 3 a 5 juntas de 12 metros de comprimento, com diâmetro de 30 ou 36 polegadas no *offshore* e 20 polegadas em terra. Normalmente os condutores são jateados, mas também podem ser perfurados e cimentados, dependendo do cenário em questão.

B) Revestimentos de superfície

O revestimento de superfície tem a função de prevenir a erosão das formações rasas inconsolidadas, além de proteger aquíferos rasos de contaminação. Tais revestimentos também suportam a protegem de corrosão qualquer revestimento subsequente no poço. Usualmente a profundidade de assentamento da sapata desse revestimento é função de variáveis como a presença de perigos rasos (formações saturadas com hidrocarbonetos ou aquíferos), da janela operacional etc., sendo que o limitante superior dessa profundidade corresponde a máxima coluna de cimentação em um único estágio que se pode assegurar a

qualidade. Cimentações em dois estágios podem ser consideradas, dependendo da necessidade.

É importante destacar que o revestimento de superfície deve ter seu comprimento dimensionado também para garantir que, caso a perfuração da fase seguinte atinja zonas rasas saturadas por gás (gases rasos) não ocorra o dito *underground blowout*, fenômeno decorrente de uma pressurização excessiva do poço na ocorrência de um *kick*, que pode levar ao fraturamento da formação e a migração desse gás até a superfície por dentro da formação geológica.

C) Revestimentos intermediários

Revestimentos intermediários representam todos os revestimentos assentados entre o revestimento de superfície e o de produção. Esses revestimentos normalmente são aplicados em poços que atravessam zonas de sobrepressão, onde mais fases são projetados para o poço. Nesses casos é necessária o assentamento do revestimento como forma a proteger a região abaixo das sapatas dos revestimentos anteriores de diferenciais de pressão entre a formação e o poço. As profundidades de assentamento dependerão do critério de assentamento aplicado, bem como a janela operacional em questão.

D) Revestimentos de produção

Revestimento de produção é o nome dado a um conjunto de tubos de aço que reveste o poço até a zona do reservatório. Este tem a função de proteger as formações em caso de falha no *tubing* durante a produção, além de permitir reparos ou a substituição do *tubing* ao longo da vida produtiva do campo.

E) Liners

Os liners apresentam uma característica peculiar frente aos revestimentos citados acima, pois estes não se estendem até a superfície, mas sim são suspensos a partir da base do maior revestimento posicionado acima. A principal vantagem da aplicação dos liners é o custo reduzindo. *Liners* podem ser aplicados como forma de substituir os revestimentos de produção em poços sem completação a poço aberto.

2.3.2. Dimensionamento dos Revestimentos

Na indústria do petróleo os revestimentos, assim como outras tubulações aplicadas na cadeia produtiva, foram padronizados pela API, de forma a promover uma uniformidade na indústria quando se trabalha com dutos. Nesse contexto foi definido que se trabalharia com o conceito de grau do aço como um código que definiria a resistência mecânica daquele material. Tal código é composto por uma letra e um número, onde a letra foi escolhida de maneira arbitrária como forma a garantir a unicidade de cada aço. Já o número em questão representa o limite de escoamento mínimo do material, em *ksi*. Na Tabela 3 podemos ver os exemplos de graus de aços reconhecidos pela API.

Grau API	Limite de Escoamento	Limite de Escoamento
	Mínimo (<i>psi</i>)	Máximo (<i>psi</i>)
H-40	40.000	80.000
J-55	55.00	80.000
K-55	55.000	80.000
C-75	75.000	90.000
L-80	80.000	95.000
N-80	80.000	110.000
C-90	90.000	105.000
C-95	95.000	110.000
P-110	110.000	140.000

Tabela 3 – Graus de aços reconhecidos pela API (Bourgoyne et al., 1986, adaptado)

A resistência do aço, definida pelo grau, desempenha papel fundamental no estudo de dimensionamento dos revestimentos de um poço de petróleo. Segundo Bourgoyne *et al.* (1986) os revestimentos, durante sua fase de instalação, são solicitados mecanicamente por diversos carregamentos, dos quais os principais seriam: tensão axial, pressão interna e pressão externa. Nesse sentido o aço apresentaria três *propriedades* de resistência importantes para seu dimensionamento: resistência axial, resistência a explosão e resistência a colapso.

2.3.2.1. Resistência a Tensão Axial

O carregamento axial que solicita os revestimentos durante a sua instalação no poço é oriundo, basicamente, do peso do próprio revestimento. Em uma análise discretizada do revestimento ao longo da profundidade observa-se um perfil decrescente da tensão axial conforme se aproxima do fundo do poço. Dessa forma a máxima tensão axial seria sentida no topo do revestimento, que sustentaria o peso de toda a coluna. Esse então seria o ponto crítico para a análise do dimensionamento do revestimento para resistência a tensão axial.

Considerando que o revestimento é composto por uma coluna única, ou seja, não há juntas conectando as seções, a resistência a tensão axial do mesmo será dada por:

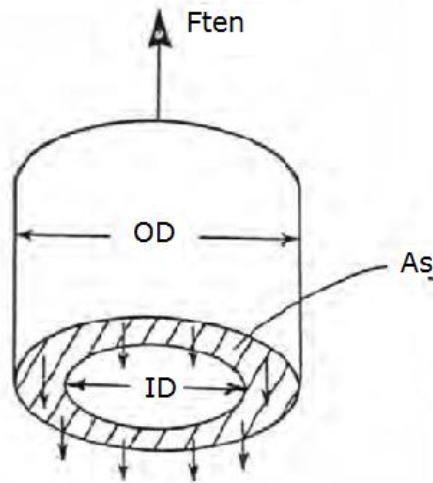


Figura 19 – Balanço de forças axiais no revestimento (Bourgoyne *et al.* 1986, adaptado)

$$F_{ten} = \sigma_y A_s \quad (25)$$

$$A_s = \frac{\pi}{4} (OD^2 - ID^2) \quad (26)$$

Nessas formulações temos que:

- F_{ten} resistência a tensão axial, em *lbf*;

- σ_y limite de escoamento mínimo do material, em *psi*;
- A_s área da superfície do revestimento, em *in*²;
- OD diâmetro externo do revestimento, em *in*;
- ID diâmetro interno do revestimento, em *in*.

Portanto, objetivando o correto dimensionamento do revestimento devemos garantir que o mesmo tenha dimensões e grau condizentes para que no ponto crítico a carga axial aplicada seja inferior a resistência a tensão axial.

$$F = w_{aco} \beta (Z_{well} - Z) \leq F_{ten} \quad (27)$$

$$\beta = 1 - \frac{\rho_f}{\rho_{aco}} \quad (28)$$

Nessas formulações temos que:

- F carga axial solicitada na profundidade Z , em *lbf*;
- w_{aco} peso específico do aço no revestimento, em *lbf/ft*;
- Z profundidade onde estamos calculando a tensão, em *ft*;
- Z_{well} profundidade do poço, em *ft*;
- β fator de empuxo, adimensional.

2.3.2.2. Resistência a Pressão Interna

Durante o assentamento dos revestimentos estes são solicitados ao mesmo tempo por pressões internas – pressão hidrostática oriunda do fluido que preenche esse revestimento – e externas – pressão hidrostática oriunda do fluido presente no anular do poço. Dessa forma diferenciais de pressão entre a pressão interna e externa ao revestimento podem levar a solicitações excessivas ao mesmo. Caso ele não seja dimensionado adequadamente,

podemos ocorrer a ruptura do revestimento ou colapso do mesmo, como veremos na seção seguinte.

O desequilíbrio entre as pressões interna (P_i) e externa (P_e) criam uma pressão equivalente (P_{eq}), de forma que quando $P_{eq} \neq 0$ significa que existe um diferencial solicitando o revestimento. Caso esta pressão equivalente seja negativa podemos ter uma situação característica de ruptura por pressão interna do revestimento, a depender da grandeza dessa pressão. Portanto o projeto de revestimentos deve ser feito de forma a garantir que na situação mais crítica onde $P_{eq} > 0$, esta não atinja a resistência a pressão interna.

$$P_{eq} = 1/OD (P_e OD - P_i ID) \quad (29)$$

A resistência a pressão interna de um tubo, segundo a padronização API, segue a equação de Barlow, que calcula a pressão interna de falha considerando que o a tensão circunferência atingiu a tensão de escoamento do material.

$$P_{br} = 0,875 \frac{2\sigma_y \delta}{OD} \quad (30)$$

Sendo assim é necessário que a pressão equivalente na condição crítica não ultrapasse a pressão de ruptura dada pela equação de Barlow. Portanto a condição a ser satisfeita no projeto será, basicamente:

$$P_{eq} \leq P_{br} \quad (31)$$

2.3.2.3. Resistência ao Colapso

Caso o desequilíbrio entre as pressões interna e externa gere uma pressão equivalente positiva o revestimento poderá ser submetido a colapso. A pressão de colapso de um tubo é função principalmente da razão entre o diâmetro externo e a espessura de parede (OD/δ), do limite de escoamento do material e da ovalização do tubo, sendo fenômeno

consideravelmente mais complexo que os fenômenos de explosão ou escoamento por tensão axial. Dessa forma a API definiu um critério para cálculo de pressão de colapso baseado apenas na razão OD/δ , considerando quatro perfis característicos de colapso: por resistência ao escoamento, plástico, de transição e elástico.

A) Colapso por resistência ao escoamento

$$P_{cr} = 2\sigma_y \left[\frac{OD/\delta - 1}{(OD/\delta)^2} \right] \quad (32)$$

O limite superior para aplicação desse critério é dado em função da razão OD/δ , sendo dado por:

$$\left(\frac{OD}{\delta} \right)_{yp} = \frac{\sqrt{(F_1 - 2)^2 + 8(F_2 + F_3/\sigma_y)} + (F_1 - 2)}{2(F_2 + F_3/\sigma_y)} \quad (33)$$

Nessas formulações temos que:

- P_{cr} pressão máxima para resistência ao colapso, em *psi*
- F_1, F_2, F_3 coeficientes empíricos que variam com o grau do aço

B) Colapso plástico

$$P_{cr} = \sigma_y \left[\frac{F_1}{(OD/\delta)} - F_2 \right] - F_3 \quad (34)$$

O limite superior para aplicação desse critério é dado em função da razão OD/δ , sendo dado por:

$$\left(\frac{OD}{\delta} \right)_{pt} = \sigma_y \frac{(F_1 - F_4)}{F_3 + \sigma_y(F_2 - F_5)} \quad (35)$$

Nessas formulações temos que:

- F_4, F_5 coeficientes empíricos que variam com o grau do aço

C) Colapso de transição

$$P_{cr} = \left[\frac{F_4}{(OD/\delta)} \right] - F_5 \quad (36)$$

O limite para a aplicação desse critério é definido pelos limites do colapso plástico e elástico, de forma que quando $\left(\frac{OD}{\delta}\right)_{pt} < (OD/\delta) < \left(\frac{OD}{\delta}\right)_{el}$ teremos o colapso pro transição.

D) Colapso elástico

$$P_{cr} = \frac{46,95 \times 10^6}{(OD/\delta)[(OD/\delta) - 1]^2} \quad (37)$$

O limite inferior para aplicação desse critério é dado em função da razão OD/δ , sendo dado por:

$$\left(\frac{OD}{\delta}\right)_{el} = \frac{2 + F_2/F_1}{3F_2/F_1} \quad (38)$$

Dessa forma podemos garantir que o revestimento projetado estará corretamente dimensionado para colapso caso, no ponto crítico do mesmo, tenhamos satisfeita a seguinte condição:

$$P_{eq} \leq P_{cr} \quad (39)$$

Portanto, tendo satisfeitas todas as condições de dimensionamento listadas acima ao mesmo tempo garantimos que o revestimento não será solicitado a carregamentos maiores que ele poderá suportar. Em um projeto fatores de segurança devem ser considerados, de forma a aumentar a segurança nos resultados do dimensionamento. Vale frisar que não

iremos abordar efeitos combinados nesse estudo, embora estes devam ser considerados para um segundo momento, a fim de tornar o dimensionamento mais próximo do caso real.

2.4. Projeto dos Equipamentos de Segurança - BOP

O BOP – *Blowout Preventer* – é um equipamento de segurança do poço que tem a função de vedar o anular do mesmo em casos de risco a segurança operacional, como, por exemplo, a ocorrência de influxos no poço. O BOP é um equipamento instalado após o assentamento do revestimento de superfície em conjunto com o riser de perfuração, provendo conexão do poço com a sonda de perfuração, e, portanto, permitindo o retorno do fluido de perfuração até a unidade, fato que não era possível até esse momento.

Como podemos ver na Figura 20 o BOP é um composto por um conjunto de válvulas que podem ser fechadas, vedando o espaço anular do poço ou fechando o poço como um todo a partir do acionamento da gaveta cega.

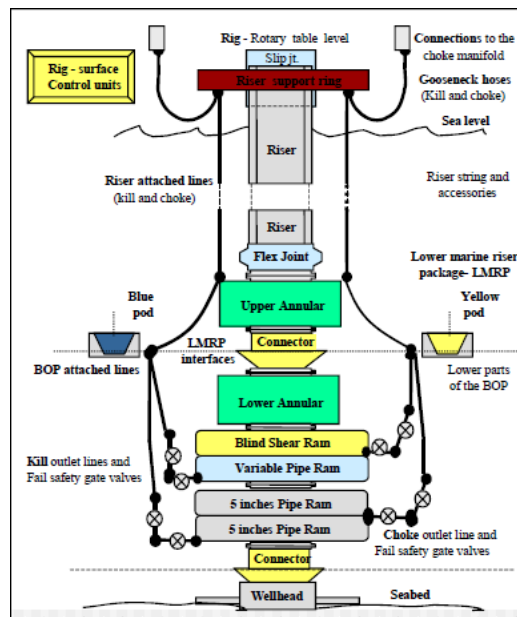


Figura 20 – Desenho esquemático de um BOP submarino (Thomas *et al.* 2002).

Segundo Rocha *et al.* (2006) existem atualmente no mercado preventores capazes de resistir a pressões da ordem de 5000, 10000, 15000 e até 20000 *psi*. O ponto importante é, portanto, verificar a pressão de trabalho do BOP para uma posterior escolha de sonda. Para

tal devemos lançar mão de critérios de dimensionamento de BOP, que levam em conta diversos fatores inerentes a perfuração, como pressão de poros, lâmina d'água, pressão de fratura, entre outras. Portanto avaliaremos quatro critérios de dimensionamento dos preventores: (i) baseada na pressão de poros, (ii) considerando o efeito de lâmina d'água, (iii) baseado na pressão de fratura e (iv) baseado no limite de resistência do revestimento.

A) Escolha do BOP baseada unicamente na pressão de poros

O critério de dimensionamento do BOP é baseado na hipótese de ocorrência de um kick no poço e, assim, todo o fluido presente tenha sido expulso, deixando o mesmo totalmente preenchido por gás. Nessas condições o BOP deve ser dimensionado para suportar a pressão máxima de poros no trecho perfurado, descontada a coluna hidrostática de gás no poço.

$$P_{BOP} = P_{p_{max}} - P_{H_{gas}} \quad (40)$$

$$P_{BOP} = \alpha (Z_{Gp_{max}} G_{pr_{max}} - Z_s \rho_{gas}) \quad (41)$$

Nessas formulações temos que:

- P_{BOP} pressão máxima de dimensionamento no BOP, em *psi*;
- $Z_{Gpr_{max}}$ profundidade da máxima pressão de poros, em *m*.

B) Escolha do BOP levando em conta os efeitos da lâmina d'água

Esse critério corresponde a uma correção na pressão máxima do BOP calculada por outros critérios de escolha devido à pressão hidrostática da coluna d'água que solicita o BOP. Esta atua no sentido oposto a pressão vinda de dentro do poço, proporcionando um balanço de pressões e uma redução na pressão de dimensionamento.

$$P_{BOP} = P_{p_{max}} - P_{H_{gas}} - P_{H_w} \quad (42)$$

$$P_{BOP} = \alpha (Z_{Gp_{max}} - Z_s \rho_{gas} - LDA \rho_w) \quad (43)$$

C) Escolha do BOP baseada na pressão de fratura

Esse critério visa estabelecer uma pressão máxima de dimensionamento no BOP que garanta que não haverá fraturamento na formação mais frágil caso ocorra um *kick* no poço, e todo o volume de fluido no poço seja deslocado pelo gás. Portanto a limitante desse critério será o gradiente de fratura abaixo da sapata do último revestimento.

$$P_{BOP} = P_{f_{min}} - P_{H_{gas}} \quad (44)$$

$$P_{BOP} = \alpha (Z_{G_{f_{min}}} G_{f_{min}} - Z_s \rho_{gas}) \quad (45)$$

Nessas formulações temos que:

- $Z_{G_{f_{min}}}$ profundidade da formação mais frágil do trecho perfurado, em m.

D) Escolha do BOP baseada na resistência do revestimento

Esse critério visa limitar a pressão máxima do BOP a pressão máxima de trabalho do revestimento, de forma que o preventor possa atingir tal pressão sem levar o revestimento a romper ou escoar. Dessa forma a pressão máxima do BOP seria equivalente a pressão de trabalho do revestimento.

$$P_{BOP} = P_{rev_{max}} \quad (46)$$

Nessas formulações temos que:

- $P_{rev_{max}}$ pressão máxima de trabalho do revestimento, em *psi*.

2.5. Projeto de Poço

O projeto de poço é uma fase fundamental no planejamento da perfuração, uma vez que é a partir dele, que se obtém informações como as profundidades de assentamentos das sapatas, configuração dos diâmetros dos revestimentos, peso da fluido aplicado entre outros. Esse projeto utiliza resultados oriundos do projeto de geopressões e, juntamente com o projeto de revestimentos e de equipamentos, compõe uma parte importante na fase de pré-perfuração.

Em linhas gerais o poço, independentemente de ser exploratório ou de desenvolvimento, é dividido em fases, sendo cada uma das fases determinada pelo diâmetro da broca ou do alargador, quando usado. Tais diâmetros devem seguir uma sequência lógica entre diâmetro da broca e do revestimento, de acordo que o número de fases do projeto e os diâmetros disponibilizados pelo mercado, além de outros critérios técnicos, como veremos ao longo dessa seção. Outro fator relevante no projeto de poço é a definição das profundidades de assentamento das sapatas. Segundo Rocha *et al.*(2006) estas devem ser definidas no sentido oposto ao da perfuração (da profundidade objetivo até a superfície), de forma a atender os requisitos exigidos pela geologia (para poços exploratórios) ou pelo reservatório (para poços de desenvolvimento). A combinação entre o número de fases, profundidades de assentamento do revestimento de cada fase, o diâmetro de cada fase, características da cada revestimento corretamente dimensionado entre outras constitui um projeto de poço típico.

2.5.1. Assentamento da Sapata dos Revestimentos

2.1.2.1. Introdução

Para o desenvolvimento de um projeto de poço é necessário que se determine a profundidade do assentamento da sapata do revestimento de cada fase do poço. Para tal são introduzidos determinados critérios que devem ser seguidos de forma a satisfazer determinados objetivos, como por exemplo:

- Atender às finalidades do poço;

- Fornecer segurança operacional a perfuração, bem como das instalações, das pessoas e os recursos aplicados na atividade;
- Preservar o meio ambiente;
- Fornecer um poço executável;
- Minimizar o custo da atividade dentro do projeto global

Durante a perfuração de um poço diferentes problemas podem ser enfrentados, sendo característicos da profundidade nas quais estes ocorrem (Rocha *et al.* 2006). Problemas como gases rasos, presença de aquíferos entre outros são típicos de profundidades mais rasas, abrangidas pelos revestimentos condutor e de superfície. Já para zonas mais profundas temos as regiões de sobrepressão, que eleva a possibilidade da ocorrer influxo (Aadnoy, 1999). Tal fato diminui a janela operacional do poço, aumentando o número de estágios do poço. Sendo assim, passa a ser interessante analisar o assentamento das sapatas em duas frentes: revestimentos em zonas superficiais e revestimentos em zonas mais profundas.

Basicamente o assentamento de revestimentos em zonas superficiais – condutor e revestimento de superfície – seguem os critérios apresentados na seção 2.3.1 deste trabalho. Dessa forma focaremos nos critérios de assentamento de revestimentos em zonas mais profundas, em particular os revestimentos intermediários.

2.1.2.2. Critérios de assentamento

Companhias de petróleo têm desenvolvido diferentes critérios para determinação das profundidades de assentamento (Rocha *et al.* 2006). Tais critérios têm como base a janela operacional definida pelos gradientes de poros e de fratura, como explicado no início da seção 2.2, uma vez que, independentemente do critério abordado, este deve garantir que peso de lama utilizado nos diferentes estágios do poço esteja dentro desta janela, mantendo assim a estabilidade do poço. Nesse contexto serão apresentados dois critérios de assentamento a seguir: critério baseado apenas na janela operacional e critério da margem de segurança de riser.

A) Critério baseado apenas na janela operacional

Esse critério leva em conta apenas a existência de uma janela operacional pré-definida pelo projeto das geopressões, que limitará a faixa do peso de fluido de perfuração utilizando dentro dos seguintes limites:

- Limite superior definido pelo menor valor entre os gradientes de fratura e de colapso superior;
- Limite inferior definido pelo maior valor entre os gradientes de poros e de colapso superior

Definidos os limites da janela operacional devemos projetar margens de segurança para a aplicação do critério. A partir da definição de tais margens, usualmente aplicadas sobre o limite inferior da janela operacional (Rocha *et al.* 2006), tem-se a redefinição da mesma, reduzindo a zona segura para a densidade do peso de fluido.

Portanto o critério consiste, basicamente, em fixarmos a densidade do fluido de perfuração como sendo igual ao limite inferior da janela operacional na profundidade objetivo do poço. A partir daí caminha-se ao longo da trajetória do poço em direção a superfície, mantendo-se o peso do fluido em questão. No ponto em que esse peso atingir o limite superior da janela deve-se assentar a sapata do revestimento. No mesmo ponto devemos redefinir o peso do fluido de perfuração como sendo igual ao limite inferior da janela para aquela mesma profundidade. Esse procedimento deve ser repetido sucessivamente até atingirmos a profundidade onde estaria assentado o revestimento de superfície. Na Figura 21 podemos observar o critério baseado na janela operacional em uma determinada trajetória de poço.

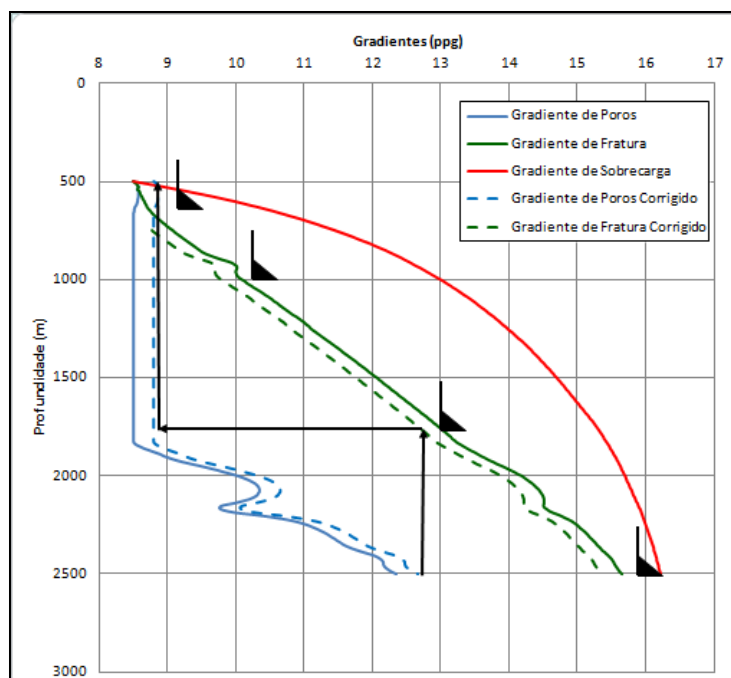


Figura 21 – Critério de assentamento de sapatas baseado apenas na janela operacional (elaboração própria).

B) Critério baseado na margem de segurança do riser

O critério de margem de riser foi desenvolvido para solucionar problemas causados por eventuais desconexões de emergência do riser no BOP em cenários onde a plataforma perde posição, ficando a deriva no mar. Nesses casos não é possível manter o riser dentro dos limites de inclinação permitidos para sua operação, acarretando na desconexão e no conseqüente fechamento do poço (através do fechamento das gavetas do BOP).

No cenário operacionalmente correto o BOP suportaria o adicional de pressão anteriormente fornecido pela coluna hidrostática de fluido ao longo do *riser*, mantendo a pressão de fundo. Entretanto, caso o BOP não funcione conforme o esperado, o poço ficaria aberto ao mar, reduzindo a pressão hidrostática de fundo. Dessa forma, caso a pressão atinja níveis inferiores a pressão de poros poderíamos ter eventos indesejados como influxos de gás.

Nesse sentido a margem de riser corresponde a um acréscimo na densidade do fluido de perfuração de modo a compensar a hidrostática dentro do poço caso aconteça uma

desconexão do riser e uma falha do BOP, levando a uma substituição da coluna de fluido de perfuração por uma coluna de água. Dessa forma o adicional de densidade do fluido será dado por:

$$\rho_{MR} = \frac{\rho_f Z - \rho_w LDA}{Z_s} - \rho_f \quad (47)$$

Nessas formulações temos que:

- ρ_{MR} densidade adicional dada pelo critério da margem de riser, em *ppg*;
- ρ_f densidade do fluido de perfuração, em *ppg*.

Caso a perfuração esteja sendo feita sem nenhum *overbalance* teríamos que:

$$\rho_{MR} = \frac{G_p Z - \rho_w LDA}{Z_s} - G_p \quad (48)$$

A partir do momento onde definimos a densidade adicional de fluido a ser utilizado a janela operacional para a perfuração também será modificada. Redefinida tal janela a metodologia da definição das profundidades das sapatas seguirá a mesma lógica do critério baseado apenas na janela operacional.

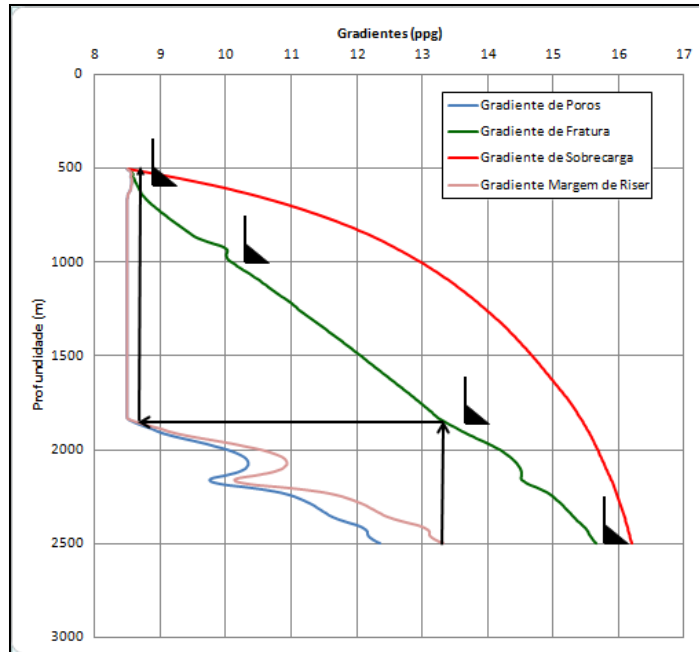


Figura 22 – Critério de assentamento de sapatas segundo a margem de segurança do riser (elaboração própria).

2.5.2. Projetos Típicos de Poços

No projeto de poço, assim como a determinação das sapatas a partir dos critérios colocados acima ou de outros, é fundamental a determinação do conjunto de diâmetros que serão utilizados em cada fase do poço. Este conjunto deverá ser projetado como forma de atender os objetivos do projeto, como citado na seção 2.5.1.1. Para cada estágio são determinados dois diâmetros fundamentais:

- O diâmetro da fase, que é definido pelo diâmetro da broca ou pelo diâmetro dos alargadores, caso estes sejam utilizados
- O diâmetro do revestimento

Nesse sentido existem diversas combinações possíveis entre diâmetros de brocas e de revestimentos de forma a compor o projeto do poço. É importante que a escolha leve em conta minimamente os seguintes critérios:

- O diâmetro da broca utilizada na fase seguinte deve ser menor ou igual o diâmetro de *drift* do revestimento
- O revestimento utilizado em cada fase deve ter diâmetro minimamente igual ao da broca (ou do alargador) utilizada na perfuração daquela fase

Satisfeitos tais critérios a escolha dos diâmetros específicos será feita basicamente como forma a atingir a profundidade final do poço com um revestimento de produção com diâmetro limitado pelo tamanho dos equipamentos de completção daquele poço e o número e fases que o mesmo terá. Todavia outros fatores também serão relevantes, tais como disponibilidade dos equipamentos no mercado, etc. A Figura 23 apresenta um fluxo que pode ser usado como guia na escolha de revestimentos, segundo critérios da Petrobras.

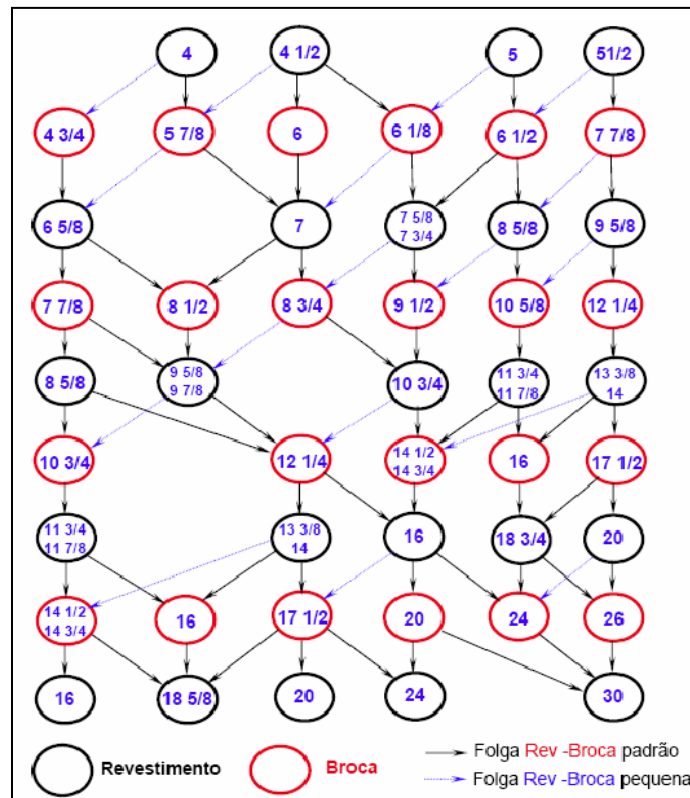


Figura 23 – Fluxo para escolha da combinação entre diâmetros de poços e de revestimentos, partindo-se do princípio do *drift* do revestimento (Rocha *et al.* 2006, adaptado)

Vale lembrar que o diâmetro do condutor em poços *offshore* costuma ser pré-definido em 30 polegadas (Petrobras), para uma fase de 36 polegadas, sem sofrer influência de fatores

como os citados acima. Para a fase do revestimento de superfície, normalmente aplica-se um revestimento de 20 polegadas para um diâmetro de broca de 26 polegadas, embora outros diâmetros também possam ser aplicados. Entretanto, poços projetados para serem perfurados com menos fases poderão ter configurações diferentes dos demais projetos, como é o caso das configurações *Slender*, onde não é descido o revestimento de superfície, conforme apresentado na Figura 24 a seguir.

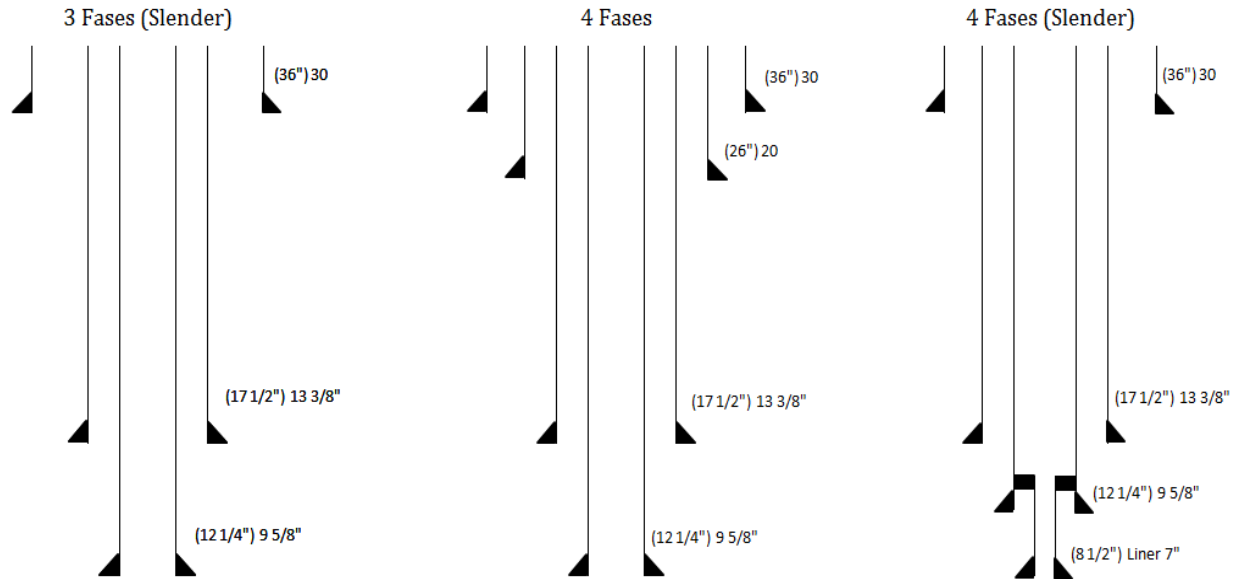


Figura 24 – Projetos típicos usualmente aplicados em poços rasos ou zonas normalmente pressurizadas (Rocha *et al.* 2006, adaptado)

3. Abordagem Teórica do Módulo de Perfuração em Tempo Real

3.1. Introdução

O módulo de perfuração em tempo real constitui o segundo bloco do simulador e tem como objetivo principal fornecer uma ferramenta através da qual se pode acompanhar a atividade de perfuração de um poço na velocidade que mesma acontece, levantando parâmetros relevantes e permitindo atuação direta do usuário nessa atividade, bem como tal atuação influenciou no comportamento da perfuração.

É importante observar que a perfuração é uma atividade de complexidade elevada, tendo diversos conceitos e modelos envolvidos, tais como hidráulica de poço, interação broca-rocha, comportamento da coluna, entre vários outros. Todavia, a fim de caracterizarmos o nosso modelo de perfuração em tempo real, iremos focar em dois grandes blocos: modelagem da hidráulica de poço e a modelagem do comportamento da broca, com enfoque no modelo de perfuração para brocas com partes móveis

3.2. Modelagem da Hidráulica de Poço

A mecânica dos fluidos exerce papel muito importante na atividade de perfuração de poços. Ao longo de toda essa atividade lidamos com pressões elevadas dentro do poço, devendo-se manter cuidado para que tais pressões não ultrapassem os limites impostos pela respectiva janela operacional. Além disso, o perfil de pressão no interior do poço é importante para uma análise estrutural da coluna de perfuração, além de ter impacto na taxa de penetração do mesmo (Bourgoyne *et al.* 1986).

Portanto, nessa seção, abordaremos a modelagem teórica dos fenômenos de escoamento ao longo da coluna e do anular tanto em condições dinâmicas em regime permanente, quanto em condições estáticas, além da modelagem do escoamento na broca e na bomba. Para tal consideraremos a aplicação de três reologias diferentes: fluidos newtonianos, plásticos binghamianos e que seguem a lei de potências, além de considerarmos escoamentos tanto laminares quanto turbulentos.

3.2.1. Hidráulica de poço em condições estáticas

Na perfuração de poços condições estáticas, ou hidrostáticas, são usualmente observadas em ocasiões de parada de circulação, seja para manobra, descida de revestimento, circulação de kick ou qualquer outra circunstância que a justifique.

Ao contrário da pressão em condições dinâmicas, a pressão hidrostática para líquidos é função apenas da densidade do fluido, da profundidade e da aceleração da gravidade, sendo caracterizada basicamente pelo Teorema de Stevin, onde a pressão em um determinado ponto é diretamente *proporcional* ao peso da coluna de fluido acima deste.

$$\frac{dP}{dZ} = \rho_f g \quad (49)$$

$$\int_{P_0}^P dP = \int_{Z_0}^Z \rho_f g dZ \quad (50)$$

$$P_h = \rho_f g (Z - Z_0) + P_0 \quad (51)$$

Para o escopo da perfuração podemos reescrever a equação acima como sendo:

$$P_h = \alpha \rho_f Z + P_0 \quad (52)$$

Nessas formulações temos que:

- P_h pressão hidrostática no poço, em *psi*;
- P_0 pressão na cabeça do poço, em *psi*;
- α fator de conversão de unidade, igual a 0,1704 se $[Z] = m$ ou 0,052 se $[Z] = ft$;
- ρ_{fluido} massa específica do fluido de perfuração, em *ppg*.

Uma aplicação importante do perfil de pressão em condições estáticas está na determinação da densidade de fluido de perfuração a ser utilizado na perfuração de um

determinado estágio. É importante que a densidade do fluido utilizado seja sempre adequado para uma determinada janela de operação, independentemente de o *ECD* estar ou não adequado. Isso é relevante, pois, caso haja uma parada de circulação, as componentes dinâmicas da pressão de fundo não existirão mais, e a pressão será regida apenas pela equação 52. Nesse sentido, caso a densidade de fluido não seja adequada, poderemos ter que lidar com *kick*.

Na Figura 25 podemos ver o perfil de pressão hidrostática ao longo de um trecho do poço que está preenchido um fluido de 10,5 *ppg*.

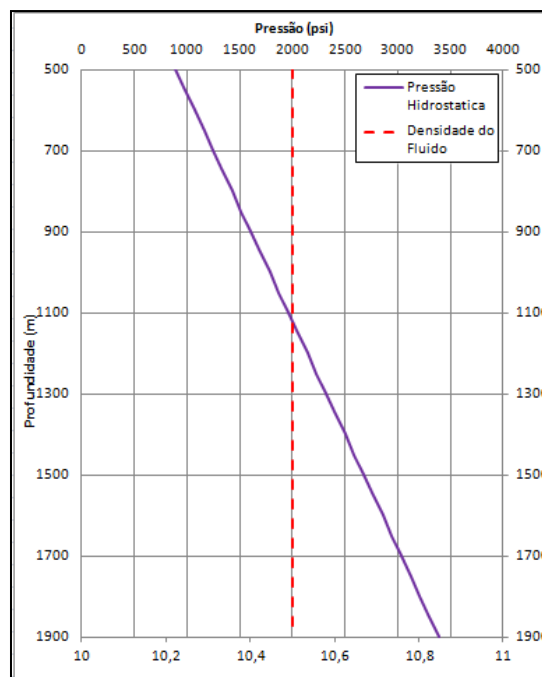


Figura 25 – Perfil de pressão hidrostática em um poço com um fluido de 10,5 *ppg* (elaboração própria).

3.2.2. Hidráulica de poço em condições dinâmicas

3.2.2.1. Introdução

Durante a atividade de perfuração não é possível caracterizar o perfil de pressão apenas através do modelo hidrostático. Isso ocorre uma vez que o fluido de perfuração é circulado ao longo do poço como forma a carrear os cascalhos até a superfície. Nessa atividade, chamada

de circulação, o fluido de perfuração atravessa uma bomba que eleva a pressão deste, dotando-o de energia. Com essa condição o fluido entra no poço pela coluna, atravessando-a até atingir a broca. Daí o fluido segue para o anular, onde migra até a superfície, carregando os cascalhos gerados durante a perfuração. Ao longo de todo esse percurso, ilustrado na Figura 26, o fluido perde pressão por fricção, de forma que a pressão dinâmica para uma determinada profundidade Z seja sempre maior que a pressão estática. Isso pode ser observado na equação abaixo, resultado do balanço de energia do escoamento do fluido de perfuração por dentro da coluna.

$$P_{coluna}(Z) = P_p + P_h(Z) - (\Delta P_f)_{coluna} \quad (53)$$

$$(\Delta P_f)_{coluna} = (\Delta P_f)_{dp} + (\Delta P_f)_{hwdp} + (\Delta P_f)_{dc} + (\Delta P_f)_b \quad (54)$$

Ou então, por dentro do anular.

$$P_{anular}(Z) = P_{coluna}(Z_{well}) - P_h(Z) - (\Delta P_f)_{anular} \quad (55)$$

$$\begin{aligned} (\Delta P_f)_{anular} = & (\Delta P_f)_{anular/dp} + (\Delta P_f)_{anular/hwdp} + (\Delta P_f)_{anular/dc} \\ & + (\Delta P_f)_{riser/dp} + (\dots) \end{aligned} \quad (56)$$

Nessas formulações temos que:

- P_{coluna} pressão dinâmica ao longo da coluna de perfuração, em *psi*;
- P_p pressão de bombeio, em *psi*;
- ΔP_f perda de carga por fricção, em *psi*;
- P_{anular} pressão dinâmica ao longo da coluna de perfuração, em *psi*.

Nas seções seguintes veremos como devem ser calculadas as variáveis em questão, como a pressão de bombeio, perda de carga na broca e, particularmente, a perda de carga ao longo da coluna e no anular, que será função da reologia do fluido utilizado, bem como tipo de regime de escoamento e outras variáveis relevantes.

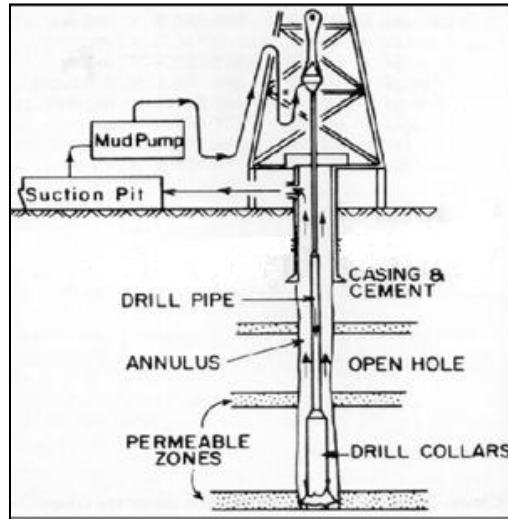


Figura 26 – Diagrama ilustrativo de um sistema de circulação de fluidos (Bourgoyne *et al.* 1986)

3.2.2.2. Hidráulica na bomba de lama

A pressão de bombeio executa papel fundamental na regulação da pressão no poço. Esta é a responsável por garantir que o fluido seja capaz de vencer toda a perda de carga ao longo do poço, dada por $(\Delta P_f)_{coluna} + (\Delta P_f)_{anular}$, e atingir os tanques de lama na superfície com uma determinada pressão inicial. Portanto, a menor pressão que a bomba deverá fornecer para fornecer ao fluido é dada por:

$$P_p = (\Delta P_f)_{coluna} + (\Delta P_f)_{anular} \quad (57)$$

$$\begin{aligned} P_p = & (\Delta P_f)_{dp} + (\Delta P_f)_{hwdp} + (\Delta P_f)_{dc} + (\Delta P_f)_{broca} + (\Delta P_f)_{anular/dp} \\ & + (\Delta P_f)_{anular/hwdp} + (\Delta P_f)_{anular/dc} + (\Delta P_f)_{riser/dp} \\ & + (\dots) \end{aligned} \quad (58)$$

Dessa forma podemos definir a potência exigida ao motor da bomba ao longo da perfuração. Segundo Bourgoyne *et al.* (1986) a potência da bomba de lama é função da pressão e da vazão fornecida ao fluido, que por sua vez é um parâmetro do projeto. A mesma será dada, em unidades de campo, por:

$$POT_p = \frac{P_p q}{1714} \quad (59)$$

Nessas formulações temos que:

- POT_p potência exigida do motor da bomba, em *HP*;
- q vazão do fluido que sai da bomba, em *gal/min*.

3.2.2.3. Hidráulica na broca

Para um estudo de hidráulica do poço a análise do comportamento de pressão na broca se faz importante, uma vez que é nesse ponto que o fluido experimenta boa parte da sua perda de carga. Isso ocorre devido a redução acentuada na velocidade do fluido e na direção do escoamento.

Em linhas gerais, antes de atingir a broca, o fluido escoava internamente ao longo da coluna de perfuração com diâmetros nominais característicos que variam entre 2 3/8” e 6 5/8” (Thomas *et al.* 2001). Ao atingir a broca o fluido deverá escoar através dos jatos da mesma, de diâmetros substancialmente menores. Como a vazão ao longo da circulação pode ser considerada constante e a área permissível ao fluxo na coluna é muito maior que nos jatos da broca o fluido terá a sua velocidade aumentada de maneira substancial. Daí, ao ser expelido pelos jatos atingindo a formação, a direção do fluxo é alterada abruptamente, bem como a velocidade do mesmo. Esses fenômenos levam a uma perda de carga acentuada nessa zona. Nesse sentido a perda de carga na broca pode ser representada, em unidades de campo, por:

$$(\Delta P_f)_b = \frac{8,311 \times 10^{-5} \rho_f q^2}{C_d^2 A_t^2} \quad (60)$$

$$A_t = \frac{\pi}{4} \sum_1^n (OD_{jatos})_n^2 \quad (61)$$

Nessas formulações temos que:

- A_t área total permissível ao fluxo, em in^2 ;
- C_d coeficiente de descarga, normalmente igual a 0,95
(Bourgoyne *et al.* 1986)

Devido aos fenômenos de redução da velocidade e alteração abrupta da direção de escoamento, além da perda de carga acentuada dada pela equação acima, a broca impõe a formação uma força de impacto hidráulico que tem como objetivo principal a facilitação na limpeza do fundo do poço, auxiliando no carreamento dos cascalhos. Além disso, tal força de impacto tem influência sobre a taxa de penetração da perfuração, em particular para as formações mais superficiais (Bourgoyne *et al.* 1986). Nesse sentido é considerado que ocorrerá a total transferência do momento do fluido para o fundo do poço, resultando em que a força de impacto hidráulico, em unidades de campo, seja dada por:

$$F_j = 0,01823 C_d q \sqrt{(\Delta P_f)_b \rho_f} \quad (62)$$

Nessas formulações temos que:

- F_j força de impacto hidráulico, em lbf .

3.2.2.4. Hidráulica ao longo do poço – Modelos Reológicos

A reologia constitui papel fundamental na caracterização do fluido para a análise de hidráulica do poço. Cada modelo reológico apresenta uma solução distinta para perda de carga, além de diferentes parâmetros reológicos significativos. Sendo assim é importante

estudar cada modelo separadamente, como forma a apresentar as soluções clássicas para hidráulica em poços preenchidos com fluidos que respeitam os modelos (i) newtonianos, (ii) plástico de Bingham e (iii) lei de potências.

A) Modelo Newtoniano

Um fluido dito newtoniano é caracterizado por uma relação diretamente proporcional entre a tensão e a respectiva taxa de cisalhamento, sem haver resistência inicial ao fluxo. Essa *proporção* é dada pela viscosidade do fluido, que será necessariamente constante ao longo de todo escoamento.

$$\tau = \mu \dot{\gamma} \quad (63)$$

$$\frac{\partial \tau}{\partial \dot{\gamma}} = \mu = \text{constante} \quad (64)$$

Nessas formulações temos que:

- τ tensão cisalhante sobre o fluido, em lbf/ft^2 ;
- $\dot{\gamma}$ taxa de cisalhamento do fluido, em s^{-1} ;
- μ viscosidade, ou viscosidade aparente do fluido, em cp .

Dessa forma o modelo de fluido newtoniano se caracteriza por ter na viscosidade o único parâmetro reológico significativo no seu estudo, ao contrário do que acontece com os demais modelos que serão analisados em seguida.

B) Modelo Plástico de Bingham

Ao contrário do modelo newtoniano, fluidos que respeitam o modelo plástico de Bingham, ou modelo plástico ideal (Machado, 2002), apresentam uma resistência inicial ao escoamento quando aplicado uma tensão cisalhante inferior aquela necessária para deformar o fluido. Essa tensão limite é definida como ponto de cisalhamento (Bourgoyne *et al.* 1986).

A partir do instante que a tensão aplicada supera o ponto de cisalhamento o fluido começará a se deformar, de forma que a relação entre a taxa e a tensão de cisalhamento será

constante e positiva, da mesma forma que no modelo newtoniano. A proporcionalidade será dada pela viscosidade plástica daquele fluido.

$$\tau = \mu_p \dot{\gamma} + \tau_0, \quad \tau > \tau_0 \quad (65)$$

$$\dot{\gamma} = 0, \quad \tau < \tau_0 \quad (66)$$

$$\frac{\partial \tau}{\partial \dot{\gamma}} = \mu = \text{constante} \quad (67)$$

Nessas formulações temos que:

- μ_p viscosidade plástica do fluido, em *cp*;
- τ_0 ponto de cisalhamento do fluido, em *lbf/100 ft²*;

C) Modelo de Lei de Potências

Fluidos que obedecem à lei de potências (ou lei de Ostwald de Waale) têm a característica de não apresentar uma relação linear entre a tensão de cisalhamento sofrida pelo fluido e a taxa com que ele se deforma (assim como outros fluidos não newtonianos). Nesse sentido passam a ser parâmetros significativos na análise reológica desses fluidos o índice de consistência, *K*, e o índice de comportamento, ou de fluxo, *n*.

$$\tau = K (\dot{\gamma})^n \quad (68)$$

$$\frac{\partial \tau}{\partial \dot{\gamma}} = n K (\dot{\gamma})^{n-1} \neq \text{constante} \quad (69)$$

Nessas formulações temos que:

- *K* índice de consistência, em *eq. cp*;
- *n* índice de comportamento, adimensional.

Dependendo o valor assumido pelo índice de comportamento do fluido este apresentará características reológicas diferenciadas, que impactarão sensivelmente nos seus cálculos de perda de carga. Se o índice n for maior que 1 teremos um fluido tipicamente dilatante, onde a razão entre tensão e a taxa de cisalhamento varia positivamente, não sendo interessante aos olhos da perfuração. Já fluidos que apresentam o índice n maior que zero e menor que 1 são tipicamente pseudo-plásticos, onde a razão entre tensão e a taxa de cisalhamento varia negativamente, sendo, esse sim, interessante para aplicação como fluidos de perfuração.

Interessante notar que o modelo newtoniano é uma particularização desse modelo, onde n é igual a 1 e K representa a viscosidade aparente.

Na Figura 27 podemos observar o perfil apresentado pelos fluidos que obedecem aos modelos descritos acima.

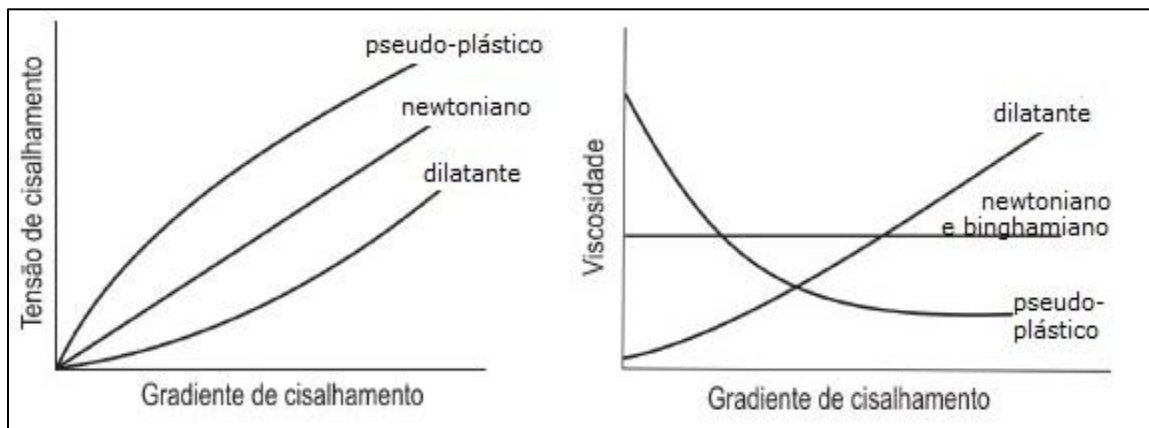


Figura 27 – Modelos Reológicos (Machado, 2002, adaptado).

3.2.2.5. Hidráulica ao longo do poço – Cálculo da perda de carga

O cálculo da hidráulica ao longo do poço em condições dinâmicas constitui uma das etapas mais complexas do estudo do escoamento do fluido de perfuração durante a perfuração. Os cálculos das componentes de perda de carga por fricção ao longo da coluna, bem como no anular do poço, são funções de diversos parâmetros, tais como aqueles relativos a dinâmica do escoamento, como a vazão; a geometria da coluna, como o diâmetro externo (OD), diâmetro interno (ID), rugosidade da superfície, etc.; e parâmetros reológicos

do fluido em questão, como densidade e viscosidade; além do próprio modelo reológico do mesmo.

Nesse sentido iremos analisar separadamente o equacionamento para estimação da perda de carga por fricção de cada um dos modelos reológicos explicados na seção 3.2.2.4, tanto para dentro da coluna de perfuração quanto para o espaço anular do poço, além de considerar regimes turbulentos e laminares de escoamento. Todavia os modelos desenvolvidos para o cálculo da perda de carga consideram algumas simplificações genéricas, visando tornar a análise mais simples para ser desenvolvida analiticamente. Dentre tais simplificações podemos citar:

- O fluido é incompressível;
- A temperatura durante a perfuração não impacta no comportamento do escoamento;
- Os períodos típicos de escoamento transiente – início da circulação, mudança de vazão e parada da circulação – não serão considerados. A modelagem considerará que todo o escoamento se dá de forma permanente;
- No instante do cálculo de perda de carga a coluna de perfuração esta estática em relação ao fluido escoando

A) Modelo Newtoniano

Como apresentado na seção anterior um fluido dito newtoniano é todo e qualquer fluido que obedece ao modelo newtoniano em questão. A partir desse modelo podemos expressar equações que estimem a perda de carga de um fluido desse tipo tanto para dentro da coluna quanto para o anular do poço, tanto para regimes turbulentos quanto laminares. Tais discretizações passam a depender dos parâmetros geométricos do problema, além dos reológicos e físicos que regem o escoamento.

i) Escoamento ao longo da coluna

O processo de circulação do fluido ao longo da perfuração se inicia na coluna, por onde a lama atravessa respectivamente o espaço interno dos *drill pipes*, dos *heavy-weight drill pipes* e, finalmente, dos *drill collars*, até atingir a broca. Esse escoamento pode ser

caracterizado como laminar ou turbulento, dependendo das condições que o regem. Para cada um desses perfis existem diferentes modelos de perda de carga *propostos* na literatura, sendo fundamental a determinação do perfil de escoamento para uma correta estimativa dessa perda de carga.

A importância da definição do perfil de escoamento em laminar ou turbulento começou a ser estudada ainda no século XIX, quando George Stokes observou que, quando o razão entre as forças inerciais e as forças viscosas de um fluido atingia um determinado limite o escoamento perdia seu perfil laminar anterior. A partir desse ponto o escoamento passou a ser caracterizado como de transição, até, finalmente, atingir o perfil turbulento. E a essa razão deu-se o nome de Número de Reynolds.

Para o escoamento dentro da coluna de um fluido newtoniano o número de Reynolds, em unidades de campo, é dado por:

$$Re = 928 \frac{\rho_f \bar{v} ID}{\mu} \quad (70)$$

$$\bar{v} = \frac{q}{2,448 ID^2} \quad (71)$$

Nessas formulações temos que:

- Re número de Reynolds, adimensional;
- \bar{v} velocidade média de área, em ft/s .

Para considerações de engenharia o escoamento pode ser considerado laminar se o número de Reynolds for inferior a 2100. Já escoamentos que apresentam o número de Reynolds superior a 2100 já podem ser considerados turbulentos, embora, em análises mais criteriosas, para $2000 < Re < 4000$ tem-se uma fase de escoamento de transição. Dessa forma a modelagem da perda de carga ao longo da coluna por um fluido newtoniano pode ser representada pela equação de Fanning apresentada abaixo em unidades de campo (Bourgoyne *et al.* 1986).

$$\frac{dP_f}{dL} = f \frac{\rho_f \bar{v}^2}{25,8 ID} \quad (72)$$

Nessas formulações temos que:

- f coeficiente de fricção, adimensional, varia entre zero e 1

A partir dessa equação é possível calcularmos a perda de carga para qualquer tipo de escoamento, seja ele laminar ou turbulento, apenas pela definição do coeficiente de fricção f . Esse, por sua vez, pode ser obtido graficamente a partir do diagrama de Stanton (Figura 28), que é uma representação dos resultados compilados do $f = f(Re, \epsilon/ID)$ através da função de Colebrook (Eq.73) para regime turbulento e da solução de $f = f(Re)$ (Eq.74) para regime laminar.

$$\frac{1}{\sqrt{f_{turb}}} + 4 \log \left(0,269 \frac{\epsilon}{ID} + \frac{1,255}{Re \sqrt{f_{turb}}} \right) = 0 \quad (73)$$

$$f_{lam} = \frac{16}{Re} \quad (74)$$

Nessas formulações temos que:

- ϵ rugosidade absoluta da superfície da coluna, em *in*

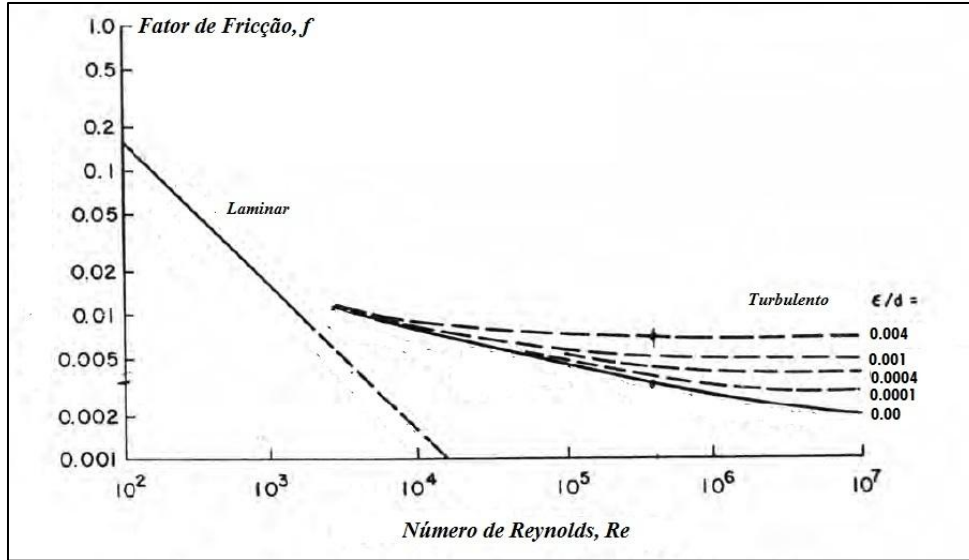


Figura 28 – Diagrama de Stanton (Bourgoyne *et al.* 1986, adaptado)

ii) *Escoamento ao longo do anular do poço*

Após atravessar toda a coluna e a broca o fluido de perfuração atinge o anular, através do qual ele atingirá a os tanques de lama na superfície, carregando os cascalhos. Nesse trajeto o fluido também perderá carga, assim como ao longo da coluna. Entretanto é uma prática comum aplicar a mesma metodologia apresentada anteriormente para escoamento dentro de colunas para outras geometrias, apenas corrigindo-se o diâmetro aplicado. Esse novo diâmetro é definido como diâmetro equivalente. Nesse sentido o anular é estudado exatamente como um tubo de geometria circular, com diâmetro igual ao diâmetro circular equivalente (Bourgoyne *et al.* 1986)

Na literatura existem 4 correlações típicas para definição do diâmetro circular equivalente para geometrias anulares.

$$(De)_1 = D_2 - D_1 \quad (75)$$

$$(De)_2 = \sqrt{D_2^2 - D_1^2 - \frac{D_2^2 - D_1^2}{\ln(D_2/D_1)}} \quad (76)$$

$$(De)_3 = 0,816(D_2 - D_1) \quad (77)$$

$$(De)_4 = \frac{\sqrt[4]{D_2^4 - D_1^4 - \frac{(D_2^2 - D_1^2)^2}{\ln(D_2/D_1)}} + \sqrt{D_2^2 - D_1^2}}{2} \quad (78)$$

Nessas formulações temos que:

- De diâmetro circular equivalente, em *in*;
- D_1 diâmetro externo da coluna, em *in*;
- D_2 diâmetro da fase ou do interno do revestimento, em *in*.

A partir da definição do diâmetro equivalente correto a metodologia para os cálculos da perda de carga ao longo do anular é a mesma apresentada na seção anterior, apenas corrigindo-se o diâmetro interno da coluna pelo diâmetro circular equivalente.

B) Modelo Plástico de Bingham

Conforme apresentado na seção 3.2.2.4 o modelo plástico de Bingham em muito se assemelha ao modelo newtoniano, exceto pelo ponto de cisalhamento. Entretanto, a partir de resultados empíricos, foi observado que o efeito desse parâmetro na perda de carga ao longo do seu escoamento só é significativo durante o regime laminar. Para taxas de cisalhamento muito altas, equivalentes a um regime turbulento plenamente desenvolvido, este parâmetro perde sua significância e a perda de carga passa a ser afetada basicamente pela densidade e viscosidade plástica do fluido (Bourgoyne *et al.* 1986).

i) Escoamento ao longo da coluna

Foi observado experimentalmente que as equações *propostas* para fluidos que obedecem ao modelo newtoniano também apresentavam resultados satisfatórios quando aplicadas para fluidos binghamianos, apenas substituindo-se a viscosidade aparente newtoniana pelo viscosidade plástica no cálculo do número de Reynolds, e definindo, a partir daí, o critério de escoamento. Isso se dá uma vez que para taxas de cisalhamento elevadas ambos os fluidos

passam a ter comportamentos bem semelhantes. Entretanto essa simplificação não é suficiente para descrever todos os eventos comuns ao longo da perfuração.

Sendo assim a forma mais aplicada na indústria para determinação do cálculo da perda de carga de um fluido binghamiano ao longo de uma coluna, bem como para a aplicação do critério de escoamento do mesmo, consiste em calcularmos a viscosidade newtoniana equivalente a partir dos parâmetros reológicos significativos desse fluido. Assim estaríamos estudando um fluido de Bingham de acordo com o modelo newtoniano. Essa viscosidade equivalente para escoamento dentro da coluna seria dada por:

$$\mu_a = \mu_p + 6,66 \frac{\tau_0 ID}{\bar{v}} \quad (79)$$

A partir daí as mesmas equações apresentadas na seção anterior para fluidos newtonianos podem ser aplicadas. Deve-se calcular a velocidade média de área pela Eq. 71, o número de Reynolds a partir da Eq. 70, e assim verificar se o escoamento é caracterizado como laminar ou turbulento, comparando com o limite para escoamento laminar $Re = 2100$. Caso o mesmo seja laminar determina-se o coeficiente de fricção pela Eq. (74) e a perda de carga aplicando a equação de Fanning (Eq. 72). Caso o escoamento seja turbulento ($Re > 2100$) basta aplicar a equação de Colebrook (Eq. 73) para determinar o coeficiente de fricção. Em todas essas equações a viscosidade aplicada é a viscosidade newtoniana equivalente.

Entretanto Hanks *et al.* (1967) *apud* Borgoyne *et al.* (1986) propôs um novo critério de turbulência para fluidos binghamianos, ao descobrir que existe uma correlação matemática entre o número adimensional de Hedstron e o número de Reynolds a partir do qual tais fluidos apresentam comportamento turbulento. Esse novo critério fornecia resultados melhores que aquele da viscosidade newtoniana equivalente quando aplicada a equação de Colebrook (Eq. 73) (Bourgoyne *et al.* 1986).

Esse critério não estuda o fluido de Bingham como sendo um fluido newtoniano e, portanto, utiliza os próprios parâmetros reológicos do próprio modelo ao longo dos cálculos. Em unidades de campo o número de Hedstron pode ser dado por:

$$He = \frac{37100\rho_f\tau_0ID}{\mu_p^2} \quad (80)$$

Nessas formulações temos que:

- He número adimensional de Hedstron

O número de Reynolds crítico para início do escoamento turbulento se correlaciona com o número de Hedstron através do gráfico apresentado na Figura 29. Tal gráfico é resultado da solução numérica simultânea do sistema de equações não linear apresentado a seguir (Eqs. 81 e 82) (Hanks *et al.* 1967 *apud* Bourgoyne *et al.* 1986).

$$\frac{\left(\frac{\tau_0}{\tau_w}\right)}{\left(1 - \frac{\tau_0}{\tau_w}\right)^3} = \frac{He}{16800} \quad (81)$$

$$Re_c = \frac{1 - \frac{4}{3}\left(\frac{\tau_0}{\tau_w}\right) + \frac{1}{3}\left(\frac{\tau_0}{\tau_w}\right)^4}{8\left(\frac{\tau_0}{\tau_w}\right)} \quad (82)$$

Nessas formulações temos que:

- τ_w tensão de cisalhamento na parede do duto, em $lb_f/100ft^2$

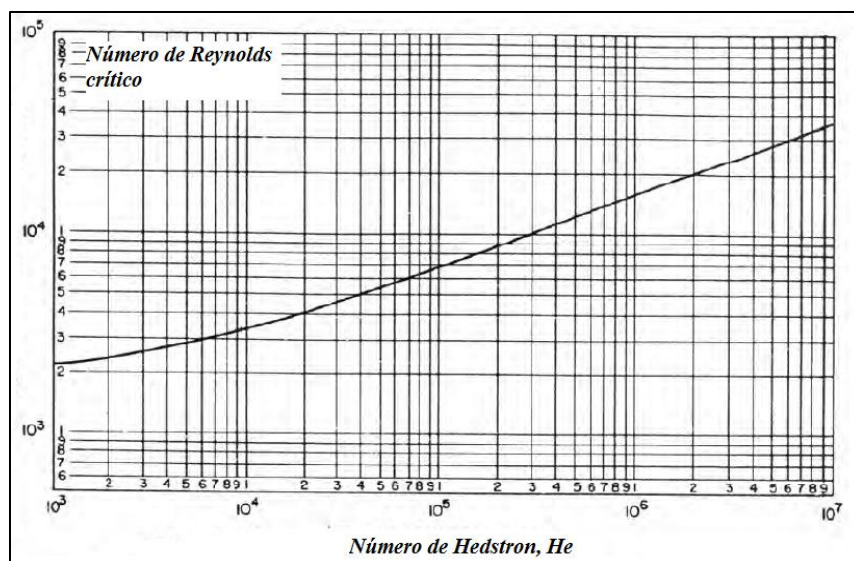


Figura 29 – Número de Reynolds crítico para fluidos plásticos de Bingham (Bourgoyne *et al.* 1986, adaptado)

Nesse critério o escoamento é considerado turbulento se o número de Reynolds calculado a partir da Eq. 70, considerando a viscosidade como a própria viscosidade plástica do fluido de Bingham, for maior que o número de Reynolds crítico obtido pelo gráfico acima (Figura 29). Dessa forma o cálculo da perda de carga seguirá os mesmos passos apresentados para o fluido newtoniano, entretanto é importante frisar que a viscosidade aplicada nas equações será a viscosidade plástica, e não mais a viscosidade newtoniana equivalente, como propunha o critério anteriormente apresentado para fluidos binghamianos.

ii) *Escoamento ao longo do anular*

Para fluidos que respeitam o modelo plástico de Bingham o cálculo da perda de carga ao longo do espaço anular do poço segue o mesmo modelo apresentado para a perda de carga ao longo da coluna, apenas corrigindo-se a geometria circular de diâmetro ID (no caso da coluna) para um diâmetro circular equivalente (Eqs. 75, 76, 77 e 78). Isso é válido tanto para o critério de turbulência proposto por Hanks (advindo do número de Hedstron) quanto aplicando-se a viscosidade newtoniana equivalente. Neste último a equação que fornece tal viscosidade passa a ser dada por:

$$\mu_a = \mu_p + 5 \frac{\tau_0 (D_2 - D_1)}{\bar{v}} \quad (83)$$

C) Modelo de Lei de Potências

As correlações para o cálculo da perda de carga durante o escoamento turbulento de fluidos que obedecem à lei de potências foram desenvolvidas por Dodge e Meltzner, através de estudos realizados em tubos circulares.

i) *Escoamento ao longo da coluna*

A exemplo do critério de turbulência baseado em uma viscosidade newtoniana equivalente, aplicada em fluidos binghamianos, foram desenvolvidas correlações que sejam capazes de traduzir o comportamento de um fluido não-newtoniano como um fluido newtoniano, a partir de uma viscosidade equivalente que seja capaz de representar o mesmo.

Dessa forma, ao calcularmos tal parâmetro poderemos proceder da mesma maneira que proposto na seção anterior, calculando-se o número de Reynolds, comparando-se com um determinado valor crítico e aplicando-se a correlação correta para o coeficiente de fricção. A única ressalva, no entanto, está na correlação que fornecerá tal parâmetro, que, em unidades de campo para geometria circular é dada por.

$$\mu_a = \frac{K}{96} \left(\frac{ID}{\bar{v}} \right)^{(1-n)} \left(\frac{3 + 1/n}{0,0416} \right)^n \quad (84)$$

Entretanto a utilização desse critério, a exemplo do que acontece com fluidos plásticos de Bingham, fornecem valores inconsistentes para escoamento turbulento quando as aplicamos na função de Colebrook (Eq. 73). Nesse sentido Dodge e Meltzner propuseram uma função empírica para o coeficiente de fricção em fluidos que obedecem a lei de potências, escoando em superfície lisa no regime turbulento.

$$\frac{1}{\sqrt{f_{turb}}} - \frac{4}{n^{0,75}} \log(Re f^{1-n/2}) + \frac{0,395}{n^{1,2}} = 0 \quad (85)$$

Para escoamento em regime laminar a Eq. 74 continua válida para aplicação na equação de Fanning (Eq. 72) (Bourgoyne *et al.* 1986).

Para esses fluidos o valor crítico do número de Reynolds acima do qual o escoamento pode ser considerado turbulento passa a ser função do índice de consistência do fluido, n . Esse limite do número de Reynolds pode encontrado no gráfico representado na Figura 30, além das respectivas curvas do fator de fricção em função do Re e do parâmetro n , segundo a função empírica de Dodge e Metzner (Eq. 85). Interessante observar que para $n = 1$, fluido tipicamente newtoniano, o limite do escoamento laminar continua sendo 2100, como apresentado nas seções anteriores.

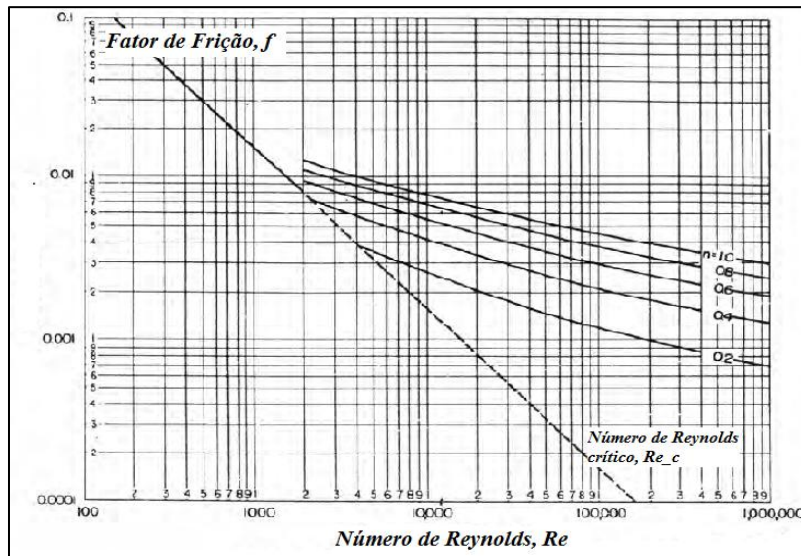


Figura 30 – Número de Reynolds crítico e fator de fricção para modelo de lei de potências (Bourgoyne *et al.* 1986, adaptado)

iii) Escoamento ao longo do anular

Para fluidos que obedecem a lei de potências, assim como para os demais, o cálculo da perda de carga ao longo de uma geometria diferente da circular é realizada baseada nas mesmas equações, apenas corrigindo-se o diâmetro para um diâmetro circular equivalente.

Para o caso desses fluidos, devemos, além de corrigir a geometria, aplicar outra expressão para a viscosidade newtoniana equivalente, que, em unidades de campo, será dada por:

$$\mu_a = \frac{K}{144} \left(\frac{D_2 - D_1}{\bar{v}} \right)^{(1-n)} \left(\frac{2 + 1/n}{0,0208} \right)^n \quad (86)$$

3.3. Modelagem do Comportamento de Brocas

A taxa de penetração é um dos parâmetros mais importantes na perfuração de poços, uma vez que esta tem relação direta com o tempo gasto perfurando (*PT*), bem como o custo final daquela atividade. Como colocado no capítulo 1 desse trabalho a maximização da taxa de penetração é sempre visada, como forma a minimizar os custos associados a mesma. Nesse sentido uma série de modelos de estimação do *ROP* já foram *propostos* na literatura, considerando diferentes tipos de broca, bem como uma gama grande de parâmetros que afetariam tal taxa.

Segundo Bourgoyne *et al.* (1986) os principais parâmetros seriam: (i) tipo de broca, (ii) características da formação, (iii) propriedades dos fluidos de perfuração, (iv) condições operacionais, como peso sobre a broca e rotação, (v) desgaste da broca e (vi) hidráulica da broca. Ao longo dessa seção focaremos o estudo em brocas com partes móveis, mais especificamente as tricônicas, baseado no modelo de perfuração *proposto* para tais brocas e a análise de como cada um desses parâmetros afeta a taxa de penetração.

3.3.1. Introdução

Toda a estimação da taxa de penetração deve começar com uma análise do tipo de broca, a maneira como ela atua e os parâmetros que influenciam no seu desempenho. Atualmente na indústria de petróleo existem diversos tipos de brocas, que podem ser divididas, basicamente, em dois grandes grupos: brocas sem partes móveis e brocas com partes móveis (Thomas *et al.* 2002).

As brocas com partes móveis (*rolling cutter bits*) são caracterizadas, tipicamente, pela existência de uma estrutura cortante e rolamentos, que não existem nas brocas sem partes móveis. As mesmas podem ter entre um e quatro cones, sendo mais comum a utilização de brocas com três (Thomas *et al.* 2002). Esses cones giram ao redor de um eixo de rotação conforme os cortadores giram no fundo. Segundo Bourgoyne *et al.* (1986) a ampla variedade

de designs possíveis para essas brocas permitem que a mesma seja adaptada para perfurar formações com características variadas. A Figura 31 apresenta duas das principais brocas tricônicas na indústria atualmente: com insertos de tungstênio e insertos de aço.

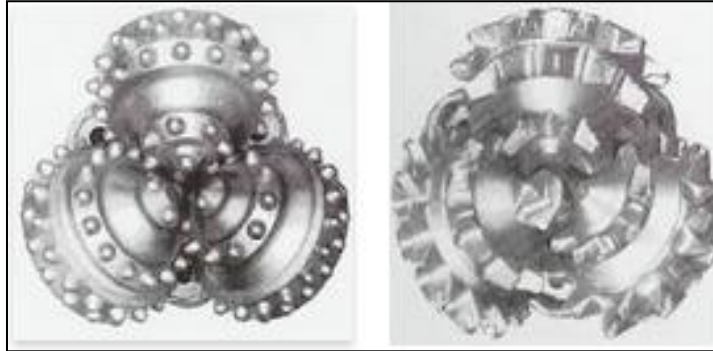


Figura 31 – Brocas tricônicas com insertos de tungstênio e aço, respectivamente (Thomas *et al.*, 2002)

3.3.2. Mecanismos de Fraturamento durante a Perfuração

De acordo com Bourgoyne *et al.* (1986) as brocas com partes móveis usualmente perfuram segundo o mecanismo de percussão, também conhecido como mecanismo de crateração. Este é governado por diversos parâmetros, como veremos na seção seguinte. Há também uma diferença consistente na perfuração de formações dúcteis e frágeis, como veremos a seguir. A compreensão dos mecanismos de fraturamento da formação é fundamental para o desenvolvimento e discussão acerca de modelos de perfuração. Poderemos ver na Figura 32 todas as etapas do mecanismo de perfuração para brocas com partes móveis, explicados em seguida.

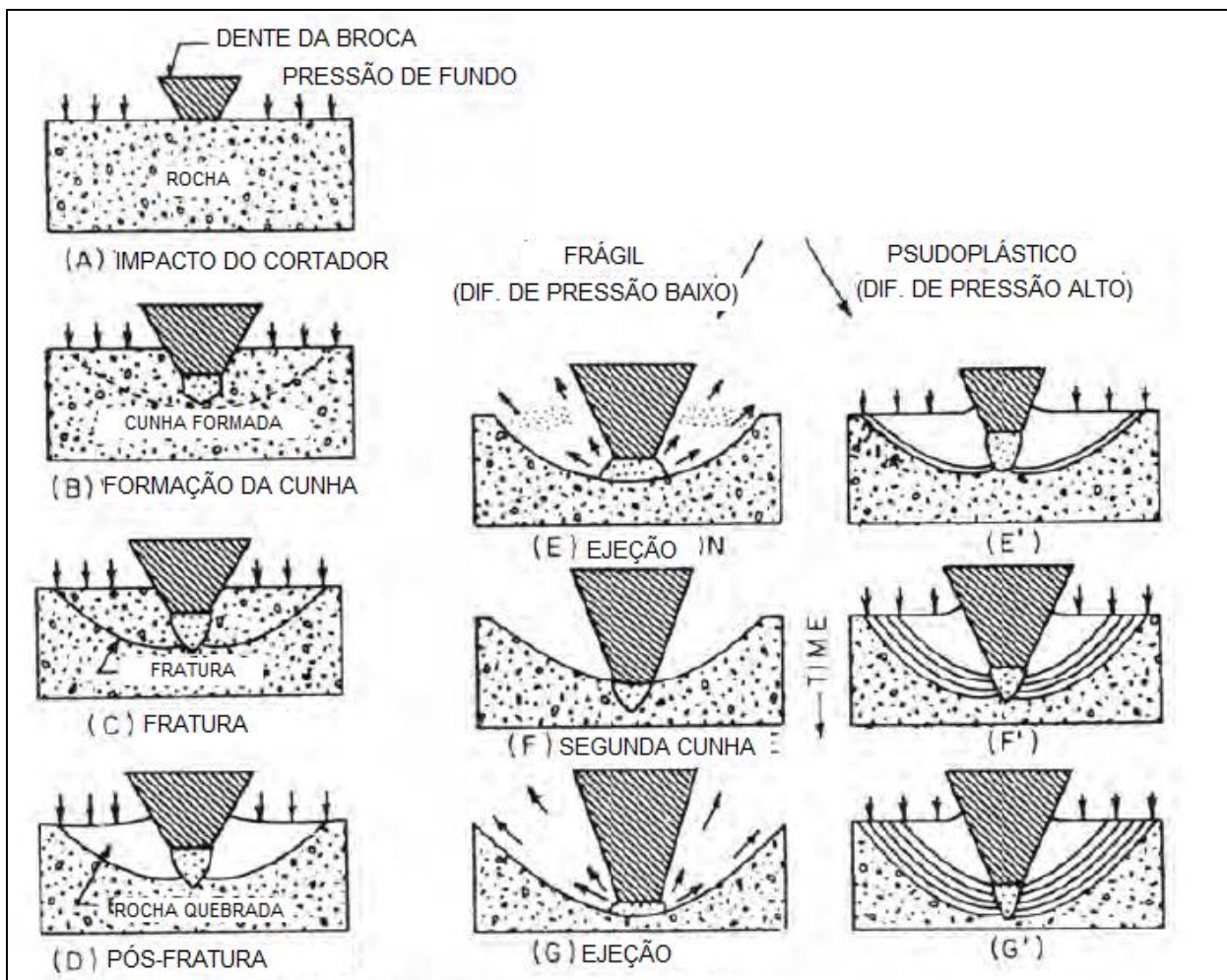


Figura 32 – Mecanismo de perfuração em brocas com partes móveis (Bourgoyne et al. 1986, adaptado)

Segundo Bourgoyne *et al.* (1986) o início da perfuração ocorre quando os cortadores (ou dentes) da broca penetram na formação devido a uma carregamento vertical, dado pelo peso sobre a broca (Figura 32.A). Em seguida a pressão exercida pela broca sobre a formação tende ao aumentar, até o ponto em que ultrapassa o limite de resistência da rocha (CCS⁴). Nesse ponto uma cunha é formada abaixo do dente (Figura 32.B). Todavia essa cunha só é efetivamente “quebrada” quando a força de compressão exercida pelo dente aumenta ao ponto em que as forças laterais induzidas pela cunha nas formações adjacentes atingirem o limite de resistência ao cisalhamento da rocha. Nesse ponto ocorre a fratura ao longo do plano da cunha criada (Figura 32.C). Tais fraturas se propagam ao longo do plano de

⁴ CCS = Confined Compressive Strength

cisalhamento máximo. O fraturamento tem início quando a força vertical atinge a chamada força limitante. Conforme a força aumenta acima da força limitante fraturas subsequentes ocorrem na região acima da primeira fratura, formando uma zona de rocha fraturada (Figura 32.D).

A partir daí a característica frágil ou dúctil da rocha influencia diretamente na penetração. Zonas submetidas a diferenciais de pressão⁵ elevados comportam-se pseudoplasticamente, enquanto que diferenciais de pressão baixos levam a comportamentos frágeis. Para zonas de comportamento frágil a zona de rocha fraturada é facilmente retirada da formação, em forma de cascalhos (Figura 32.E). Os cortadores da broca então seguem adiante, até atingirem a fundo da cratera, e o processo, é então, reiniciado (Figura 32.F e Figura 32.G). Para rochas com comportamento pseudo-plástico o diferencial de pressão, bem como as forças de fricção dificultam a retirada dos cascalhos (Figura 32.E'). Dessa forma, conforme a broca continua perfurando a rocha absorve energia, deformando plasticamente, e formando as chamadas crateras pseudo-plásticas (Figura 32.F' e Figura 32.G'). Rochas submetidas a esse comportamento demandam uma maior força de compressão para uma mesma penetração, se comparado a zonas de comportamento frágil (Bourgoyne *et al.* 1986).

3.3.3. Modelo de Perfuração – Estimação do ROP

Atualmente, na literatura, existem diversos modelos para estimação da taxa de penetração tanto de brocas sem partes móveis quanto brocas com partes móveis. Para brocas tricônicas, de uma forma geral, tais modelos assumem a taxa de penetração como função do peso sobre a broca, desgaste dos cortadores, rotação, entre outros. Nesse sentido Bourgoyne & Young propuseram em 1974 um modelo de perfuração onde a taxa de penetração dessas brocas é função de efeitos independentes de oito relações funcionais representativas de diferentes variáveis de perfuração, correlacionadas pelo produto:

$$ROP = (f_1) (f_2) (f_3) (f_4) (f_5) (f_6) (f_7) (f_8) \quad (87)$$

No modelo de Bourgoyne-Young tais relações funcionais – f_1, f_2, \dots, f_8 – são dadas pelas equações abaixo.

⁵ Diferencial de pressão (Δp) é a diferença entre a pressão de fundo do poço e a pressão de poros

$$f_1 = \exp (2,303a_1) = K \quad (88)$$

$$f_2 = \exp [2,303a_2(10.000 - Z)] \quad (89)$$

$$f_3 = \exp [2,303a_3Z^{0,69}(G_p - 9,0)] \quad (90)$$

$$f_4 = \exp [2,303a_4Z(G_p - ECD)] \quad (91)$$

$$f_5 = \left[\frac{\left(\frac{WOB}{D_b}\right) - \left(\frac{WOB}{D_b}\right)_t}{4 - \left(\frac{WOB}{D_b}\right)_t} \right]^{a_5} \quad (92)$$

$$f_6 = \left(\frac{N}{60}\right)^{a_6} \quad (93)$$

$$f_7 = \exp (-a_7h) \quad (94)$$

$$f_8 = \left(\frac{F_j}{1000}\right)^{a_8} \quad (95)$$

Nessas formulações temos que:

- a_1, \dots, a_8 parâmetros de calibração que devem ser estimados baseado nas condições locais de perfuração;
- ECD densidade equivalente de circulação, em *ppg*;
- Z profundidade do poço, em *ft*;
- WOB peso sobre a broca, em *lbf*;
- D_b diâmetro de broca, em *in*;
- N rotação da broca, em *rpm*;

- h parâmetro de desgaste dos dentes da broca, adimensional;
- $\left(\frac{WOB}{D_b}\right)_t$ força mínima necessária para iniciar o fraturamento da formação, em lbf/in ;

Os coeficientes a_1, \dots, a_8 são parâmetros que visam ajustar o desempenho da broca tricônica em questão ao perfurar uma determinada rocha submetida a determinadas condições de subsuperfície. Tais coeficientes podem ser calculados tomando como base dados de perfuração obtidos em poços de correlação, e serem atualizados baseado em dados advindos a perfuração do próprio poço, em tempo real. Além de fornecer uma estimacão da taxa de perfuração ao longo do poço o modelo de perfuração também é aplicado para diagnosticar alteração na pressão de poros da formação perfurada. É importante frisar que o modelo de Bourgoyne-Young não é capaz de modelar determinados fenômenos, como enceramento de broca, aprisionamento de broca, entre outros. A Figura 33 apresenta a sensibilidade do ROP frente alteração dos parâmetros do modelo.

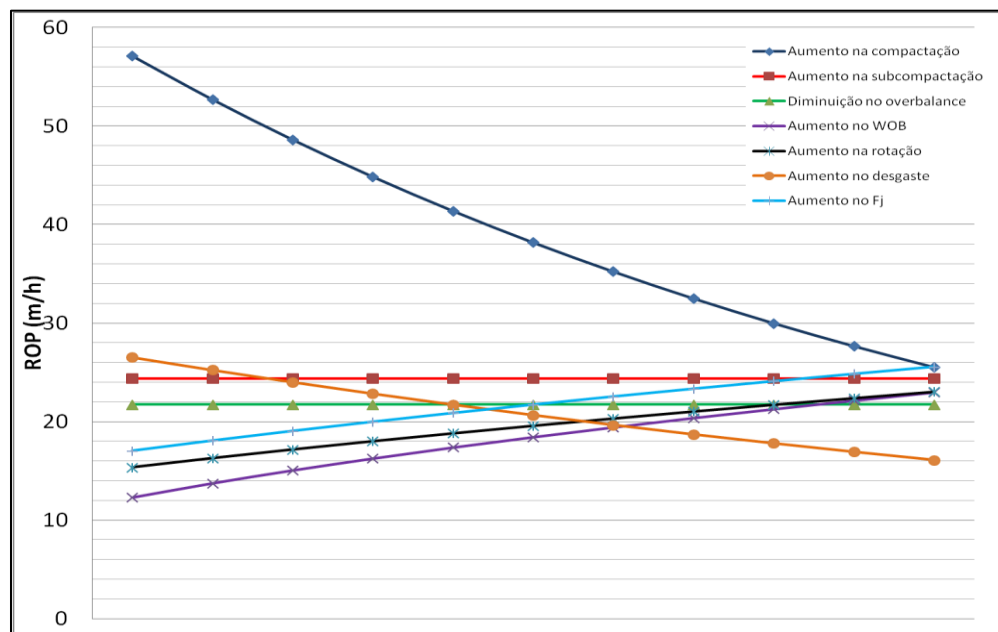


Figura 33 – Sensibilidade do ROP aos diversos parâmetros da perfuração, no modelo de Bourgoyne (elaboração própria)

3.3.3.1. Efeito da resistência da formação, projeto de broca, entre outros

De uma forma geral a relação funcional f_1 visa englobar, através de uma constante, diversos parâmetros de perfuração que não foram abordados no modelo de perfuração, como resistência de formação, projeto da broca, tipo de fluido de perfuração, conteúdo de sólidos, entre outros. Nesse sentido, alteração em algum desses parâmetros deve levar a um impacto sobre o coeficiente a_1 , para, conseqüentemente, ajustar a relação funcional f_1 bem como taxa de penetração.

3.3.3.2. Efeito da Compactação da Rocha

As relações f_2 e f_3 visam representar o impacto da compactação da rocha sobre a taxa de penetração. A relação f_2 tem por objetivo modelar o acréscimo da compactação da rocha com a profundidade, bem como a contribuição negativa da mesma frente a taxa de penetração. Foi considerada uma profundidade de referência de 10000 *ft*, como podemos observar na relação. Esta relação não esta considera a presença de zonas subcompactadas ou sobrecompactadas. Tais zonas deverão ser modeladas na relação f_3 .

A presença de zonas subcompactadas ou sobrecompactadas é observada a partir de dados de pressão de poros da formação. O modelo mostra que a relação entre a subcompactação e a taxa de penetração é positiva, sendo coerente com a relação funcional f_2 . O gradiente de poros de referência, considerado gradiente de poros normal, é de 9,0 *ppg*.

3.3.3.3. Efeito do Overbalance

A relação funcional f_4 visa modelar o efeito do diferencial de pressão entre a pressão de poros da formação e a pressão do fundo do poço com a taxa de penetração. Essa relação é positiva, ou seja, quanto maior a pressão de poros frente o *BHP* maior será o *ROP*. Isso ocorre, pois, como explicado na seção 3.3.2, diferenciais de pressão menores levam a comportamento frágil da formação, e, conseqüentemente, penetrações maiores quanto mantidas todas as demais variáveis de perfuração constantes.

3.3.3.4. Efeito do Peso sobre a Broca

A relação funcional f_5 visa modelar o efeito do peso sobre a broca frente à taxa de penetração. Podemos observar no modelo em questão que não é o peso sobre a broca que é

considerado para a estimação, mas sim o peso por unidade de diâmetro da mesma. Essa relação é cuidadosa no sentido de garantir que para qualquer peso que seja inferior ao limite de resistência da formação não haverá penetração, já que não teremos o fraturamento da cunha, como apresentado na seção 3.3.1. Ao mesmo tempo, conforme a força vertical aumenta frente ao limite de resistência da rocha, a contribuição da mesma para a perfuração cresce potencialmente com expoente α_5 . Tal expoente assume, normalmente, valores entre 0,5 e 2,0, dependendo das condições de perfuração.

Segundo Bourgoyne *et al.* (1986) o limite de resistência da formação costuma ser consideravelmente pequeno, podendo ser negligenciado em vários casos, como no Golfo do México por exemplo. Nesse modelo a diferença entre o peso sobre a broca e o limite de resistência da formação é normalizado pela diferença entre um peso sobre a broca de referência, comumente igual a 4000 lbf/in , e o limite de resistência da rocha.

O peso sobre a broca é dado pelo peso do BHA, descontado o fator de empuxo. Entretanto, como é fundamental que a linha neutra da coluna passe pelos comandos, já que os drill pipes devem atuar sobre tração, devemos projetar o BHA com um coeficiente de segurança tal que garanta que o centro de gravidade esteja posicionado no BHA. Dessa forma podemos equacionar que o peso sobre a broca seria dado por:

$$WOB = (n_{hwdp}w_{hwdp}L_{hwdp} + n_{dc}w_{dc}L_{dc})(FS)\alpha \quad (96)$$

Nessas formulações temos que:

- n número de seções de cada componente do *BHA*;
- w peso específico da cada componente do *BHA*, em lbf/ft ;
- L comprimento de cada seção dos componentes, em ft ;
- FS coeficiente de segurança, normalmente igual a 0,7.

3.3.3.5. Efeito da Rotação da Broca

A relação funcional f_6 tem como objetivo modelar a influência da rotação da broca para com a taxa de perfuração. A relação proposta nesse modelo de perfuração afirma que o impacto da rotação no ROP varia potencialmente com expoente a_6 , que varia usualmente entre 0,4 e 1,0. Nesse modelo a rotação é normalizada para por uma rotação de referência, assim como existe a figura do peso sobre a broca de referência na relação funcional f_5 . O modelo de Bourgoyne-Young sugere uma rotação de referência igual a 60 rpm .

A rotação sobre a broca pode ter diferentes origens, dependendo do tipo de perfuração que é aplicado. A perfuração com mesa rotativa e top drive, que levam a rotação de toda coluna, acarretam na perda de torque, de forma que a broca terá uma velocidade de rotação menor que aquela impressa na superfície. A implementação de motores de fundo também podem ser aplicados como forma a aumentar a rotação da broca.

3.3.3.6. Efeito do Desgaste da Broca

A relação funcional f_7 objetiva equacionar a influência do desgaste da broca sobre a taxa de penetração. Para a modelagem desse desgaste existem duas possíveis abordagens: o desgaste dos cortadores e o desgaste dos rolamentos. Segundo Bourgoyne *et al.* (1986) o primeiro é mais simples de ser estimado, motivo pelo qual focaremos neste ao longo desse trabalho.

A equação global para desgaste instantâneo nos cortadores da broca pode ser obtida a partir da combinação dos efeitos da geometria dos dentes, peso sobre a broca e rotação. A partir de a equação a seguir seremos capazes de derivar o desgaste nos dentes.

$$\frac{dh}{dt} = \frac{1}{J_2 \tau_H} \left(\frac{1}{1 + H_2 h} \right) \quad (97)$$

$$J_2 = \left(\frac{60}{N} \right)^{H_1} \left[\frac{\left(\frac{WOB}{D_b} \right)_m - \left(\frac{WOB}{D_b} \right)}{\left(\frac{WOB}{D_b} \right)_m - 4} \right] \left(\frac{1}{1 + H_2/2} \right) \quad (98)$$

Nessas formulações temos que:

- τ_H constante de abrasividade da formação, em horas;
- $\left(\frac{WOB}{D_b}\right)_m$ máximo peso por polegada de diâmetro da broca na qual esta falhará instantaneamente, ou seja, $\frac{WOB}{D_b} < \left(\frac{WOB}{D_b}\right)_m$, em 1000 *lbf/in*;
- H_1, H_2 parâmetros da broca, bem como o $\left(\frac{WOB}{D_b}\right)_m$;
- t tempo com a broca perfurando, em horas.

A partir daí podemos estimar o valor do desgaste da broca a integração da Eq. 97, em função do tempo e do desgaste da broca, ou seja:

$$\int_0^t dt = J_2 \tau_H \int_0^h (1 + H_2 h) dh \quad (99)$$

$$\frac{J_2 \tau_H H_2}{2} h^2 + J_2 \tau_H h - t = 0 \quad (100)$$

Dessa forma, resolvendo-se a equação de segundo grau em questão (Eq. 100) obteremos o desgaste dos cortadores da broca, que será assumido como sendo o desgaste da própria broca. Ou seja, quando h atingir 100% significaria dizer que a broca atingiu seu limite de corte. O parâmetro h alimentará o modelo de perfuração através da relação funcional f_7 . O parâmetro de calibração desse modelo costuma variar entre 0,3 e 1,5, dependendo das condições locais de perfuração.

3.3.3.7. Efeito da Hidráulica da Broca

A relação funcional f_8 objetiva modelar a influência da hidráulica da broca na taxa de perfuração. Segundo Bourgoyne *et al.* (1986) a escolha desse parâmetro é arbitrário, pois resultados muito semelhantes são conseguidos utilizando-se outros parâmetros hidráulicos, como potência hidráulica de fundo, por exemplo. Essa relação, a exemplo das relações f_5 e

f_6 , apresentam a variável estudada normalizada por um valor de referência, que é de 1000 lbf . A relação apresentada prevê que haverá uma variação potencial de expoente a_8 entre a taxa de penetração e a grandeza da força de impacto hidráulico. Esse expoente variaria entre 0,3 e 0,6.

4. Metodologia aplicada no Módulo de Planejamento da Perfuração

4.1. Introdução

Conforme apresentado na seção 1.2, o objetivo do presente trabalho é desenvolver um simulador computacional que seja capaz de provisionar um projeto de poço, contendo todos os resultados citados na seção 1.1.2, que será desenvolvido através do fluxo de trabalho do módulo de planejamento de poço e, em seguida, permitir que o usuário simule a atividade de perfuração daquele mesmo poço, alterando os parâmetros de entrada, como peso sobre a broca, rotação, propriedades dos fluidos de perfuração, entre outras, acompanhando o desempenho da mesma através de diversos parâmetros de saída, como *ROP*, *BHP*, entre outros, vide seção 1.1.2. Este será provisionado no módulo de perfuração em tempo real. No entanto vale lembrar que tais módulos não são independentes, já que boa parte dos resultados obtidos ao longo do módulo de planejamento serão aplicados como entradas no módulo de perfuração em tempo real, como veremos no decorrer desse trabalho.



Figura 34 – Tela inicial do simulador

Dessa forma esse capítulo visa apresentar toda a metodologia por trás do primeiro módulo. Quais as simplificações assumidas, quais as variáveis de entrada do módulo, como

ele processa tais informações como forma de fornecer os resultados desejados e, principalmente, como os usuários devem atuar no processo, de forma a obter os resultados desejados.

Em linhas gerais o módulo de planejamento da perfuração pode ser caracterizado como um módulo computacional pseudo-interativo, uma vez que a única atuação do usuário frente a metodologia desenvolvida por ele está na escolha das variáveis de entrada, que denominaremos de *inputs*, bem como na escolha de algumas metodologias diferentes que foram implementadas no simulador, como, por exemplo, optar que a pressão de poros deva ser calculada pelo método de Eaton, pelo método de Bowers ou simplesmente carregada de um arquivo externo.

O módulo supracitado é composto então por um total de 14 blocos, como podemos ver na Figura 35 abaixo. Cada um desses blocos é responsável por uma parte do módulo

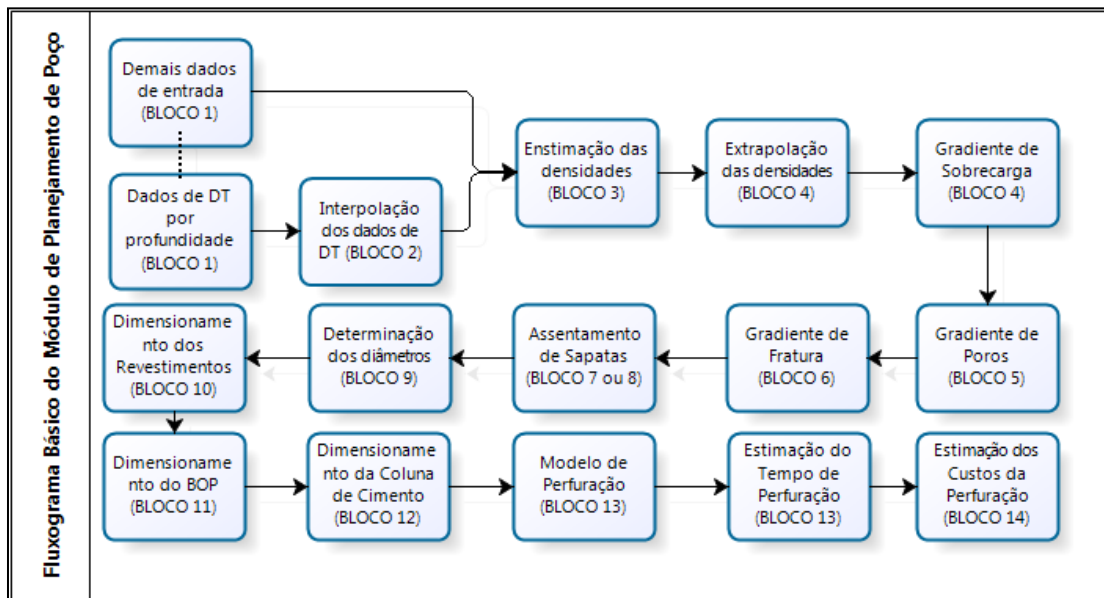


Figura 35 – Fluxograma básico do módulo de planejamento de poço (elaboração própria)

Na Figura 36 podemos observar a tela inicial do módulo de planejamento de poço do simulador de perfuração supracitado. Vale observar que a tela permite ao usuário a escolha seqüencial das opções listadas. É importante que todas as opções sejam escolhidas na ordem em que estão apresentadas, uma vez que os cálculos realizados em um bloco normalmente

são dependentes daqueles realizados em blocos anteriores, como se pode ter idéia pelo fluxograma acima. Cada um dos blocos presentes será apresentado mais profundamente no decorrer desse capítulo.

```
*****
*          MODULO DE PLANEJAMENTO DE POCO - UFRJ DRILLING SIMULATOR
*
*          MENU INICIAL:
*
*          PRESS 1> DEFINICAO DE PARAMETROS
*          PRESS 2> INTERPOLACAO DOS DADOS
*          PRESS 3> DEFINICAO DAS DENSIDADES
*          PRESS 4> EXTRAPOLACAO DAS DENSIDADES
*          PRESS 5> DEFINICAO DO GRADIENTE DE POROS E SOBRECARGA
*          PRESS 6> DEFINICAO DO GRADIENTE DE FRATURA
*          PRESS 7> PROJETO DE POCO - CRITERIO DE GRADIENTE
*          PRESS 8> PROJETO DE POCO - CRITERIO DE MARGEM DE RISER
*
*          OPCAO:
```

Figura 36 – Tela inicial do módulo de planejamento de poço do simulador *proposto*.

Todas as demais funcionalidades desse módulo serão desenvolvidas independentemente da atuação do usuário, e seguirão o fluxo apresentado na Figura 35 após ser definido as profundidades do assentamento das sapatas pela aplicação do Bloco 7 ou Bloco 8. A partir daí até o Bloco 14 toda a interação será feita pelo próprio simulador.

4.2. Características Gerais do Simulador

Tudo que for referente ao simulador – considerando ambos os módulos – está presente na pasta UFRJDRILLINGSIMULATOR. O conteúdo dessa pasta é composto de um arquivo executável UFRJ DRILLINGSIMULATOR.EXE, que corresponde ao simulador em si e cinco bibliotecas de C++ que devem ser carregadas na posteriormente na respectiva pasta DEV-CPP/INCLUDE. Cada uma dessas bibliotecas será apresentada posteriormente.

Além das bibliotecas e do executável do simulador a pasta UFRJDRILLINGSIMULATOR também contém três outras pastas. A primeira, UFRJDRILLINGSIMULATOR/INPUTS, contém todos os arquivos texto que serão, em algum momento, lidos pelo programa para o desenvolvimento do projeto do poço nos quais o usuário tem acesso. Em outras palavras o

controle que o usuário tem do planejamento, como a entrada do perfil sônico, dados de calibração, custos de perfuração, dados de poços de correlação, entre vários outros, estarão nessa pasta.

A segunda pasta, UFRJDRILLINGSIMULATOR/DEBUG, contém dos os arquivos que o próprio simulador gerou e que serão lidas em um segundo momento ao longo do processo. Já a última pasta, UFRJDRILLINGSIMULATOR/OUTPUTS, contém os resultados do simulador, também para o módulo de planejamento quanto para o módulo de perfuração em tempo real. Neste serão armazenados os arquivos do relatório de planejamento, gráficos, relatório de perfuração, dados de acompanhamento da perfuração, entre vários outros.

No contexto do módulo de planejamento do poço os seguintes arquivos deverão estar, obrigatoriamente, presentes na pasta UFRJDRILLINGSIMULATOR/INPUTS. Todos os arquivos apresentam, abaixo das informações que eles contém, o que cada valor representa e se há alguma limitação de ordem numérica como, por exemplo, limite no número de dados de entrada, profundidades devem ser valores interiores, entre outros.

Nome do Arquivo	Descrição
InputMain.txt	Contém um pacote de dados genéricos, como densidades, RFTs, LOTs, entre outros
InputDT.txt	Contém o perfil sísmico ao longo do poço, além do tipo de litologia (folhelho ou arenito) e o mecanismo de sobrepressão
InputRHO.txt	Contém o perfil de densidade a ser carregada externamente, se for de opção do usuário
InputGP.txt	Contém o perfil de gradiente de poros a ser carregada externamente, se for opção do usuário
InputFCoefficient.txt	Contém dos parâmetros F para cálculo de colapso para vários graus de aço
InputStagesConfig.txt	Contém diversas combinações típicas de diâmetros das fase e dos revestimentos para construir o poço
InputFixedCosts.txt	Contém custos de 13 insumos de custo fixo na plataforma
InputVariableCosts.txt	Contém custos de 16 insumos de custo variável na plataforma
InputCasingPlacement.txt	Contém os dados de tempo e profundidade para o assentamento de revestimentos em um poço de correlação
InputBourgoyneParameters.txt	Contém os 8 coeficientes do modelo de perfuração de Bourgoyne-Young para diferentes litologias
InputBitRecord.txt	Contém dados de broca utilizados em poços de correlação ao longo da perfuração

Tabela 4 – Arquivos que devem estar presente na pasta UFRJDRILLINGSIMULATOR/INPUTS

4.3. Carregamento e Processamento dos Dados

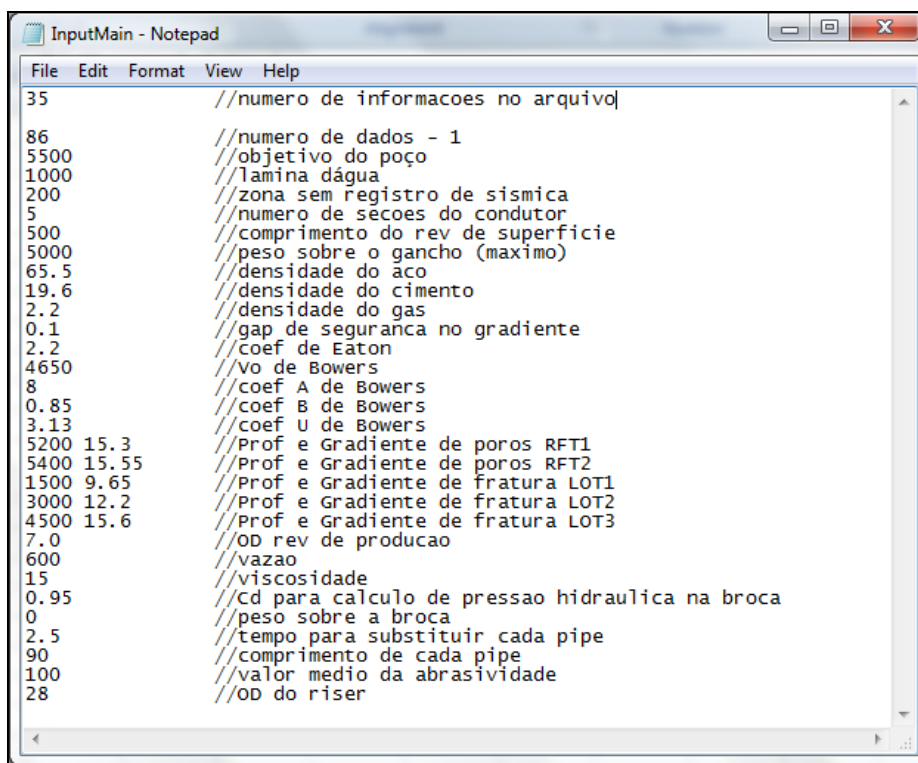
Após garantir que os arquivos de entrada listados acima estão corretamente preenchidos com os dados que usaremos para projetar o poço desejado o início do processo se dá de maneira simples. Abrindo o arquivo executável basta definir um nome para o poço, pressionar ENTER, selecionar a Opção 1 (MÓDULO DE PLANEJAMENTO DA PERFURAÇÃO). Na sequência a tela apresentada na Figura 34 deverá aparecer na tela. A esse ponto nenhum dado foi sequer armazenado no simulador. O carregamento desses dados será realizado ao longo do módulo, de acordo com a demanda do simulador, de maneira dinâmica.

Optou-se por esse tipo de armazenamento de dados uma vez que ao armazenar diversos perfis em estruturas matriciais de milhares de elementos o programa perdeu desempenho até que chegou a um ponto onde ele não foi mais capaz de rodar. A solução encontrada para esse problema foi o armazenamento dinâmico. Conforme o usuário, ou o módulo, muda de bloco, um novo arquivo é lido, seus dados armazenados, processados e finalmente descartados da memória.

4.3.1. Perfil Sísmico e outros Dados Básicos para Planejamento

Nessa fase do simulador o controle da escolha do módulo é toda do usuário. Cabe a ele a definição de qual bloco iniciar, mas, como já foi dito, é importante que até o Bloco 6 (DEFINIÇÃO DO GRADIENTE DE FRATURA) siga o fluxo natural.

Ao selecionar o Bloco 1 (DEFINIÇÃO DE PARÂMETROS) o programa estará armazenando a informação presente no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT, apresentado na Figura 37. Através desse o programa receberá informações pontuais importantes para seu fluxo de trabalho, como, por exemplo, profundidade objetivo do poço, lâmina d'água, espessura do trecho superficial sem sísmica, dados de RFT e LOT para calibração, coeficiente de Eaton, coeficientes de Bowers, entre outros, totalizando 35 dados pontuais referentes ao projeto.



```
35 //numero de informacoes no arquivo
86 //numero de dados - 1
5500 //objetivo do poço
1000 //lamina d'agua
200 //zona sem registro de sismica
5 //numero de secoes do condutor
500 //comprimento do rev de superficie
5000 //peso sobre o gancho (maximo)
65.5 //densidade do aco
19.6 //densidade do cimento
2.2 //densidade do gas
0.1 //gap de seguranca no gradiente
2.2 //coef de Eaton
4650 //vo de Bowers
8 //coef A de Bowers
0.85 //coef B de Bowers
3.13 //coef U de Bowers
5200 15.3 //Prof e Gradiente de poros RFT1
5400 15.55 //Prof e Gradiente de poros RFT2
1500 9.65 //Prof e Gradiente de fratura LOT1
3000 12.2 //Prof e Gradiente de fratura LOT2
4500 15.6 //Prof e Gradiente de fratura LOT3
7.0 //OD rev de producao
600 //vazao
15 //viscosidade
0.95 //Cd para calculo de pressao hidraulica na broca
0 //peso sobre a broca
2.5 //tempo para substituir cada pipe
90 //comprimento de cada pipe
100 //valor medio da abrasividade
28 //OD do riser
```

Figura 37 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT.

Além disso, ainda nesse bloco, teremos o armazenamento da informação contida no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTDT.TXT, que corresponde a todo o perfil sísmico ao longo do poço, discretizado em TVD^6 e com uma diferença na profundidade para cada dado constante. Além disso, esse arquivo também contém informações de litologia (1- folhelho, 2- arenito) e origem da sobrecarga (1- subcompactação ou compactação normal, 2- expansão de fluidos) também discretizados na. Esse último é relevante para a estimação da pressão de poros pelo método de Bowers, vide seção 2.2.4.3.

⁶ Profundidade vertical do poço referenciada a mesa rotativa

PROF	DT	LIT	ORIGEM	SOBREPRESSAO
1200	165	1	1	
1250	158	1	1	
1300	155	2	1	
1350	148	2	1	
1400	149	1	1	
1450	145	1	1	
1500	142	1	1	
1550	141	1	1	
1600	149	2	1	
1650	140	2	1	
1700	138	2	1	
1750	137	2	1	
1800	135	2	1	
1850	133	2	1	
1900	132	2	1	
1950	126	1	1	
2000	123	1	1	
2050	125	1	1	
2100	124	1	1	
2150	121	1	1	
2200	118	1	1	
2250	119	2	1	
2300	115	2	1	
2350	105	2	1	
2400	104	2	1	
2450	110	2	1	
2500	99	1	1	

Figura 38 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTDT.TXT

Na sequência o programa perguntará por repetir ou não a operação. A resposta negativa retornará a tela inicial (Figura 34), permitindo a escolha de um novo bloco. Caberá ao usuário a escolha ou não do Bloco 2 (INTERPOLAÇÃO DOS DADOS). Essa funcionalidade foi implementada para permitir que se tenha uma precisão maior nos cálculos desenvolvidos na sequência do simulador, já que o refinamento do perfil sísmico não é muito grande, vide seção 2.2.1. Caso o Bloco 3 (DEFINIÇÃO DAS DENSIDADES) seja escolhido o processo seguirá normalmente, entretanto o refinamento será o mesmo que presente no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTDT. A Figura 39 fornece um fluxograma básico do funcionamento dos blocos 1 e 2 do módulo de planejamento do poço.

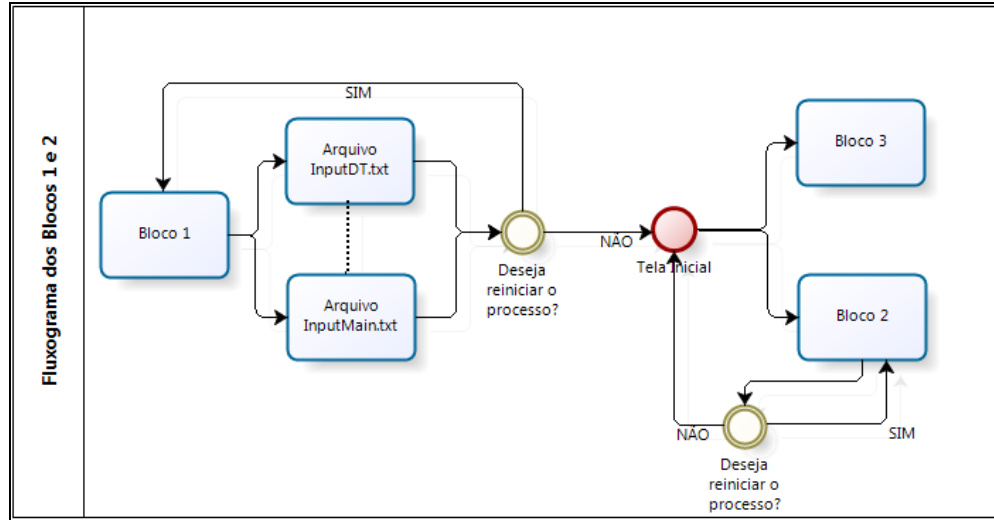


Figura 39 – Fluxograma dos blocos 1 do módulo planejamento do poço (elaboração própria)

Caso o usuário opte pelo Bloco 4 caberá a ele definir o intervalo de interpolação daquele perfil. Nesse ponto reside outra simplificação importante do simulador: o intervalo deve ser sempre valor inteiro e divisor do intervalo original do perfil, sendo que o intervalo de uma unidade de comprimento não é permitida por causar erros numéricos.

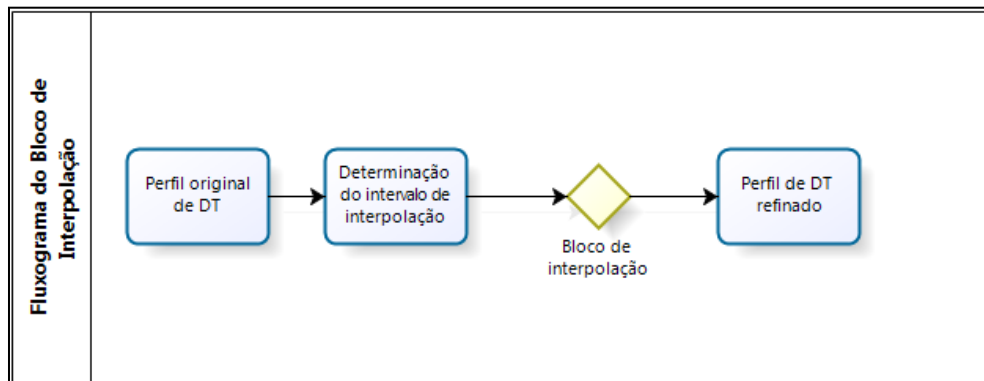


Figura 40 – Fluxograma do bloco 2 – interpolação do perfil de *DT* (elaboração própria).

4.3.2. Estimação e Extrapolação das Massas Específicas

4.3.2.1. Perfil de Massa Específica

O Bloco 3 (DEFINIÇÃO DAS DENSIDADES) deve, necessariamente, ser escolhido pelo usuário. Nesse bloco teremos a estimação ou o carregamento do perfil de densidade ao longo do poço, que será aplicado para calcularmos o gradiente de sobrecarga. Esse bloco foi desenvolvido com intuito de permitir ao usuário tanto carregar diretamente os dados de densidade discretizados ao longo do poço quanto utilizarmos o perfil sísmico previamente carregado no Bloco 1, convertendo para densidade a partir do método de Gardner (Eq. 1).

Ao optarmos pelo Bloco 3 o programa perguntará se o usuário deseja utilizar o método de Gardner ou inserir através de um arquivo texto. Se a opção for a segunda o programa armazenará como perfil de densidade os dados presentes no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTRHO.TXT. É importante frisar que, por limitação do simulador, o número de dados presente nesse arquivo deve ser o mesmo que aquele presente no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTDT.TXT.

Caso o usuário faça a opção por aplicar o método de Gardner este deverá fornecer ao bloco quais os coeficientes que o programa deverá adotar. Existe, no entanto, a possibilidade de o usuário lançar mão de um perfil de densidade ao longo de um trecho pequeno no poço para realizar a correta calibração do modelo de Gardner. Caso não seja de interesse basta definir os parâmetros a e b e o Bloco calculará o perfil de densidade. O programa só permite um valor para os parâmetros, o que pode levar a inconsistências na calibração ao longo de várias litologias. A Figura 41 apresenta um fluxograma do funcionamento do Bloco 3.

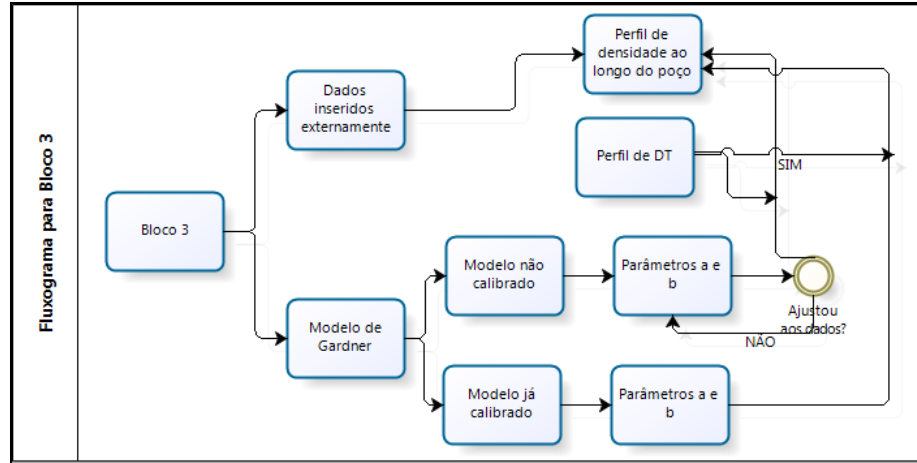


Figura 41 – Fluxograma do Bloco 3 do módulo de planejamento da perfuração (elaboração própria)

4.3.2.2. Perfis Extrapolados

Definido o perfil de massa específica ao longo do poço pode ser importante para o projeto que se tenha modelado o trecho superficial onde há ausência de dados sísmicos. Caso esse trecho seja pequeno não causará maiores problemas para a perfuração, já que será revestido com o condutor ou o revestimento de superfície. Entretanto, caso ele seja profundo, passará a ser significativo para perfuração. Nesse sentido o módulo permite ao usuário que este extrapole a curva de densidade segundo o modelo proposto por Traugott (Eq. 2). Essa funcionalidade está contida no Bloco 4 (EXTRAPOLAÇÃO DAS DENSIDADES). Esse bloco, como um todo apresenta duas funcionalidades: extrapolar a massa específica a partir da Eq. 2, e, a partir dessa informação, estimar o perfil sísmico a partir da equação de Gardner invertida, dada abaixo.

$$DT = 10^6 \sqrt[3]{\frac{\rho_b}{a}} \quad (101)$$

Caberá ao usuário definir os parâmetros a , b e c de calibração para Eq. 2, bem como os parâmetros a e b , para Eq. 101. O bloco também permitirá ao usuário uma determinação desses parâmetros de maneira automatizada, garantindo coeficientes que melhor ajustem a curva extrapolada com os dados do perfil. Esse mecanismo automatizado utiliza a

minimização da variância, sendo solicitado que o usuário defina qual a profundidade limite que ele deseja garantir o melhor ajuste. É importante que essa profundidade seja maior que o trecho superficial sem dados sísmicos, caso contrário teremos erros numéricos.

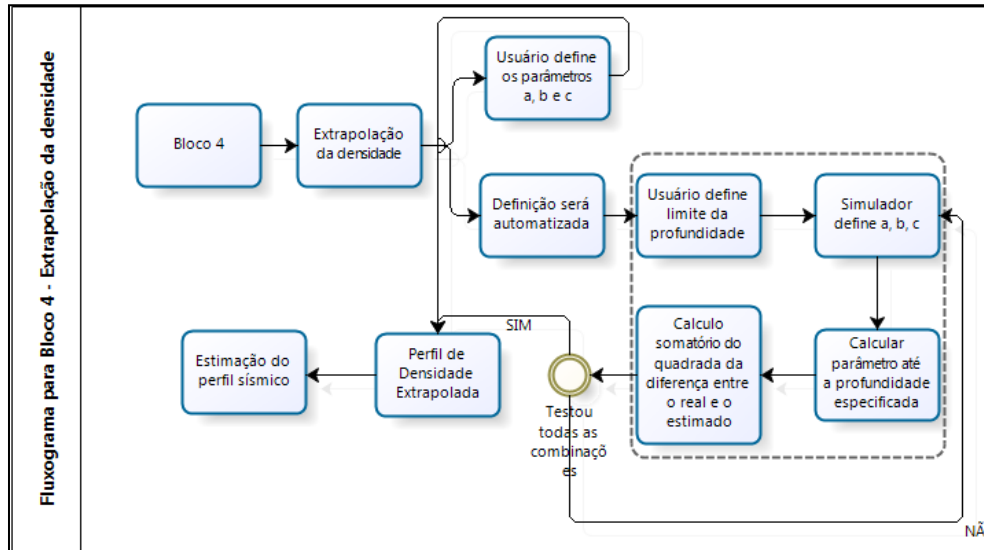


Figura 42 – Fluxograma para Bloco 4 do módulo de planejamento de poço, focando na extrapolação da massa específica das formações (elaboração própria)

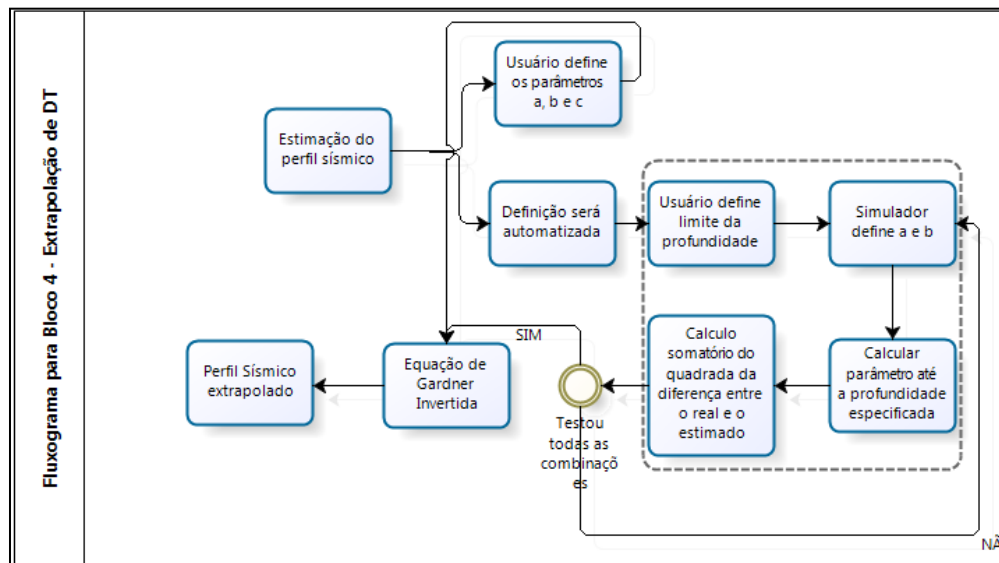


Figura 43 – Fluxograma para Bloco 4 do módulo de planejamento de poço, focando na extrapolação do perfil sísmico (elaboração própria)

4.3.3. Estimação das Geopressões

4.3.3.1. Gradiente de Sobrecarga

Após o cálculo do perfil de densidades extrapolado, bem como o perfil de DT , o simulador já tem armazenado ambos os perfis, que serão utilizados para o cálculo dos gradientes, conforme a Figura 35.

Iniciando o Bloco 5 (DEFINIÇÃO GRADIENTE DE POROS E SOBRECARGA) o simulador calculará o gradiente de sobrecarga ao longo da trajetória do poço. Como o simulador já tem armazenados os dados de densidade (Bloco 3 e 4) e outros adicionais, como lâmina d'água (oriundos do arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT) o bloco calcula a pressão, e, posteriormente o gradiente, aplicando respectivamente as equações 4 e 5. Esses dados serão exportados posteriormente pelo programa, como forma de arquivo texto.

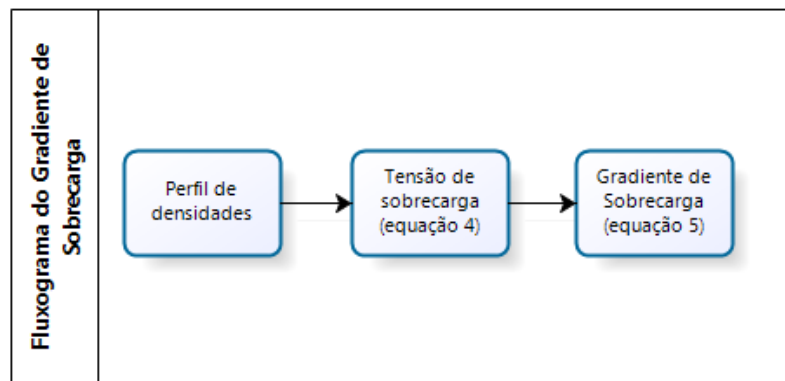


Figura 44 – Fluxograma para o cálculo do gradiente de sobrecarga (elaboração própria)

4.3.3.2. Gradiente de Poros

Após a definição do perfil de gradiente de sobrecarga o bloco desenvolverá a estimação de pressão de poros, conforme prevê o fluxograma básico deste módulo. Nessa etapa o simulador permite ao usuário a opção de estimar a pressão de poros a partir do perfil sísmico pelos métodos de Eaton e pelo método de Bowers, ou carregar o perfil a partir de um arquivo externo, a exemplo do perfil de densidades, no bloco 3.

Caso o usuário opte por carregar diretamente de um arquivo externo o programa armazenará os dados e, a partir dele, seguirá o fluxograma. Caso o usuário prefira a estimação por algum método geomecânico (Eaton ou Bowers) diferentes fluxos de trabalhos serão implementados.

```
*****
* DETERMINE A ENTRADA DOS GRADIENTES DE POROS
*
* PRESS 1) MODELOS GEOMECANICOS
* PRESS 2) INSERIR POR ARQUIVO .TXT
*
* OPCAO:
```

Figura 45 – Tela de escolha de como obter os dados do gradiente de poros

```
*****
* DETERMINE O MODELO GEOMECANICO DESEJADO
*
* PRESS 1) MODELO DE EATON
* PRESS 2) MODELO DE BOWERS
*
* OPCAO:
```

Figura 46 – Tela de escolha do método a ser aplicado

A estimação segundo o modelo de Eaton devera seguir os princípios descritos na seção 2.2.4.3. Entretanto caberá ao usuário definir a curva de tendência na qual o modelo de Eaton se baseia. O simulador permite 3 diferentes formas de fazê-lo. A primeira é a simples definição de pontos (Z, DT) quaisquer no perfil. O programa obterá a curva a partir desses dois pontos de maneira direta, bem como o parâmetro m , obtido pela Eq. 8.

O usuário também terá a possibilidade de estimar a curva de tendência de maneira automatizada. Nesta existem dois critérios possíveis para obtenção da curva: (i) otimização do ajuste da curva de tendência com os dados de calibração (RFTs ou FPWD) e (ii) otimização do ajuste da curva com o próprio perfil de compactação normal. Em ambos o usuário deverá definir quais os limites do trecho de compactação normal, e o simulador testará diversas combinações de pontos até encontrar aquele que garantirá o melhor ajuste

desejado. Vale observar que o simulador só considerará como pontos para o cálculo da curva de tendência aqueles que forem argilas, dado que o mecanismo de Eaton só é válido para esta litologia.

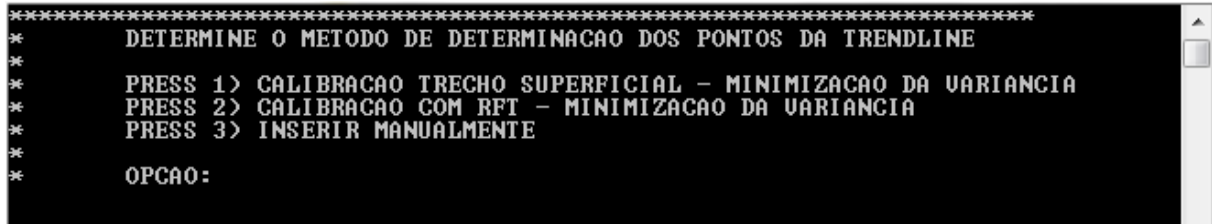


Figura 47 – Tela de escolha dos pontos da curva de tendência no método de Eaton

No caso de a comparação ser feita com os dados de RFT estes dados devem ser carregados no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT, conforme orientado no próprio arquivo. O simulador só é capaz de lidar com dois valores de RFT.

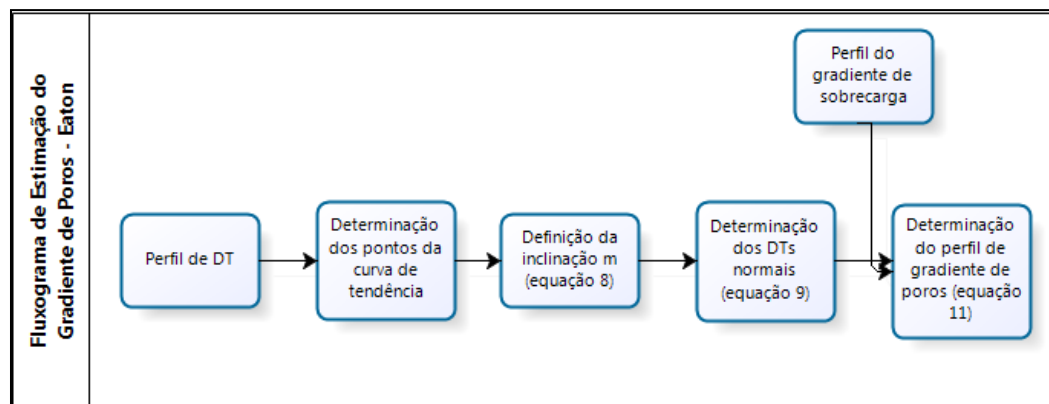


Figura 48 – Fluxograma para estimação da pressão de poros, segundo Eaton (elaboração própria)

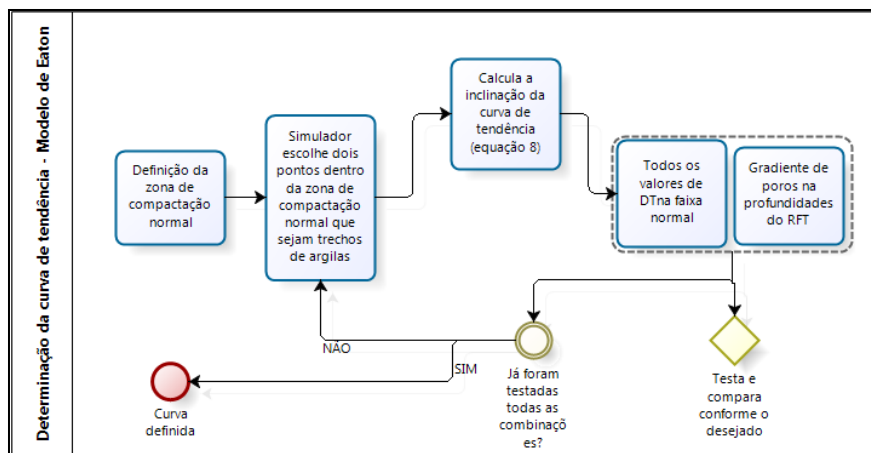


Figura 49 – Fluxograma para definição da curva de tendência para o modelo de Eaton (elaboração própria)

Caso o usuário opte por estimar a pressão de poros a partir do modelo de Bowers este deverá fornecer uma informação básica, que é se há sobrepressão e, caso haja, qual a origem desta. Essa informação deverá constar no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTDT.TXT, onde 1 corresponde a ausência e/ou origem por subcompactação e 2 corresponde a origem por expansão de fluidos. A partir daí devem ser definidos os parâmetros A, B, U e V_0 do modelo. O usuário poderá defini-los de maneira direta ou automatizada, buscando minimizar a diferença entre os valores do gradiente estimados e os dados de calibração. Conforme apresentado na seção 2.2.4.3, se, em um determinado ponto tivermos subcompactação ou compactação normal esta deverá ser calculada pela equação 16. Caso tenha sido diagnosticada a expansão de fluidos a Eq. 17 deve ser aplicada.

```

*****
* COMO DESEJA DETERMINAR OS PARAMETROS DE CALIBRA|||O?
*
* PRESS 1> ENTRADA DIRETA
* PRESS 2> MINIMIZACAO DA UARIANCIA
*
* OPCAO:
  
```

Figura 50 – Tela de opção de entrada de parâmetros no método de Bowers

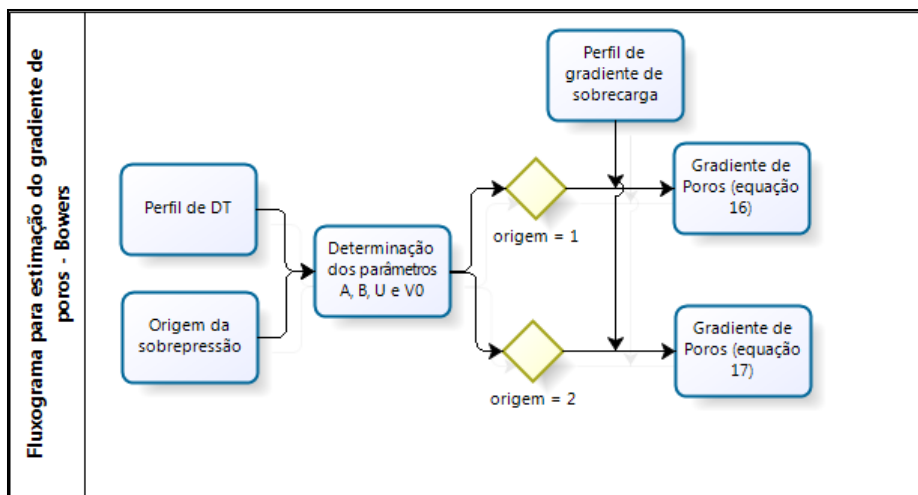


Figura 51 – Fluxograma para estimação da pressão de poros pelo método de Bowers (elaboração própria)

Finalizada a estimação do gradiente de poros o programa inicia a funcionalidade de suavizar a curva do perfil. Isso é muito interessante principalmente quando o refinamento do perfil de DT é muito grande, e a técnica de suavização é aplicada. Nesse ponto o programa permite que o usuário suavize a curva a partir de uma média ponderada com cinco os gradientes anteriores, através de parâmetros de ponderação definidos pelo usuário. Também existe a opção de realizar a ponderação por média aritmética. No arquivo UFRJDRILLINGSIMULATOR/DEBUG/DEBUGGP.TXT podemos observar os respectivos perfis de densidade da formação, tensão de sobrecarga, gradiente de sobrecarga e gradiente de poros, conforme ilustrado no mesmo.

4.3.3.3. Gradiente de Fratura

A implementação do gradiente de fratura no Bloco 6 (DETERMINAÇÃO DO GRADIENTE DE FRATURA), ao contrário do gradiente de poros, só permite uma forma de estimação, que é aquela abordada na seção 2.2.5: método da tensão mínima. Esse método demanda do programa os perfis de gradiente de poros e sobrecarga (calculados no Bloco 5), além dos dados da pressão de fechamento de fratura, advindo dos LOTs. Esses dados devem ser carregados no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT, *propriamente* conforme orientado no mesmo. O simulador só interpreta três dados de profundidade e gradiente de fratura advindos do LOT.

Nesse bloco o usuário não tem nenhuma participação. Ao selecioná-lo todo o processo descrito abaixo será implementado e o perfil de gradiente de fratura será armazenado. No arquivo UFRJDRILLINGSIMULATOR/DEBUG/DEBUGGF.TXT (Figura 53) podemos observar os perfis de gradiente de poros, gradiente de fratura e gradiente de sobrecarga, já considerando a margem de segurança para aplicação do critério de assentamento de sapatas.

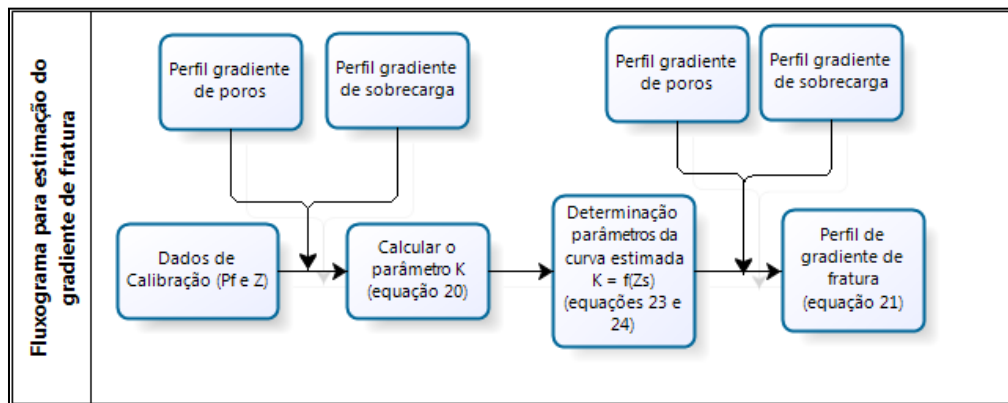


Figura 52 – Fluxograma para estimação do gradiente de fratura (elaboração própria)

PROF	GP	GP+CS	GF	GF-CS	GOV
1000.00	8.50	8.60	8.50	8.40	8.50
1010.00	8.54	8.64	8.55	8.45	8.57
1020.00	8.57	8.67	8.59	8.49	8.64
1030.00	8.59	8.69	8.63	8.53	8.72
1040.00	8.56	8.66	8.64	8.54	8.79
1050.00	8.58	8.68	8.68	8.58	8.86
1060.00	8.60	8.70	8.72	8.62	8.94
1070.00	8.62	8.72	8.76	8.66	9.01
1080.00	8.63	8.73	8.80	8.70	9.09
1090.00	8.63	8.73	8.82	8.72	9.16
1100.00	8.64	8.74	8.86	8.76	9.23
1110.00	8.64	8.74	8.88	8.78	9.30
1120.00	8.64	8.74	8.91	8.81	9.37
1130.00	8.64	8.74	8.94	8.84	9.44
1140.00	8.64	8.74	8.96	8.86	9.51
1150.00	8.63	8.73	8.99	8.89	9.58
1160.00	8.63	8.73	9.01	8.91	9.65
1170.00	8.62	8.72	9.03	8.93	9.72
1180.00	8.61	8.71	9.05	8.95	9.78
1190.00	8.60	8.70	9.07	8.97	9.85
1200.00	8.60	8.70	9.10	9.00	9.91
1210.00	8.59	8.69	9.12	9.02	9.97
1220.00	8.58	8.68	9.14	9.04	10.03
1230.00	8.57	8.67	9.15	9.05	10.09
1240.00	8.55	8.65	9.16	9.06	10.14
1250.00	8.54	8.64	9.18	9.08	10.20
1260.00	8.52	8.62	9.19	9.09	10.25
1270.00	8.51	8.61	9.21	9.11	10.31
1280.00	8.50	8.60	9.23	9.13	10.36

Figura 53 – Arquivo UFRJDRILLINGSIMULATOR/DEBUG/DEBUGGF.TXT

4.3.4. Assentamento das Sapatas

Após o cálculo das geopressões o módulo de planejamento deverá aplicar algum critério para definir qual a profundidade do assentamento das sapatas. Esse critério deverá ser estabelecido pelo usuário. O simulador, por sua vez, fornece duas possibilidades para o usuário: baseado unicamente na janela operacional e baseado na margem de riser.

Conforme apresentado na seção 2.1.2.2 a utilização de critérios de assentamento de sapatas só será conveniente para determinar as profundidades das sapatas mais profundas – intermediárias. A profundidade dos revestimentos condutor e de superfície devem ser *propostas* diretamente pelo autor no arquivo, conforme especificado. Nesse arquivo existe a solicitação de informar o número de seções do condutor, além do comprimento de cada seção. Para o revestimento de superfície a própria profundidade é definida. A partir daí aplicam-se os critérios.

4.3.4.1. Assentamento baseado na janela operacional

O Bloco 7 (PROJETO DE POÇO – CRITÉRIO DOS GRADIENTES) tem como função implementar o critério *proposto* na seção 2.1.2.2, para que assim possa-se definir o número de fases do poço, qual a profundidade onde cada sapata estará assentada, entre outras informações, que serão derivadas desse critério. Conforme apresentado na seção 2.1.2.2, esse critério é tão bom quando mais adequada for a margem de segurança aplicada (Rocha *et al.* 2006). Nesse sentido o simulador permite que se aplique uma margem em ambos os gradientes – de poros e de fratura – nesse critério. Essa margem deve ser definida no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT, na rubrica especificada.

Após a implementação do critério o simulador retornará como saída do programa um gráfico contendo os perfis dos três gradientes calculados nos blocos anteriores, suas respectivas margens de erro e os pontos de assentamento das sapatas. Outras informações podem ser obtidas a partir daí como, por exemplo, as densidades do fluido de perfuração sugerido em cada fase.

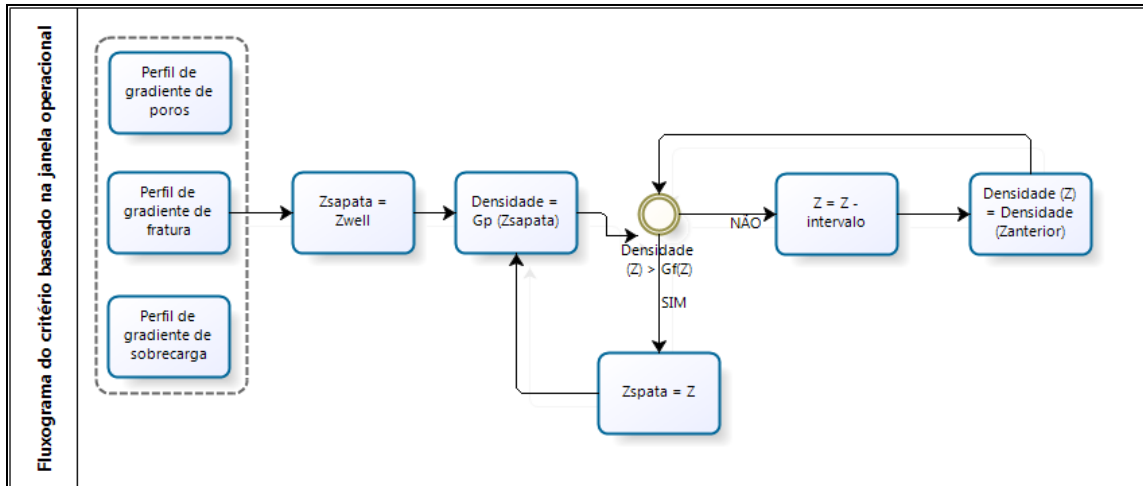


Figura 54 – Fluxograma para o critério de assentamento de sapatas baseado na janela operacional (elaboração própria)

4.3.4.2. Assentamento baseado na margem de riser

A escolha do Bloco 8 (PROJETO DE POÇO – CRITÉRIO DA MARGEM DE RISER) terá como função definir a profundidade das sapatas a partir da metodologia proposta na seção 2.1.2.3. Essa metodologia é basicamente a mesma que aquela apresentada anteriormente, excetuando-se pelo fato de que a margem de riser não levará em conta a margem de segurança usada no critério acima e recalcula essa margem baseada no limite de densidade para garantir pressão de fundo caso haja desconexão de emergência do riser.

O mesmo gráfico citado acima será plotado pelo programa também neste bloco.

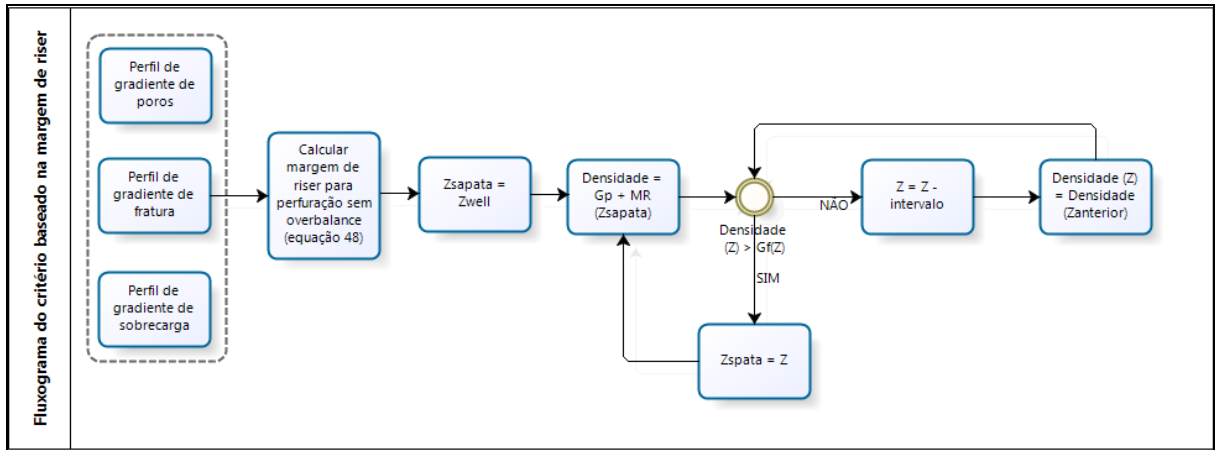


Figura 55 – Fluxograma para o critério de assentamento de sapatas baseado na margem de riser (elaboração própria)

4.3.5. Definição dos Diâmetros de Projeto

Após aplicado o critério de assentamento de sapatas escolhido pelo usuário já teremos a informação de quantas fases deverão compor o poço e questão. Com isso o próximo passo, segundo o fluxograma da Figura 33 é a definição dos respectivos diâmetros dos revestimentos e das fases, para cada estágio. Dessa forma o simulador permite que esses diâmetros sejam fornecidos ao programa de duas maneiras diferentes: (i) entrada direta do usuário, ou (ii) entrada automatizada, cabendo ao usuário essa escolha.

```

SEU POÇO TERA ? ESTAGIOS? 1 CONDUTOR;
                          1 REV. DE SUPERFICIE;
                          4 REV. INTERMEDIARIOS;
                          1 REV. DE PRODUCAO;

*****
* DETERMINE O PADRAO DE DIAMETROS
*
* PRESS 1) DEFAULT DO PROGRAMA
* PRESS 2) INSERIR MANUALMENTE
*
* OPCAO :
  
```

Figura 56 – Tela inicial da escolha da forma de carregar os arranjos dos diâmetros

Na primeira opção o programa permite que o usuário defina, para cada uma das fases (1 condutor, 1 revestimento de superfície, n revestimentos intermediários e 1 revestimento de

produção) os diâmetros da fase, do revestimento, grau do aço do revestimento e, a partir do terceiro estágio, o tipo de revestimento (*liner* ou revestimento típico).

Caso o revestimento seja pendurado na superfície o tipo de revestimento deverá ser informado como CASING ou C. Caso o revestimento seja tipo liner esse deverá ser informado ao programa como sendo LINER ou L. Na terceira informação a ser fornecida (grau do aço) o programa só se interessará pelo número representativo do limite de escoamento, ou seja, caso o aço do revestimento seja um H-40 deverá ser informado ao simulador apenas 40. Essa informação será utilizada pelo Bloco 10 para o dimensionamento dos revestimentos.

```
Determine: OD DA BROCA, OD DA REVESTIMENTO, GRAU E TIPO DO AÇO:
ESTAGIO 1: 36 30 90
ESTAGIO 2: 26 20 90
ESTAGIO 3: 18 16 100 C
ESTAGIO 4: 14 12 100 C
ESTAGIO 5: 11 10 100 C
ESTAGIO 6: 10 07 100 C
ESTAGIO 7: 07 05 110 L
```

Figura 57 – Tela do carregamento de diâmetros manualmente

Caso o usuário opte por inserir de maneira automatizada os diâmetros, segundo o padrão do programa, este deverá optar pela opção 1 – Default do Programa. Essa funcionalidade foi implementada como o objetivo de fornecer ao usuário um projeto típico de poço baseado em um esquema tradicional de arranjos de diâmetros. Esse arranjo está presente no arquivo de entrada UFRJDRILLINGSIMULATOR/INPUTS/INPUTSTAGECONFIG.TXT, que contém 22 esquemas tradicionais de diâmetro de poços, conforme apresentado na Figura 24 desse trabalho. Esse arquivo alimenta o simulador com os diâmetros, cabendo ao usuário definir, assim como da forma anterior, o grau do aço e o tipo do revestimento. A Figura 58 representa o arquivo citado, onde a coluna marcada em vermelho corresponde ao diâmetro da fase e marcado em azul representa o diâmetro do revestimento.

4	4.75	6.625	7.875	8.625	9.5	10.75	12.25	13.375	17.5	20	26
4	5.875	7	8.75	10.75	12.25	13.375	16	18.625	26		
4	5.875	7	8.75	13.375	16	18.625	26				
4.5	5.875	7	8.5	9.625	10.625	11.75	14.5	16	17.5	20	26
4.5	5.875	7	8.75	10.75	12.25	13.375	17.5	20	26		
4.5	5.875	7	8.75	13.375	17.5	20	26				
5	6.5	7.625	8.75	9.625	10.625	11.75	14.5	16	17.5	20	26
5	6.5	8.625	9.5	10.75	12.25	13.375	17.5	20	26		
5	6.5	8.625	9.5	13.375	17.5	20	26				
5	6.5	13.375	17.5	20	26						
5.5	6.5	7.625	8.75	9.625	10.625	11.75	14.5	16	17.5	20	26
5.5	7.875	7.625	8.75	9.625	10.625	13.375	17.5	20	26		
5.5	7.875	9.625	12.25	13.375	17.5	20	26				
5.5	7.875	13.375	17.5	20	26						
6.625	7.825	8.625	9.5	10.75	12.25	13.375	17.5	20	26		
6.625	8.5	9.625	10.625	13.375	17.5	20	26				
6.625	8.5	13.375	17.5	20	26						
7	8.5	9.625	10.625	11.75	12.5	13.375	14.75	16	20	20	26
7	8.5	9.625	10.625	11.75	14.5	16	17.5	20	26		
7	8.75	10.75	12.25	13.375	17.5	20	26				
7	8.75	13.375	17.5	20	26						

Figura 58 – Arquivo de entrada UFRJDRILLINGSIMULATOR/INPUTS/INPUTSTAGECONFIG.TXT

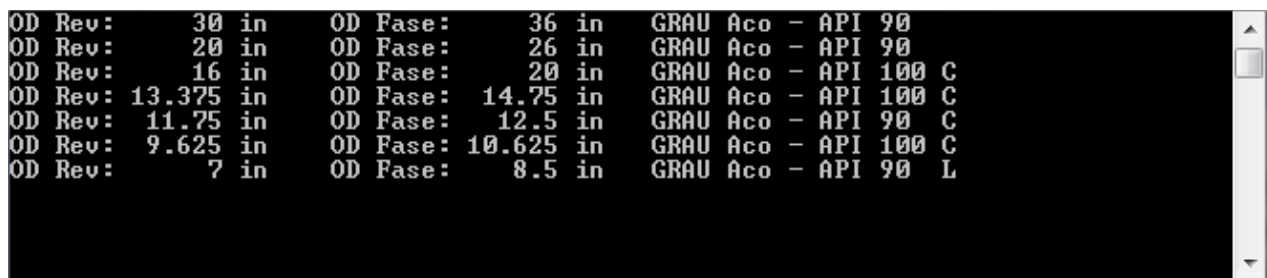
A partir desse arquivo de entrada o programa realizará uma varredura nos esquemas de forma a encontrar a combinação que satisfaça dois critérios: (i) o número de estágios, definido pelo Bloco 7 ou 8 e (ii) o diâmetro do revestimento de produção, definido pelo usuário e carregado no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT, conforme especificado no mesmo. A distribuição dos arranjos de diâmetros no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTSTAGECONFIG.TXT está esquematizada na tabela a seguir.

OD do Revestimento de Produção	Número de Estágios do Poço
4 in	7, 6, 5
4 ½ in	7, 6, 5
5 in	7, 6, 5, 4
5 ½ in	7, 6, 5, 4
6 5/8 in	7, 6, 5
7 in	7, 6, 5, 4

Tabela 5 – Esquemas de diâmetros padrão do simulador

Independentemente do número de fases ou do diâmetro do revestimento produtor o diâmetro do revestimento condutor está sempre definido como sendo 36 in na fase e 30 in no revestimento.

Dessa forma podemos inferir que, para diâmetros do revestimento produtor diferentes do conjunto apresentado na Tabela 5, não é adequado que se utiliza a opção de escolha automatizada dos diâmetros. Da mesma forma caso o bloco 7 ou 8 forneçam um número de estágios superior a 7 ou inferior a 4 essa opção também não é adequada. Essa observação cabe ao usuário. Em ambos os casos a opção de escolha direta é a mais correta.



OD Rev:	30 in	OD Fase:	36 in	GRAU Aco - API 90
OD Rev:	20 in	OD Fase:	26 in	GRAU Aco - API 90
OD Rev:	16 in	OD Fase:	20 in	GRAU Aco - API 100 C
OD Rev:	13.375 in	OD Fase:	14.75 in	GRAU Aco - API 100 C
OD Rev:	11.75 in	OD Fase:	12.5 in	GRAU Aco - API 90 C
OD Rev:	9.625 in	OD Fase:	10.625 in	GRAU Aco - API 100 C
OD Rev:	7 in	OD Fase:	8.5 in	GRAU Aco - API 90 L

Figura 59 – Tela do carregamento de diâmetros de forma automatizada

4.3.6. Dimensionamento dos Revestimentos

Após definirmos os diâmetros de cada revestimento no Bloco 9 o simulador já teria informações suficientes para realizar o dimensionamento de cada um, conforme metodologia apresentada na seção 2.3.2. Tal dimensionamento é desenvolvido no Bloco 10 (DIMENSIONAMENTO DOS REVESTIMENTOS) e ocorre independentemente da interferência do usuário. Será adotado, como forma de simplificação, que todas as seções do revestimento terão o mesmo diâmetro.

Para o revestimento condutor foi considerado apenas o dimensionamento quanto a tração, uma vez que não há existência de espaço anular e, conseqüentemente, o mesmo não seria sujeito a carregamentos externos.

Na seção 2.3.2 foi apresentada toda a metodologia para o dimensionamento de um tubo circular submetido a esforços de tração, pressão interna e pressão externa, sem que os esforços fossem conjuntos. Nesse sentido esse bloco foi implementado de forma a fornecer o

menor diâmetro que garantisse que aquele revestimento seria capaz de sustentar esses carregamentos, de maneira não conjunta, sem falhar. Para tal foram escolhidos alguns cenários críticos em que o revestimento poderia ser solicitado para que o correto dimensionamento fosse realizado.

4.3.6.1. Cenário crítico para resistência a tração

Para resistência a tração foi considerado como cenário crítico o carregamento axial que solicita o revestimento no topo do mesmo. Conforme discutido na seção 2.3.2.1, o carregamento axial no revestimento é fruto do próprio peso do tubo, descontado o fator de atrito. Assim sendo, a condição crítica para dimensionamentos será para Z igual a zero, considerando um coeficiente de segurança arbitrário de 15%.

$$F = 1,15 w_s \beta Z_{well} \quad (102)$$

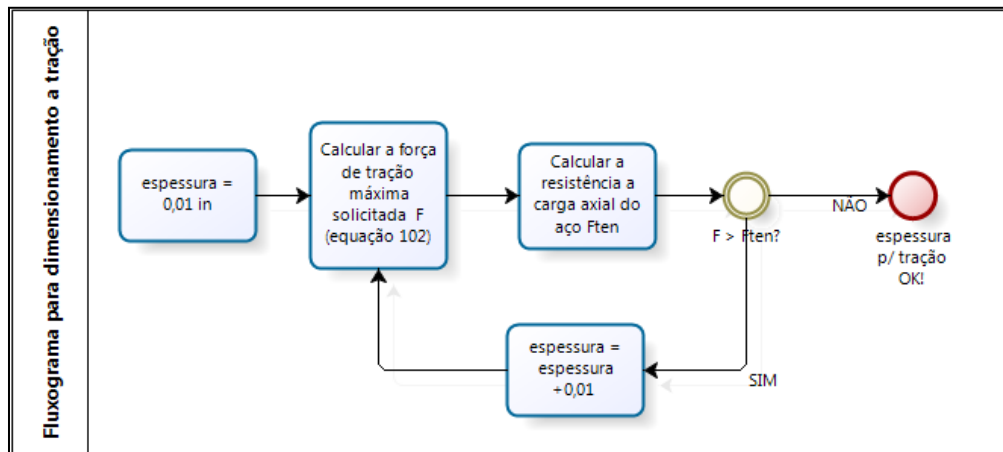


Figura 60 – Fluxograma para o dimensionamento do revestimento frente ao colapso (elaboração própria)

4.3.6.2. Cenário crítico para resistência a pressão interna

Para resistência a explosão o foi implementado como cenário crítico aquele onde o revestimento estaria preenchido internamente pelo fluido de perfuração aplicado para fase seguinte e externamente por uma coluna de gás compreendendo todo o espaço anular. Será

considerado, da mesma forma, um coeficiente de segurança de 15% sobre a pressão interna máxima permissível e ponto no qual será feita tal análise será na profundidade da sapata.

Vale observar que a análise será realizada baseada na pressão equivalente, dada pela Eq. 29, onde as respectivas pressões internas e externas serão dadas por:

$$P_i = \alpha \rho_f Z \quad (103)$$

$$P_e = \alpha \rho_g Z \quad (104)$$

Nessa formulação temos que:

- ρ_g densidade do gás, considerada constante ao longo de todo poço, e fornecida pelo usuário

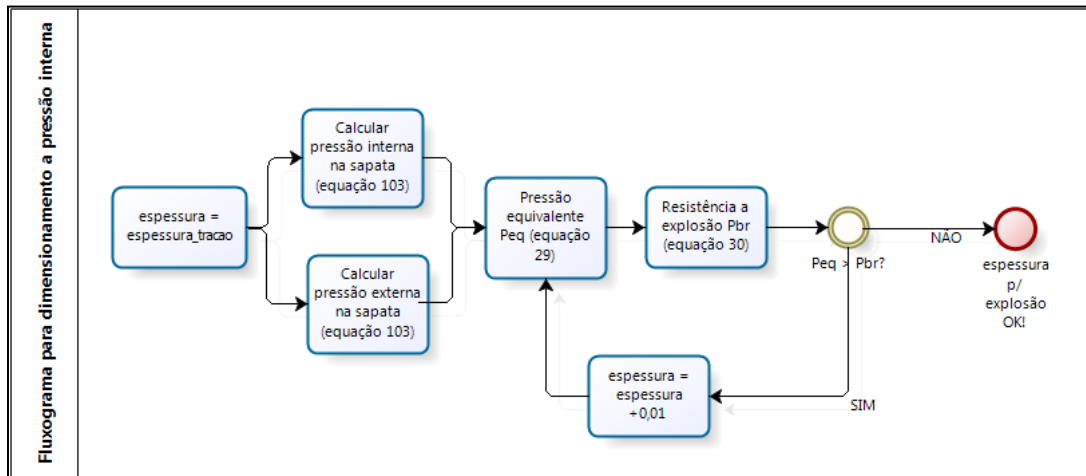


Figura 61 – Fluxograma para dimensionamento do revestimento frente a explosão (elaboração própria)

4.3.6.3. Cenário crítico para resistência a colapso

Para o dimensionamento do revestimento frente ao colapso o cenário crítico considerado será aquele onde o revestimento estará internamente preenchido pelo fluido de perfuração e, externamente, por cimento. Essa coluna de cimento estaria dimensionada de forma a garantir que, na profundidade da sapata, a pressão hidrostática será igual a pressão de fratura da

formação. Da mesma forma que nos casos anteriores consideraremos um fator de segurança de 15%.

Como visto na seção 2.3.2.3 o equacionamento para a estimação da resistência do revestimento ao colapso é função de parâmetros característicos para cada aço, denominados F_1, F_2, F_3, F_4 e F_5 . Tais parâmetros devem ser carregados no simulador, e isso é feito através do arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTFCOEFFICIENT.TXT (Figura 62).

SIGMA	F1	F2	F3	F4	F5
40000	2.950	0.0465	754	2.063	0.0325
50000	2.976	0.0515	1056	2.003	0.0347
55000	2.991	0.0541	1206	1.989	0.0360
60000	3.005	0.0566	1356	1.983	0.0373
70000	3.037	0.0617	1656	1.984	0.0403
75000	3.054	0.0642	1806	1.990	0.0418
80000	3.071	0.0667	1955	1.998	0.0434
90000	3.106	0.0718	2254	2.017	0.0466
95000	3.124	0.0743	2404	2.029	0.0482
100000	3.143	0.0768	2553	2.040	0.0499
105000	3.162	0.0794	2702	2.053	0.0515
110000	3.181	0.0819	2852	2.066	0.0532
120000	3.219	0.0819	3151	2.092	0.0565
125000	3.239	0.0895	3301	2.106	0.0582
130000	3.258	0.0920	3451	2.119	0.0599
135000	3.278	0.0971	3601	2.133	0.0615
140000	3.297	0.0971	3751	2.146	0.0632
150000	3.336	0.1021	4053	2.174	0.0666
155000	3.336	0.1047	4204	2.188	0.0683
160000	3.375	0.1072	4356	2.202	0.0700
170000	3.412	0.1123	4660	2.231	0.0734
180000	3.449	0.1173	4966	2.261	0.0769

Figura 62 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTFCOEFFICIENT.TXT

Nesse caso utilizaremos o conceito de pressão equivalente assim como para o dimensionamento a pressão interna.

$$P_i = \alpha \rho_f Z \quad (105)$$

$$P_e = \alpha G_f Z \quad (106)$$

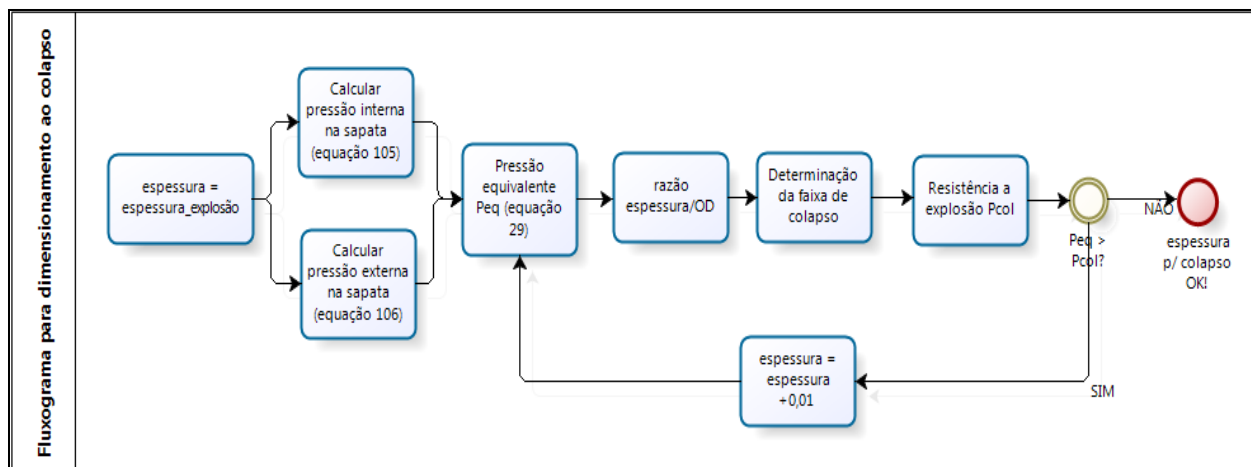


Figura 63 – Fluxograma para dimensionamento do revestimento frente ao colapso (elaboração própria)

4.3.7. Verificação do BOP

O Bloco 11 (DIMENSIONAMENTO DO BOP), assim como o bloco anterior, funciona sem a influência direta do usuário. Nesse bloco será verificada a pressão máxima de trabalho adequada para o *blowout preventer* segundo os quatro critérios apresentados na seção 2.4. Tal aplicação será direta e realizada para um dos diferentes revestimentos. Ao final das sucessivas aplicações desse bloco será escolhido como BOP adequado aquele de menor pressão de trabalho que satisfizer todos os quatro critérios. Quatro capacidades de BOP serão considerados nesse estudo: 5000, 10000 e 15000 *psi*. Caso o dimensionamento forneça uma pressão de trabalho maior que 15.000 *psi* a resposta do simulador será que a solicitação supera a disponibilidade do mercado.

4.3.8. Dimensionamento da coluna de cimento

Seguindo o fluxograma apresentado pela Figura 33 o próximo passo visando o projeto do poço seria o dimensionamento da coluna de cimento que deve aplicada para cada seção do poço. Esse dimensionamento será realizado pelo Bloco 12 e, assim como os anteriores, será executado sem participação direta ou indireta do usuário.

O dimensionamento da coluna, bem como do volume de cimento, que devemos aplicar no poço a cada etapa de assentamento de revestimento é uma etapa importante pois, caso esta seja superestimada, podemos ter problemas de fraturamento da formação abaixo da sapata,

levando a perdas, ou então teríamos a redução da qualidade da cimentação. Em contrapartida, caso a coluna seja subestimada poderia haver perda na capacidade de sustentar o revestimento. Nesse sentido assumiremos dois critérios básicos para o dimensionamento, sendo que a escolha do volume de cimento adequado seria o maior volume tal que satisfizesse ambos os critérios.

- Coluna de cimento deve ser sempre inferior a um limite máximo que garanta a qualidade da cimentação, normalmente de aproximadamente 800 m para cimentação em um estágio;
- Coluna de cimento deve ser tal que a composição do anular cimento mais fluido forneça uma pressão de fundo sempre menor que a pressão de fratura da formação

Nessa análise não estamos considerando a reologia do cimento, tampouco o perfil de pressão que ele exerce durante a pega. Esse está sendo considerado como um fluido de densidade fornecida pelo usuário que, ao ser alojado no anular revestimento/formação fornecerá uma pressão hidrostática dada pela Eq. 107. Esses critérios serão aplicados para os revestimentos intermediários e de produção. Para o revestimento de superfície é praxe da indústria que todo revestimento seja cimentado, desde a sapata até a superfície.

$$P_{cim} = \alpha [\rho_{cim} Z_{cim} + \rho_f (Z_{well} - Z_{cim})] \quad (107)$$

Nessa formulação temos que:

- ρ_{cim} densidade do cimento, fornecida pelo usuário
- Z_{cim} altura da coluna de cimento alojada no anular

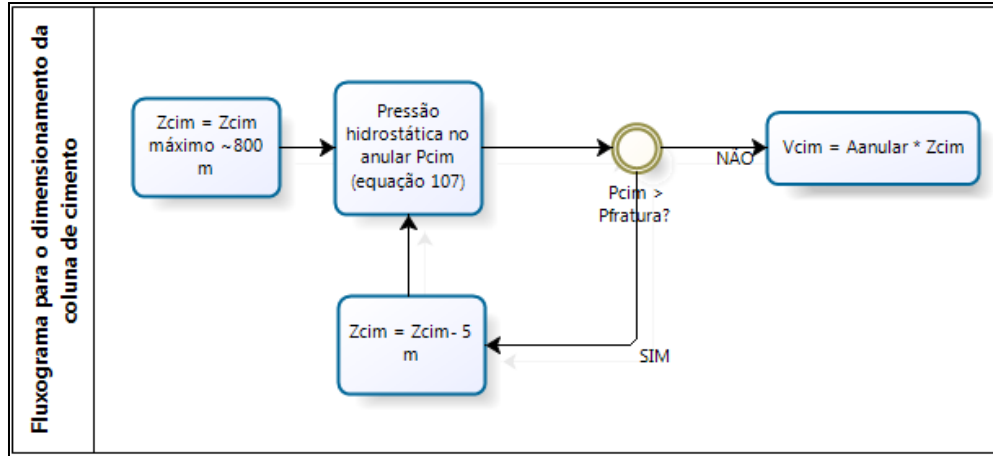


Figura 64 – Fluxograma para o dimensionamento da coluna e do volume de cimento (elaboração própria)

4.3.9. Modelo de Perfuração e Estimação do Tempo de Perfuração

O próximo bloco do módulo de planejamento de poço é o Bloco 13 (Modelo de Perfuração e Estimação do Tempo de Perfuração). Assim como os blocos anteriores este foi programado para ser implementado sem a participação do usuário e, seguindo o fluxograma da Figura 33, após o dimensionamento da coluna de cimento.

O tempo total de perfuração será composto por uma parcela produtiva e outra não produtiva aos olhos da perfuração. A parcela produtiva será dada a partir do modelo de perfuração. Entretanto, a parcela não produtiva (*NPT*) no escopo do simulador será composta por três diferentes parcelas: tempo de manobra, tempo para assentamento do revestimento e tempo de deslocamento até o início da perfuração em si. A forma como o simulador calcula todos esses parâmetros no módulo de planejamento será o assunto dessa seção.

4.3.9.1. Estimação do *PT*

O tempo de perfuração produtivo (ou *PT*) é o conceito utilizado para descrever a parcela do tempo gasto durante a perfuração em que a broca está progredindo dentro do poço. A estimação desse tempo é simples, e advém da relação:

$$PT = \frac{\Delta Z}{ROP} \quad (108)$$

Durante a estimação do tempo de perfuração os intervalos de progressão utilizados são intervalos de profundidade, ou seja, a cada valor fixo em que a broca avança no poço recalcula-se o *ROP* e todos os parâmetros relevantes para sua estimação, bem como o tempo em si. Assim sendo o termo ΔZ é conhecido e constante para todo o poço, de forma que o parâmetro variável e que deve ser estimado é a taxa de penetração *ROP*. Para tal consideraremos a broca como sendo tricônica, e aplicaremos o modelo de Bourgoyne-Young proposto na seção 3.3.3.

Como visto nesta seção o modelo de Bourgoyne-Young considera oito relações funcionais que visam abranger diferentes parâmetros que impactam no desempenho da broca, bem como na taxa de penetração. Tais relações funcionais são ajustadas ao ambiente de perfuração através de oito coeficientes, que devem ser definidos baseado em poços de correlação, como explicado na mesma seção.

Nesse sentido o bloco do simulador tem uma maneira específica de tratar todos os parâmetros envolvidos que veremos a seguir.

A) Carregamento dos parâmetros de calibração

Como dito acima os parâmetros de calibração ($a_1, a_2, a_3, \dots, a_8$) devem ser obtidos baseado em dados de *mud logging* nos poços de correlação. A partir de tais dados regressões lineares e exponenciais devem ser desenvolvidas, através das quais tais parâmetros seriam extraídos (Bourgoyne *et al.* 1986). A metodologia para a obtenção desses coeficientes não foi implementada no simulador, de forma que estes devem funcionar como entradas no mesmo.

Apesar de tais coeficientes serem função não apenas da litologia, mas também das condições físicas e mecânicas da perfuração em uma determinada profundidade, foi considerado, em nível de simplificação, que o conjunto de parâmetros será função apenas da litologia, ou seja, é permitido ao usuário carregar um conjunto de dados para folhelhos e um conjunto para arenitos. Essa lógica se manterá para o carregamento de outros parâmetros que afetam a perfuração e são propriedades da formação geológica: a constante de abrasividade (τ_H) e a tensão mínima para fraturamento da formação $\left(\frac{WOB}{D_b}\right)_t$, que serão relevantes para as relações funcionais f_7 e f_5 respectivamente.

Todos esses dez parâmetros devem estar carregados no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTBOURGOYNEPARAMETERS.TXT, apresentado na Figura 65. Podemos observar que linha marcada em azul corresponde aos dados relativos a folhelhos (litologia igual a 1), enquanto que os dados da linha marcada em vermelho corresponde os parâmetros relativos a arenitos (litologia igual a 2).

abrasiv (w/d)t	a1	a2	a3	a4	a5	a6	a7	a8	
100	0	1.5	0.00007	0.000005	0.00003	1	0.5	0.5	0.5
100	0	1.5	0.00007	0.000005	0.00003	1	0.5	0.5	0.5

Figura 65 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTBOURGOYNEPARAMETERS.TXT

B) Carregamento dos parâmetros da perfuração

Além dos parâmetros de calibração $a_1, a_2, a_3, \dots, a_8$ e os demais relativos a características da formação geológica o modelo de Bourgoyne-Young também demanda uma série de outros parâmetros da perfuração em si, como peso sobre a broca, rotação, densidade do fluido, vazão, parâmetros característicos da broca, entre outros. No módulo em questão, como a manipulação desses parâmetros não é dinâmica, como no caso do módulo de simulação em tempo real, é necessário constituir um cenário de perfuração ao longo do tempo que venha a ser representativo do caso durante a perfuração efetivamente.

Dessa forma devemos carregar um conjunto de parâmetros de perfuração dentro do simulador para que, baseado nesse conjunto, o modelo seja aplicado e o *ROP* seja estimado. Para tornar a análise mais próxima da realidade foi implementada a funcionalidade de se carregar um determinado conjunto de parâmetros para cada seção perfurada com uma broca específica em algum poço de correlação. Os parâmetros serão considerados constantes dentro da seção, e, para aquele trecho, utilizaremos o mesmo cenário que teria sido utilizado em outro poço da mesma área. Esse conjunto de dados deve estar contido no arquivo de entrada UFRJDRILLINGSIMULATOR/INPUTS/INPUTBITRECORD.TXT.

#BROCA	#Zin	Zfin	tempo	tempot	H1	H2	(w/d)m	WOB	N	jet1	jet2	jet3
1	1100	1500	10	10	1.84	6	8	60000	80	12	12	16
2	1500	2100	16	28	1.84	6	8	60000	80	12	12	16
3	2100	2500	18	46	1.84	6	8	40000	80	12	12	16
4	2500	2800	24	70	1.84	6	8	40000	80	12	12	16
5	2800	3150	30	100	1.84	6	8	60000	80	12	12	16
6	3150	3400	35	185	1.84	6	8	40000	80	12	12	16
7	3400	3650	40	225	1.84	6	8	50000	80	12	12	16
8	3650	3900	46	271	1.84	6	8	50000	80	12	12	16
9	3900	4150	45	366	1.84	6	8	50000	80	12	12	16
10	4150	4300	40	406	1.84	6	8	50000	70	12	12	16
11	4300	4500	35	441	1.84	6	8	50000	70	12	12	16
12	4500	4700	35	476	1.84	6	8	50000	70	12	12	16
13	4700	4850	37	583	1.84	6	8	60000	70	12	12	16
14	4850	4950	45	628	1.84	6	8	60000	70	12	12	16
15	4950	5200	50	678	1.84	6	8	50000	70	12	12	16

Figura 66 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTBITRECORD.TXT

Além dos parâmetros carregados através do arquivo de entrada acima outros dois parâmetros, considerados constantes ao longo de toda a perfuração, são carregados através do arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT. Estes parâmetros são a vazão no poço e a viscosidade do fluido, uma vez que o mesmo será considerado sempre newtoniano, em nível de simplificação. A densidade utilizada será a estimada para a perfuração de cada fase, a partir do bloco 7 ou 8. Outro parâmetro relevante para a estimação do *ROP* é o gradiente de poros, que será utilizado o perfil calculado no bloco 5. Com esse conjunto de parâmetros o simulador já teria o cenário pronto para iniciar a simulação.

C) Processamento dos dados

A partir do cenário configurado pelo carregamento dos dados estes devem ser processados através das relações funcionais do modelo e, finalmente, aplicadas para estimar o *ROP*. Nesse sentido a abordagem do simulador para estimar cada uma das relações do modelo será:

i) Relação funcional f_1

Para estimarmos o valor de f_1 a partir da Eq. 88 o único dado necessário é o parâmetro de calibração a_1 . Dessa forma a relação será fornecida de maneira direta e o valor será constante para toda uma litologia.

ii) *Relação funcional f_2*

Para estimarmos a relação funcional f_2 devemos aplicar na Eq. 89 apenas os dados da profundidade em que estamos perfurando e o parâmetro de calibração a_2 . Para cada instante de tempo devemos recalculá-la, uma vez que a perfuração está mais profunda e a rocha deverá estar mais compactada.

iii) *Relação funcional f_3*

Como visto na seção 3.3.3.3 essa relação visa computar o efeito da subcompactação e, para tal, demanda o perfil do gradiente de poros, além da profundidade e do parâmetro de calibração correspondente. Para cada intervalo de tempo devemos atualizar esse valor com a profundidade correspondente e o gradiente de poros respectivo. Com tais valores o simulador computa essa relação aplicando a Eq. 90.

iv) *Relação funcional f_4*

A relação funcional f_4 , ao contrário das relações anteriores, demanda um dado que não foi estimado no módulo de planejamento de poço até esse ponto. Essa informação é *ECD* (*Equivalent Circulating Density*), a partir da qual pode-se estimar a magnitude do overbalance durante a perfuração.

Para estimar o *ECD* é necessário lançar mão de conceitos de hidráulica de poço, que foram previamente abordados na seção 3.2 deste trabalho. Ao contrário do que será aplicado no próximo módulo o modelo de hidráulica aplicado foi de fluido newtoniano, explicado na seção 3.2.2.5.A, para escoamento dentro do anular. Para esse modelo serão necessários parâmetros da geometria da coluna e do revestimento, parâmetros do escoamento e do próprio fluido. Todos estes ficam a cargo do usuário, através dos mecanismos explicados anteriormente.

Após estimarmos a perda de carga ao longo do espaço anular do poço podemos estimar o *BHP* através da equação:

$$BHP_{anular} = P_h(Z_{well}) + (\Delta P_f)_{anular} \quad (109)$$

Daí o ECD será estimado pela equação abaixo. Dessa forma devemos atualizar o ECD , bem como os demais parâmetros relevantes nessa relação, a cada interação do simulador, de forma a obter o perfil dessa relação ao longo da perfuração.

$$ECD = \frac{BHP}{\alpha Z_{well}} \quad (110)$$

iv) *Relação funcional f_5*

Esta relação visa mensurar o efeito do peso sobre a broca sobre a taxa de penetração. Com o carregamento dos dados através do arquivo contendo os dados dos poços de correlação e os parâmetros de calibração referentes e a resistência do fraturamento daquela litologia o simulador já tem as informações necessárias para estimar a contribuição dessa relação funcional sobre o ROP . Esse cálculos era feito através da aplicação dos valores constantes de peso e resistência da rocha na Eq. 92, devendo ser atualizado para cada profundidade manipulada, uma vez que os valores de entrada só são constantes dentro do mesmo intervalo de perfuração de uma broca, constando no arquivo do *bit record*.

v) *Relação funcional f_6*

Assim como a relação funcional acima o simulador manipulará a mesma a de maneira direta, através da Eq. 93, onde servirão como entradas o parâmetro de calibração correspondente e a rotação, que será constante para um determinado trecho, dado pelo arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTBITRECORD.TXT, como explicado anteriormente. Essa relação também deve ser atualizada para cada avanço no poço.

vi) *Relação funcional f_7*

A relação funcional f_7 visa computar o efeito do desgaste sobre a broca na taxa de penetração. Como foi visto na seção 3.3.3.7 esse desgaste é função de diversos parâmetros, tanto físicos da própria broca, quanto da formação que a mesma está perfurando. Todos estes parâmetros devem ter sido previamente carregados no simulador. Os parâmetros referentes a broca em si serão carregados a partir do *Bit Record* dos poços de correlação, ou seja, consideramos que o poço perfurará as mesmas seções com as mesmas brocas desse outro

poço. Já os parâmetros da litologia serão carregados no arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTBOURGOYNEPARAMETERS.TXT. A partir daí o simulador seguirá a metodologia explicada naquela seção, calculando o parâmetro J_2 (Eq. 98) e resolvendo a equação quadrática Eq. 100. A solução aplicada pelo simulador para tal equação é a solução clássica de Báskara, que fornecerá o parâmetro h através da equação:

$$h = \sqrt{\left(\frac{1}{H_2}\right)^2 + \left(\frac{2t_b}{H_2 J_2 \tau_H}\right)} - \left(\frac{1}{H_2}\right) \quad (111)$$

vii) *Relação funcional f_8*

Finalmente a relação funcional f_8 visa avaliar a influência da hidráulica sobre a taxa de penetração. O parâmetro hidráulico escolhido foi a pressão hidráulica exercida pelos jatos da broca sobre a formação. Essa pressão deve ser calculada para todo o intervalo analisado pelo simulador, utilizando-se a Eq. 62. Nessa equação utiliza-se como entradas parâmetros da própria broca, como diâmetro dos jatos, e parâmetros do escoamento em si, como vazão. Todos estes deve ser carregados através dos arquivos de entrada como explicado ao longo desse capítulo.

D) Processamento dos dados

Finalmente, de posse dos respectivos valores das oito relações funcionais atualizadas a cada profundidade estudada o simulador deve aplicar a Eq. 87 como forma de obter ao longo do poço, o ROP estimado. Esse cálculo é feito de maneira dinâmica, ou seja, a cada interação o poço avança de um intervalo ΔZ , todas as relações são atualizadas e o ROP é recalculado. Daí atualizados o tempo produtivo através da Eq. 108, como mostrado anteriormente. A Figura 67 apresenta um fluxograma esquemático do funcionamento do bloco para estimação do PT .

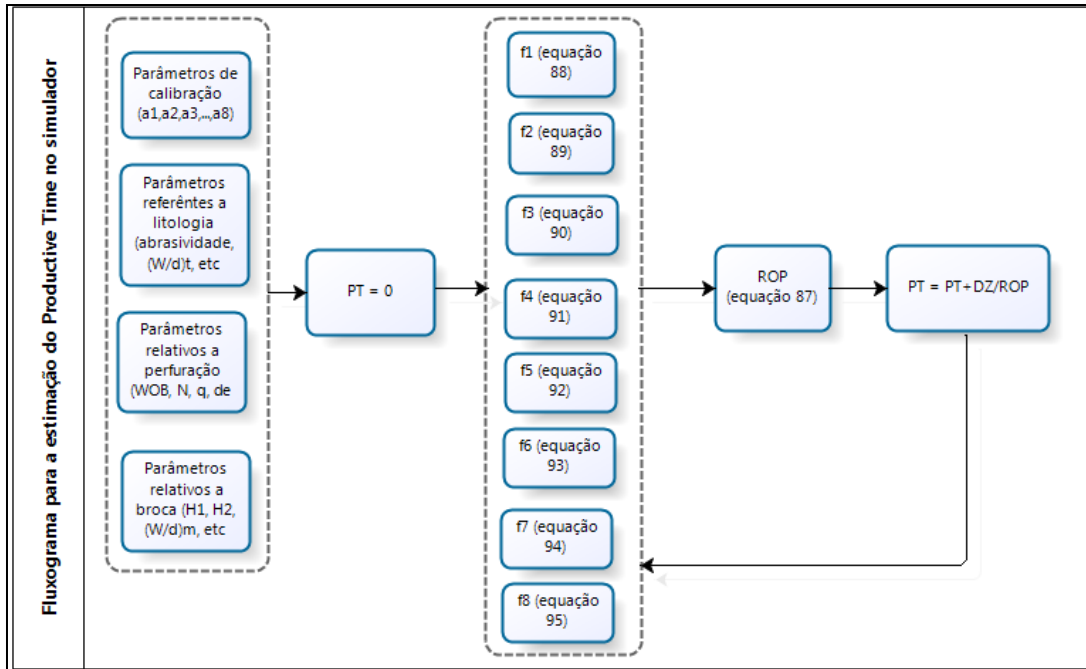


Figura 67 – Fluxograma genérico para estimação do PT

4.3.9.2. Estimação do NPT

Ao contrário da estimação PT os cálculos do tempo de perfuração não produtivo não serão realizados ao longo de toda a perfuração mas sim nos momentos em que não há avanço do poço. Como elencado anteriormente o simulador tratará de três diferentes parcelas que virão a compor o NPT . São elas: (i) tempo de deslocamento da sonda até a posição correta e início das atividades de perfuração, (ii) tempo de manobra e (iii) tempo para assentamento dos revestimentos.

A abordagem para estimação de cada uma dessas componentes, bem como os momentos em que o simulador compreende a perfuração deve ser interrompida para o início de um desses eventos será explicada em seguida.

A) Tempo de deslocamento até o início da perfuração

Será considerado como tempo de deslocamento o intervalo entre o tempo zero e o início da perfuração da fase correspondente ao revestimento de superfície, sendo o tempo zero igual a momento em que se começa a pagar pelo aluguel da sonda.

Todos os eventos que acontecem nesse íterim devem ser representados por um valor de tempo absoluto, que será carregado no simulador pelo arquivo de entrada UFRJDRILLINGSIMULATOR/INPUTS/INPUTCASINGPLACEMENT.TXT. Nesse arquivo, que veremos na próxima seção, o segundo valor representado corresponde ao tempo de deslocamento que o simulador utilizará para compor o tempo total de perfuração, cabendo ao usuário representar coerentemente tal valor, baseado na experiência de campo. Assim sendo, quando a perfuração é efetivamente iniciada o tempo total já está computando o tempo de deslocamento.

B) Tempo de manobra

A manobra é um evento da perfuração que se caracteriza pela retirada da coluna de perfuração para alguma intervenção direta na mesma. Em um contexto geral as manobras podem ser realizadas para mudar a composição do *BHA*, trocar ferramentas defeituosas ou mesmo como forma de atenuar o problema de fluência ao perfurar formações salinas. Porém, para o simulador a manobra só será executada em dois casos: (i) necessidade de trocar a broca ou (ii) retirada da coluna para o assentamento do revestimento.

Em ambos os casos a perfuração do poço será interrompida, de forma a mantermos fixo o tempo produtivo (*PT*). A coluna será retirada do poço, a broca trocada e, na sequência, a mesma será descida novamente. Porém, aos olhos do simulador, o tempo de troca da broca será desprezível frente o tempo de subida e descida da coluna, que, em nível de simplificação, serão considerados iguais. Bourgoyne *et al.* (1986) propuseram que, o tempo total de manobra pode ser simplificado como função apenas da profundidade, a partir do tempo de manipulação de uma única seção da coluna. Esse tempo seria dado por:

O simulador considerará que a broca deverá ser trocada sempre que o parâmetro de desgaste h atingir 100%, e que o revestimento deve ser assentado sempre que for atingida alguma profundidade de assentamento de sapatas estimadas nos blocos 7 ou 8.

$$t_t = 2 \left(\frac{\bar{t}_s}{L_{dp}} \right) Z \quad (112)$$

Nessas formulações temos que:

- t_t tempo total de manobra, em h ;
- \bar{t}_s tempo médio para retirar uma seção da coluna, em h ;
- L_{dp} comprimento médio de uma seção da coluna, em m .

Os parâmetros \bar{t}_s e \bar{l}_s devem ser carregados no simulador através do arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTMAIN.TXT, onde esses parâmetros devem advir de observações de campo.

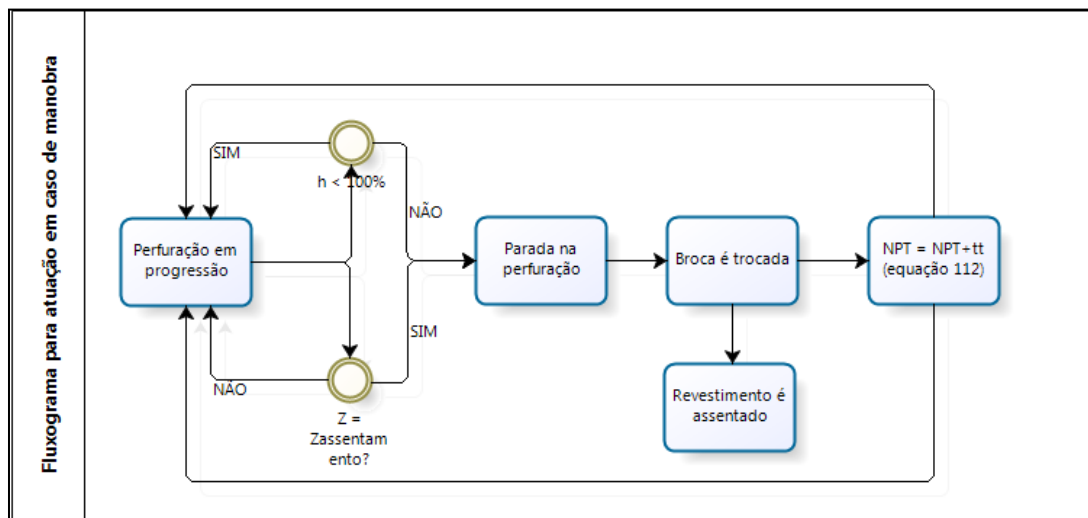


Figura 68 – Fluxograma para atuação do simulador em caso de manobra (elaboração própria)

C) Tempo de assentamento do revestimento

O assentamento dos revestimentos também compõe uma parcela importante no computo do NPT . É intuitivo pensarmos que, assim como o evento de manobra, o tempo de assentamento dos revestimento é função direta da profundidade que eles se encontram. Assim sendo, para caracterizar o tempo de assentamento dos revestimentos devemos desenvolver alguma relação funcional entre este e a profundidade em questão.

Aos olhos do simulador o tempo de assentamento compreende todo o tempo necessário para descer o revestimento, cimentar, avaliar a qualidade da cimentação, esperar a pega e

toda e qualquer outra atividade que ocorra no ínterim entre a subida da coluna e a posterior descida da coluna para iniciar uma nova fase. No caso do assentamento de revestimentos o programa considerará que haverá sempre a troca de broca de uma fase para outra, uma vez que a nova broca terá diâmetro diferente da anterior.

Dessa forma foi implementado que o tempo para assentar um revestimento a uma determinada profundidade Z poderá ser descrita por uma função linear obtida de uma regressão realizada com os dados de tempo de assentamento e profundidade do revestimento advindas de poços de correlação. Tais informações devem estar armazenadas no arquivo de entrada UFRJDRILLINGSIMULATOR/INPUTS/INPUTCASINGPLACEMENT.TXT.

```

InputCasingPlacement - Notepad
File Edit Format View Help
7 // # de revestimentos assentados
60 // tempo de deslocamento inicial
400 20 // prof e tempo (h) do assentamento do 1o revestimento
2800 35 // prof e tempo (h) do assentamento do 2o revestimento
3300 45 // prof e tempo (h) do assentamento do 3o revestimento
3600 50 // prof e tempo (h) do assentamento do 4o revestimento
4000 53 // prof e tempo (h) do assentamento do 5o revestimento
4500 60 // prof e tempo (h) do assentamento do 6o revestimento
5000 65 // prof e tempo (h) do assentamento do 7o revestimento

```

Figura 69 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTCASINGPLACEMENT.TXT

$$t_{rev} = a + b \ln(Z) \quad (113)$$

$$a = \frac{\sum Z^2 \sum t_{rev} - \sum Z t_{rev} \sum Z}{n \sum Z^2 - (\sum Z)^2} \quad (114)$$

$$b = \frac{n \sum Z t_{rev} - \sum Z \sum t_{rev}}{n \sum Z^2 - (\sum Z)^2} \quad (115)$$

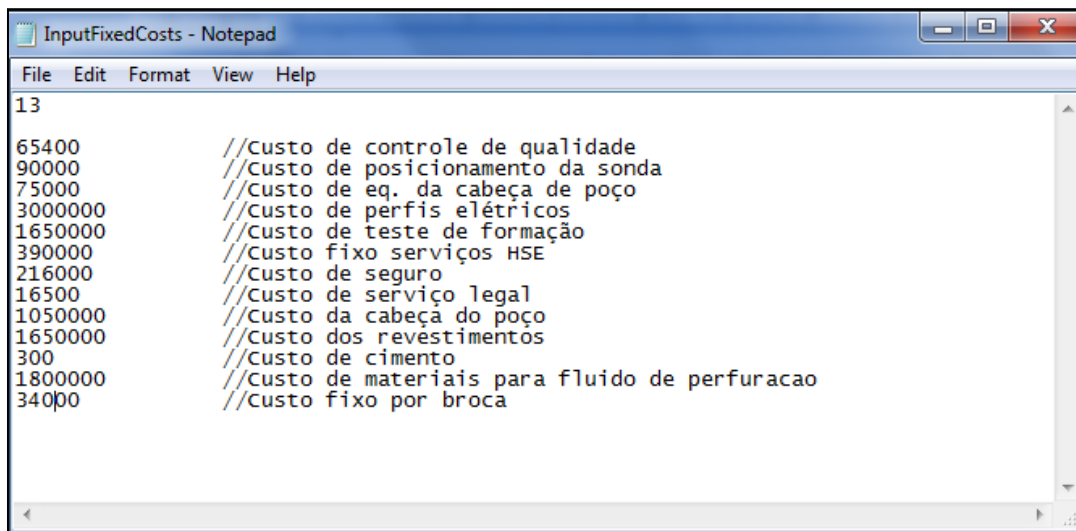
Portanto podemos representar o NPT estimado para aquele poço através da expressão:

$$NPT = t_{desl} + t_t + t_{rev} \quad (116)$$

4.3.10. Estimação do Custo da Perfuração

Finalmente, após o simulador computar todos os parâmetros listados ao longo desse capítulo este já tem armazenado um conjunto de dados posteriormente exportado para o usuário como forma de relatórios e gráficos, como veremos posteriormente. Além disso tal conjunto de dados será utilizado pelo simulador para estimar o custo associado a perfuração daquele poço, completando o conjunto de resultados fornecido pelo módulo de planejamento de poço.

Para a estimação do custo associado a essa perfuração o software demandará como entrada um conjunto de custos classificados como fixos (ou seja, aqueles que são cobrados de uma única vez) e um conjunto de custos dito variáveis (ou seja, que são cobrados ao longo do tempo). O conjunto de custos fixos considerará 16 variáveis e deverão constar no arquivo de entrada UFRJDRILLINGSIMULATOR/INPUTS/INPUTFIXEDCOSTS.TXT. Tal arquivo, bem como a origem de cada custo, pode ser encontrado na figura abaixo. Tais valores devem estar expressos na moeda de interesse.



```
File Edit Format View Help
13
65400 //Custo de controle de qualidade
90000 //Custo de posicionamento da sonda
75000 //Custo de eq. da cabeça de poço
3000000 //Custo de perfis elétricos
1650000 //Custo de teste de formação
390000 //Custo fixo serviços HSE
216000 //Custo de seguro
16500 //Custo de serviço legal
1050000 //Custo da cabeça do poço
1650000 //Custo dos revestimentos
300 //Custo de cimento
1800000 //Custo de materiais para fluido de perfuracao
34000 //Custo fixo por broca
```

Figura 70 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTFIXEDCOSTS.TXT

O conjunto de custos variáveis utilizado pelo bloco para a estimação dos custos da perfuração também será composta por 16 dados, que devem ser carregados da seguinte maneira.

```

InputVariableCosts - Notepad
File Edit Format View Help
16 // # de dados abaixo
10500 //Custo com helicóptero
60000 //Custo com a base de apoio
78000 //Custo com barcos de apoio
6000 //Custo com gerencia de residuos
450 //Custo com previsão climatológicas
10200 //Custo com descida do revestimento
10800 //Custo cimentação
12000 //Custo com ROV
6300 //Custo com equipamentos de perfuração
1650 //Custo com equipamentos de pescaria
12000 //Custo com mud logging
45000 //Custo com MWD, LWD, etc
1500 //Custo com telecomunicações
105000 //Custo com pessoal
96000 //Custo com diesel
500000 //Custo aluguel da sonda
  
```

Figure 71 – Conjunto de custos variáveis carregados no simulador

Esse bloco foi desenvolvido para analisar cada uma das rubricas de custo acima listadas individualmente. Como exemplo podemos citar que custos como o de aluguel, previsão climatológica, telecomunicação, entre outros serão computados diariamente, enquanto que outros como cimentação, por exemplo, só será computado em ocasiões de assentamento dos revestimentos. Na tabela abaixo encontra-se todas as 32 rubricas de custo, bem como em qual momento que ele que o simulador considerará a mesma no computo do custo total.

Rubrica de custo	Aplicação
Custo de controle e qualidade	Pagamento fixo no início da atividade
Custo de posicionamento dinâmico	Pagamento fixo no início da atividade
Custo de equipamentos da cabeça do poço	Pagamento fixo no assentamento do wellhead
Custo dos perfis elétricos	Pagamento fixo no início da atividade
Custos com testes de formação	Pagamento fixo no início da atividade
Custos com HSE	Pagamento fixo no início da atividade
Custos de seguro	Pagamento fixo no início da atividade
Custos com serviço legal	Pagamento fixo no início da atividade
Custos com revestimentos	Pagamento fixo no início da atividade

Custo com cimento	Pagamento fixo em cada fase de assentamento, por barril utilizado
Custo com materiais para fluido de perfuração	Pagamento fixo no início da atividade
Custo fixo por broca	Pagamento fixo sempre que há troca de broca
Custo com helicóptero	Pagamento variável em diária
Custo com base de apoio	Pagamento variável em diária
Custo com barcos de apoio	Pagamento variável em diária
Custo com gerência de resíduos	Pagamento variável em diária
Custo com descida de revestimentos	Pagamento variável durante o tempo de assentamento do revestimento
Custo com procedimento de cimentação	Pagamento variável durante o tempo de assentamento do revestimento
Custo com ROV	Pagamento variável em diária
Custo com equipamentos de perfuração	Pagamento variável em diária
Custo com <i>mud logging</i>	Pagamento variável em diária
Custo com equipamentos de medição	Pagamento variável em diária
Custo com telecomunicações	Pagamento variável em diária
Custo com pessoal	Pagamento variável em diária
Custo com diesel	Pagamento variável em diária
Custo de aluguel da sonda	Pagamento variável em diária
Custo da cabeça do poço	Pagamento fixo quando for assentar o wellhead
Custo com previsão climatológicas	Pagamento variável em diária
Custo com equipamentos de pesca	Pagamento variável em diária

Tabela 6 – Tabela de custos considerados e como foram vistos e manipulados pelo simulador

A partir da tabela acima foi implementado no simulador uma expressão para estimação dos custos, dada esquematicamente por:

$$CT = C_{fixo} + C_{var} * t_{total} + C^* \quad (117)$$

Nessas formulações temos que:

- CT custo total da perfuração do poço

- C_{fixo} componente do custo fixo, descrita como “Pagamento fixo no início da atividade”, na tabela 6
- C_{var} componente do custo variável, descrita como “Pagamento variável em diária”, na tabela 6
- C^* componente do custo momentâneo, descrito como qualquer rubrica que seja cobrado uma ou mais vezes e que não se encaixem nem em custos fixos, tampouco variáveis

4.3.11. Resultados Fornecidos pelo Módulo

Após a finalização da implementação do Bloco 14 está encerrada a atuação do módulo de planejamento de poços. Nesse ponto o módulo fornecerá ao usuário um conjunto de 6 arquivos, sendo 5 gráficos e 1 relatório, contendo toda a informação processada e apresentada ao longo desse capítulo. Todos esses arquivos poderão ser vistos e avaliados com maior cuidado no capítulo 6, entretanto vale adiantar que conterão:

- Gráfico da janela operacional e o profundidades das sapatas a serem assentadas, no arquivo UFRJDRILLINGSIMULATOR/OUTPUTS/OUTPUTCRITGRADIENTES.PNG
- Diagrama tempo por profundidade estimada, considerando não haver eventos indesejados, em UFRJDRILLINGSIMULATOR/OUTPUTS/OUTPUTTIMEDEPTHDIAGRAM.PNG
- Diagrama esquemático da visão frontal do poço a ser construído, contendo as profundidades das sapatas, a altura da coluna de cimento em cada fase, os revestimentos, sejam liners ou colunas, e a distribuição de diâmetros, no arquivo UFRJDRILLINGSIMULATOR/OUTPUTS/OUTPUTWELLPROFILE.PNG
- Diagrama esquemático de uma visão superior do poço, ilustrando a composição de diâmetros, bem como a espessura de cada um, no arquivo UFRJDRILLINGSIMULATOR/OUTPUTS/OUTPUTWELLTOPVIEW.PNG
- Diagrama custo por profundidade estimada para o poço no arquivo UFRJDRILLINGSIMULATOR/OUTPUTS/OUTPUTCRITGRADIENTES.PNG

- Relatório do poço projetado, contendo informações sobre cada fase a ser perfurada (como diâmetro da fase, do revestimento, profundidade das sapatas, densidade de lama recomendada, volume de cimento, tempo para perfurar, número de brocas utilizadas, etc.), além de um perfil das brocas utilizadas, qual seção elas conseguiram perfurar, por quanto tempo, etc. Também é apresentado um resumo da distribuição do tempo total gasto na perfuração ao longo das quatro parcelas consideradas no simulador, em
UFRJDRILLINGSIMULATOR/OUTPUTS/OUTPUTWELLPLANNING_RESULTS.TXT

Vale observar que, conforme explicado na seção 1.2 desse trabalho todo o simulador foi desenvolvido em C++, que é uma linguagem que não suporta bibliotecas gráficas como padrão. Todos os gráficos, *grids*, figuras, esquemas etc. que esse módulo forneceu precisou ser implementado desde a base, utilizando a biblioteca gráfica GD_Paul.h.

5. Metodologia aplicada no Módulo de Perfuração em Tempo Real

5.1. Introdução

Após a implementação do módulo de planejamento de poço o simulador *proposta* também contém um módulo da simulação da perfuração em tempo real. Nesse módulo o usuário poderá interagir dinamicamente com o programa, alterando o cenário da perfuração e observando o desempenho da mesma através de diversos parâmetros de saída, que são atualizadas conforme o poço avança em direção ao seu objetivo. Além de alterar os cenário da perfuração esse módulo permite ao usuário realizar as intervenções no poço que, no módulo anterior, eram realizadas de maneira automatizada. Como tais intervenções, podemos citar: a parada da perfuração para troca de broca, assentamento dos revestimentos, paradas da perfuração para circulação de *kick*, caso este seja observado, entre vários outros.

Dessa forma os resultados provenientes do módulo de planejamento e elencados a seção 4.3.11 devem servir como um guia para a simulação do poço, no entanto esse módulo permite testarmos novas estratégias de perfuração visando aumentar a taxa de penetração e reduzir o *NPT*. Além disso, a partir da funcionalidade de observação de perdas de circulação e influxos o usuário também pode adotar estratégias que garantam a perfuração de forma segura, sem que tais riscos afetem o poço.

Portanto, o longo desse capítulo veremos a metodologia que foi implementada nesse módulo, assim como os modelos, as funcionalidades, os fluxogramas, arquivos de entrada e tudo que for relevante para tal produto.

5.2. Inicializando o módulo e carregamento de dados

O módulo do simulador em tempo real é inicializado ao escolhermos a Opção 2 (SIMULADOR DINÂMICO EM TEMPO REAL) na tela inicial do simulador (Figura 34). Essa opção pode ser escolhida após terminarmos a implementação do módulo de planejamento de poço. Nesse caso, ao concluirmos todo o planejamento teremos a tela do simulador

apresentando o relatório do planejamento, igual ao arquivo de saída UFRJDRILLING SIMULATOR/OUTPUTS/OUTPUTWELLPLANNING_RESULTS.TXT. Apertando qualquer botão retornaremos a tela inicial, onde o usuário deve escolher a opção 2.

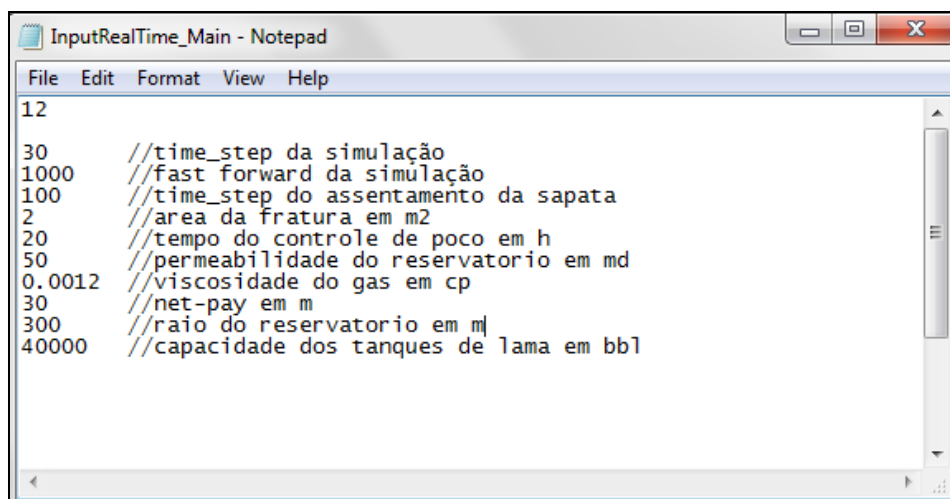
Dessa forma estaremos dentro do simulador de perfuração, onde, a primeira etapa é o carregamento dos dados e a configuração do cenário que será utilizado não início da simulação. A Figura 72 apresenta a tela de entrada do módulo de simulação em tempo real.



```
*****
* REAL-TIME DRILLING SIMULATOR PROJECT - UFRJ
*
* MENU INICIAL:
*
* PRESS 1) DEFINIR PARAMETROS DE FLUIDO
* PRESS 2) DEFINIR PARAMETROS DOS TUBOS
* PRESS 3) DEFINIR PARAMETROS DA BROCA
* PRESS 4) INICIAR PROCESSO DE ASSENTAMENTO DE SAPATA
* PRESS 5) COMPLETAR O POCO COM LAMA DE PERFURACAO
* PRESS 6) INICIAR PROCESSO DE CONTROLE DE POCO
* PRESS 7) RETORNAR A SIMULACAO
*
* PRESS 8) INICIAR SIMULACAO
*
* OPCAO: _
```

Figura 72 – Tela inicial do módulo de simulação em tempo real

Ao inicializar o módulo de simulação o programa carregará, independentemente da atuação do usuário, um arquivo de entrada inicial contendo diversos dados relativos a atividade. Tais dados deverão estar devidamente escritos no arquivo de entrada UFRJDRILLINGSIMULATOR/INPUTS/INPUTREALTIME_MAIN.TXT, como mostra a Figura 73. Dentre tais variáveis podemos observar o *time step* no qual o simulador irá interagir, atualizando os resultados da perfuração, o valor da aceleração do processo, ou seja, o qual mais rápido desejamos acompanhar a perfuração frente ao que ela de fato progride, entre outros.



```
12
30 //time_step da simulação
1000 //fast forward da simulação
100 //time_step do assentamento da sapata
2 //area da fratura em m2
20 //tempo do controle de poço em h
50 //permeabilidade do reservatorio em md
0.0012 //viscosidade do gas em cp
30 //net-pay em m
300 //raio do reservatorio em m
40000 //capacidade dos tanques de lama em bbl
```

Figura 73 – Arquivo UFRJDRILLINGSIMULATOR/INPUTS/INPUTREALTIME_MAIN.TXT

Além disso, caso o módulo de planejamento de poço não tenha sido acionado, o módulo carregará o arquivo contendo os parâmetros de calibração do modelo de Bourgoyne-Young. Os perfis de gradiente de poros e gradiente de fratura serão automaticamente carregados a partir dos resultados do módulo de planejamento.

Na sequência caberá ao usuário carregar os dados referentes ao cenário da perfuração. Para armazenar tais dados foram implementadas três bibliotecas que compõem o simulador. Cada uma delas será responsável por armazenar os dados referentes ao fluido de perfuração utilizado, a coluna de perfuração utilizada (dados de *drill pipes* e do *BHA*) e os dados da broca. As seções seguintes apresentarão como tais carregamentos devem ser feitos.

5.2.1. Dados do fluido de perfuração

No módulo de planejamento o fluido de perfuração foi modelado como sendo newtoniano de viscosidade constante e densidade variável conforme o previsto pelo critério de assentamento de sapatas. Já nesse módulo o fluido de perfuração poderá ser caracterizado dentre três possíveis modelos reológicos, com suas *propriedades* reológicas significativas ajustadas através dos resultados no viscosímetro rotacional, além da densidade. Tais modelo reológicos considerados serão: (i) modelo newtoniano, (ii) modelo plástico de Bingham e (iii) modelo de lei de potências.

Na tela inicial temos a opção DEFINIR PARÂMETROS DOS FLUIDOS. Escolhendo essa opção o programa perguntará pelo modelo reológico do fluido. É importante notar que o usuário deve, obrigatoriamente, respeitar as opções dadas pelo simulador. Caso contrário o modelo não será reconhecido. Em seguida são solicitados os valores lidos no viscosímetro rotacional θ_{300} e θ_{600} . O módulo calculará, a partir desses valores, os parâmetros reológicos significativos do fluido. As equações que regem tal relação são dadas por:

$$(\mu_a)_{newton} = \theta_{300} \quad (118)$$

$$(\mu_p)_{bingham} = \theta_{600} - \theta_{300} \quad (119)$$

$$(\tau_0)_{bingham} = \theta_{300} - (\mu_p)_{bingham} \quad (120)$$

$$(n)_{power\ law} = 3.32 \log\left(\frac{\theta_{600}}{\theta_{300}}\right) \quad (121)$$

$$(K)_{power\ law} = \frac{510 \theta_{300}}{511^n} \quad (122)$$

Na sequência o usuário deverá definir a densidade do fluido em questão e a vazão com a qual o mesmo será injetado no poço.

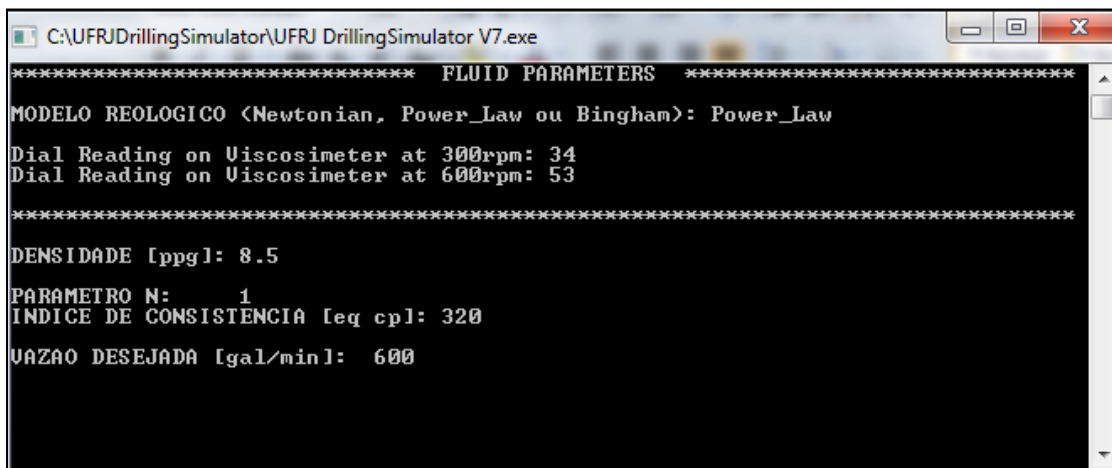


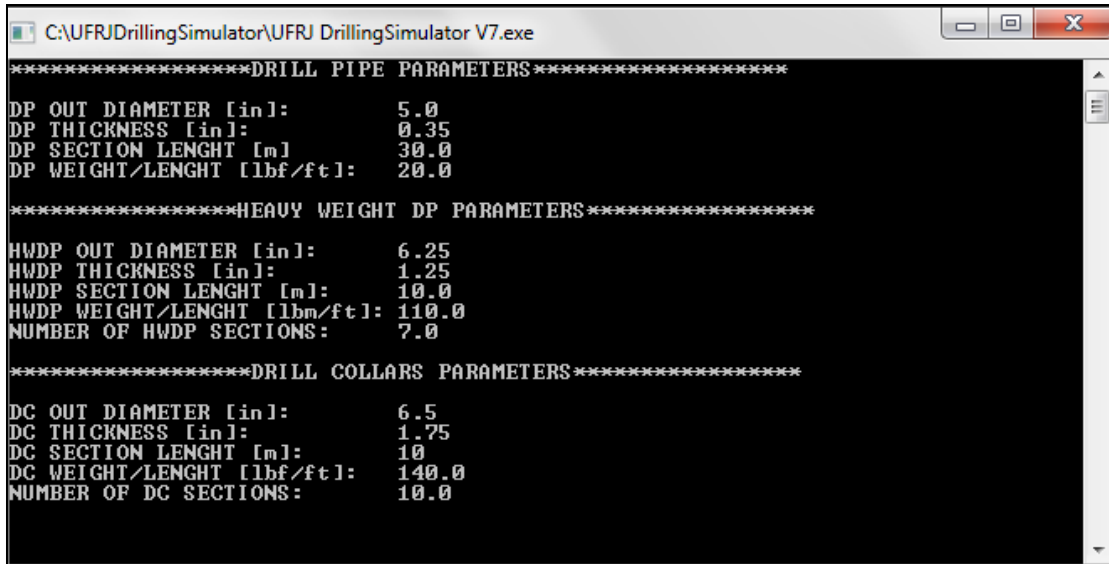
Figura 74 – Tela do carregamento de dados de fluido

5.2.2. Dados da coluna de perfuração

O usuário também deverá carregar os dados referente a coluna de perfuração. O simulador considera que essa coluna será formada apenas por *drill pipes*, *heavy weight drill pipes* e *drill collars*, que o usuário deverá dimensionar.

Ao optar pela opção 2 na tela inicial (DEFINIR PARÂMETROS DOS TUBOS) o usuário será levado a tela de carregamento de dados relativos a coluna. O programa perguntará se o usuário deseja carregar novos dados para o *drill pipe*, uma vez que considera-se que ao longo da perfuração apenas a composição do *BHA* viria a ser modificada para alterar o peso sobre a broca e manter a linha neutra nos comandos. Caso o carregamento nunca tenha sido feito ou deseja ser alterado o usuário deve responder afirmativamente a pergunta. Caso contrário a última configuração será a válida.

Daí o simulador perguntará por quatro parâmetros do *drill pipe*: (i) diâmetro externo, (ii) espessura, (iii) comprimento da seção e (iv) peso específico. Já para os *heavy-weight drill pipes* e os *drill collars* o simulador perguntará por cinco parâmetros: (i) diâmetro externo, (ii) espessura, (iii) comprimento da seção, (iv) peso específico da seção e (v) número de seções utilizadas. Caberá ao usuário o correto dimensionamento do *BHA* de forma a garantir que a linha neutra da coluna passe pelos comandos. O simulador supracitado não se propõe a analisar a física de coluna, logo o efeito de flambagem sobre os *drill pipes* não será sentida no nível da simulação. Na Figura 75 podemos observar a tela com os dados dos componentes da coluna sendo carregados.



```
C:\UFRJDrillingSimulator\UFRJ DrillingSimulator V7.exe
*****DRILL PIPE PARAMETERS*****
DP OUT DIAMETER [in]:      5.0
DP THICKNESS [in]:        0.35
DP SECTION LENGHT [m]:    30.0
DP WEIGHT/LENGHT [lbf/ft]: 20.0

*****HEAVY WEIGHT DP PARAMETERS*****
HWDP OUT DIAMETER [in]:    6.25
HWDP THICKNESS [in]:      1.25
HWDP SECTION LENGHT [m]:  10.0
HWDP WEIGHT/LENGHT [lbm/ft]: 110.0
NUMBER OF HWDP SECTIONS:  7.0

*****DRILL COLLARS PARAMETERS*****
DC OUT DIAMETER [in]:     6.5
DC THICKNESS [in]:       1.75
DC SECTION LENGHT [m]:   10
DC WEIGHT/LENGHT [lbf/ft]: 140.0
NUMBER OF DC SECTIONS:   10.0
```

Figura 75 – Tela de carregamento dos parâmetros da coluna

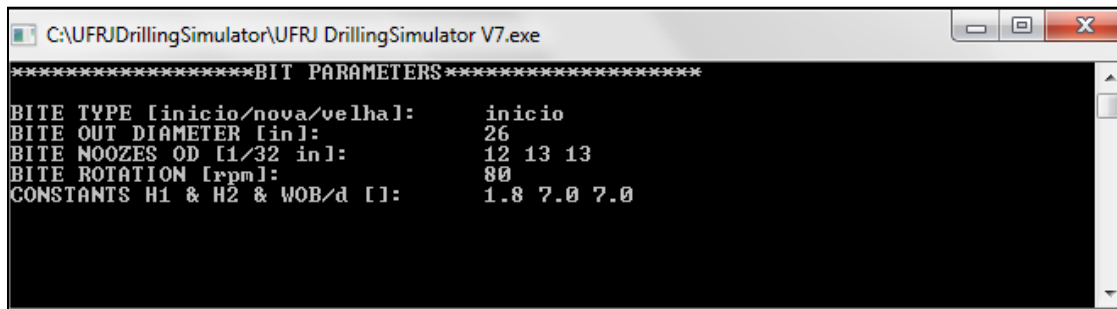
A partir desses dados o simulador calculará o peso sobre a broca a partir da configuração do BHA dada acima, através da Eq. 96.

5.2.3. Dados da broca

Os últimos que devem ser carregados para compor o cenário inicial da simulação são os parâmetros de broca. Estes devem ser carregados pelo usuário ao selecionar a opção DEFINIR PARÂMETROS DE BROCA na tela inicial do módulo.

O simulador trabalha apenas com brocas tricônicas e dividirá as diversas brocas utilizadas em três grandes grupos para manipulação: (i) início, (ii) nova e (iii) velha. O que o simulador reconhece por broca primeira é aquela broca nova que iniciará a perfuração de uma nova fase do poço. Brocas novas são brocas que irão substituir outras já desgastadas durante a perfuração de uma mesma fase. Já brocas velhas são brocas que não serão substituídas, porém, algum parâmetro que é carregado nessa tela será alterado pelo usuário, sem impactar na troca de broca. É necessário fazer tal distinção devido à manipulação dos parâmetros no que diz respeito ao tempo de perfuração, bem como outros parâmetros que serão analisados durante a manobra, como veremos no decorrer desse capítulo.

Além da escolha da classe de broca que a atual broca pertence caberá ao usuário carregar na tela em questão outros parâmetros relevantes a mesma como: (i) diâmetro da broca, (ii) diâmetro dos jatos da broca, (iii) rotação sobre a broca e (iv) parâmetros relativos ao desgaste da broca $H_1, H_2, (W/d)_m$. Na tela abaixo podemos observar o carregamento de dados da broca para a simulação inicial.



```
*****BIT PARAMETERS*****
BITE TYPE [inicio/nova/velha]: inicio
BITE OUT DIAMETER [in]: 26
BITE NOOZES OD [1/32 in]: 12 13 13
BITE ROTATION [rpm]: 80
CONSTANTS H1 & H2 & WOB/d []: 1.8 7.0 7.0
```

Figura 76 – Tela de carregamento dos parâmetros da broca

5.3. Interrelação e Manipulação das Variáveis

Após o carregamento dos dados e configuração do cenário inicial de perfuração todas as condições para o início da simulação já estão satisfeitas. Dessa forma o usuário poderá, ao selecionar a opção 8 da tela inicial (INICIAR SIMULAÇÃO) dar a partida na simulação.

A estrutura computacional implementada é interativa, de forma que a alteração de um parâmetro poderá impactar em outro, que por sua vez poderia impactar num terceiro ou mesmo no anterior, e daí por diante, dependendo do modelo considerado. Essa estrutura de dados é conhecida como cíclica. Além dessa o simulador também trabalha com estrutura de dados contínua, onde o fluxo é direto.

Na tabela a seguir podemos observar, de maneira geral, como o simulador enxerga cada variável de entrada e de saída e como cada uma dessas variáveis influencia e é influenciada pelas demais. Só constam na tabela as variáveis de entrada que podem ser alteradas pelo usuário durante a perfuração.

Variáveis de Entrada	São influenciadas por:	Influenciam diretamente:
Modelo reológico	-----	<i>BHP, Pp</i>
Parâmetros reológicos	-----	<i>BHP, Pp</i>
Densidade	-----	<i>BHP, WOH, WOB</i>
Vazão	-----	<i>BHP, ROP, POTp</i>
OD e δ dos componentes da coluna	-----	<i>BHP, WOB</i>
Comprimento e peso específico das seções dos componentes da coluna	-----	<i>BHP, WOB, WOH</i>
OD da broca	-----	<i>ROP</i>
Dimensão dos jatos da broca	-----	<i>ROP, BHP</i>
Rotação da broca	-----	Desgaste da broca, <i>ROP</i>
Parâmetros da broca	-----	Desgaste da broca
Peso sobre a broca	Comprimento e peso específico das seções dos componentes da coluna, densidade	Desgaste da broca, <i>ROP</i>
Variáveis de Saída	São influenciadas por:	Influenciam diretamente:
Profundidade	<i>ROP</i>	<i>ROP</i> , volume de lama no poço
Pressão de fundo	Reologia do fluido, geometria da coluna, vazão, pressão de bombeio, densidade, volume de lama no poço	<i>ROP, Vkick, Vperdas</i>
Pressão de bombeio	Reologia do fluido, geometria da coluna, vazão, densidade, profundidade	<i>BHP</i>
Potência da bomba	pressão bombeio, vazão	-----
Desgaste na broca	<i>WOB</i> , rotação, parâmetros da broca, litologia	<i>ROP</i>
Taxa de Penetração	<i>BHP</i> , profundidade, litologia, <i>Gp</i> , rotação, <i>WOB</i>	Profundidade
Gradiente de Poros	-----	<i>ROP</i>

Gradiente de fratura	-----	<i>ROP</i>
Peso sobre o gancho	Geometria e peso específico da coluna, densidade	-----
Volume de kick	<i>Gp, BHP</i>	Volume nos tanques
Volume de perdas	<i>Gf, BHP</i>	Volume no poço
Volume de lama no poço	Geometria do poço, geometria da coluna, <i>Vperdas</i> , profundidade	<i>BHP</i>
Volume de lama nos tanques	<i>ROP</i> , vazão, geometria do poço, <i>Vkick</i>	-----

Tabela 7 – Tabela mostrando as relações entre as variáveis do problema

Nesse sentido podemos observar o quão complexa pode ser uma análise de perfuração em tempo real. Embora a ferramenta supracitada apresente algumas simplificações na sua implementação, na qual a principal delas pode ser estar analisando cada um dos parâmetros relevantes a perfuração de maneira permanente, ou seja, independente do tempo, a modelagem ainda se faz por demasiado complexa.

Como forma de contornar as relações cíclicas entre as variáveis foi implementado a capacidade de analisar o problema para intervalos temporais pré-definidos como *time steps*. Para cada *time step* o modelo implementado será executado, utilizando os valores das variáveis computadas no tempo imediatamente anterior. Assim sendo, caso as variáveis sejam cíclicas, como, por exemplo, o volume de lama no poço, que, ao mesmo tempo em que é função do *BHP* também influencia nesse *BHP*, utilizaremos a informação anterior ao tempo de análise para decidir com qual valor trabalhar. Em outras palavras se, no tempo t_{i-1} o volume de lama no poço for um volume específico, no tempo t utilizaremos essa informação para estimar o *BHP*. Em seguida, a partir do *BHP* no tempo t estimaremos o novo volume de lama no poço que será utilizado no tempo seguinte, e daí sucessivamente. Assim sendo podemos observar que a qualidade do modelo de simulação será proporcional ao intervalo de tempo *time step* utilizado.

Já nas estruturas contínuas o fluxo de dados é direto, com todos os valores sendo analisados dentro do mesmo tempo t . Como, por exemplo, podemos citar a relação entre desgaste na broca e taxa de penetração, onde, para um mesmo tempo, é calculado o primeiro e, daí, o segundo, se repetindo o processo para o novo tempo t sem carregarmos os valores anteriores.

Além disso também é permitido ao usuário alterar o cenário de perfuração de maneira direta durante a atividade. Este poderá atuar no simulador tanto alterando qualquer dos parâmetros apresentado na seção 5.2 quanto iniciando uma manobra ou assentamento de sapatas, além de iniciar a circulação de um *kick* ou completar o nível de lama no poço. Essa atuação ocorre de maneira direta, onde, ao término de uma interação no simulador (tempo t) o usuário poderá atuar na perfuração através de teclas de atalho no teclado, que redirecionarão para uma determinada tela de entrada de dados. Assim, a partir do tempo t_{i-1} o cenário de perfuração já estará modificado.

Tais estruturas devem apresentar o arranjo esquemático que rege o funcionamento do simulador. Os cálculos das variáveis de saída, por sua vez, serão realizados através de blocos internos, que são modelos implementados computacionalmente e estruturados para fornecer uma resposta a partir de um conjunto de entradas. A figura seguir apresentará tal esquema.

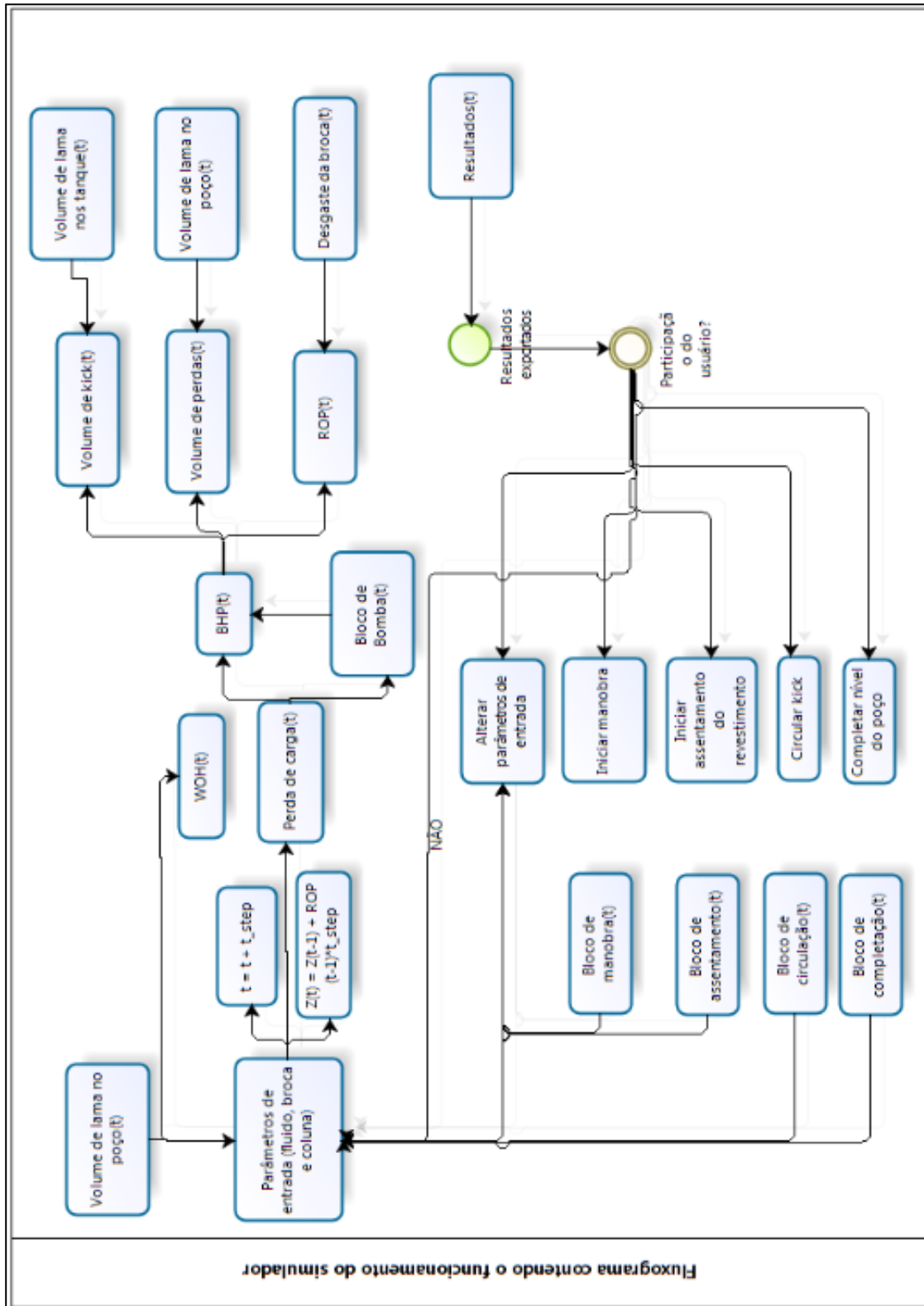


Figura 77 – Fluxograma básico do funcionamento do módulo de simulação em tempo real (elaboração própria)

5.4. Blocos Internos do Módulo de Simulação

Os blocos internos, como introduzidos na seção anterior, compõem os algoritmos desenvolvidos, baseado nos modelos da literatura, para calcular as variáveis de interesse do processo de perfuração do poço. A disposição dos blocos no simulador foi definida pela estrutura de dados que o mesmo deve lidar, como apresentado na Figura 77. Dessa forma apresentaremos, ao longo dessa seção, todos os blocos internos que compõem este módulo.

5.4.1. Bloco de Hidráulica

Nesse bloco será implementado todo o modelo de hidráulica para o poço, onde calcularemos a perda de carga ao longo da coluna, a perda de carga no anular, na broca, a pressão e potência de bombeio, pressão de fundo de poço e *ECD*. Para tal o simulador é capaz de modelar fluidos tanto newtonianos, quanto binghamianos ou que seguem a lei de potências, a partir dos dados carregados pelo usuário, como apresentado na seção 5.2. Para compreender a modelagem computacional desse bloco dividiremos o mesmo em sub-rotinas, que serão listadas na sequência.

5.4.1.1. Perda de carga ao longo do poço

Ao calcularmos a perda de carga ao poço podemos obter o perfil de hidráulica ao longo de todo o comprimento, desde a superfície até a profundidade $Z(t)$. Para tal implementaremos computacionalmente todo o modelo de hidráulica que foi descrito na seção 3.2, para o modelo reológico em questão, considerando desprezível a rugosidade da parede, a nível de simplificação.

O cálculo da perda de carga no simulador será discretizado, fundamentalmente, em 8 seções ao longo do poço, sendo 3 na coluna, 1 na broca e 4 no anular. Essa discretização se deu basicamente em função das características geométricas de cada uma das seções, que levaria a valores distintos para as variáveis representativas do escoamento, bem como da perda de carga. As seções representativas podem ser observadas na Figura, abaixo.

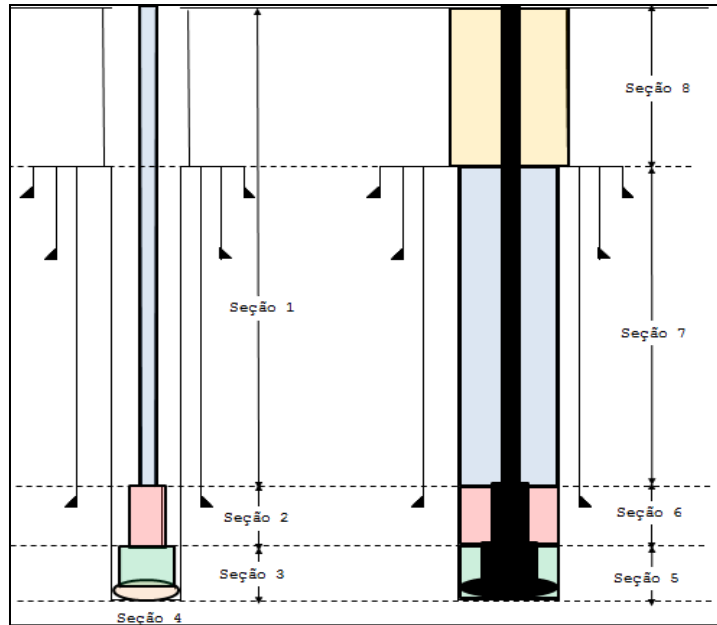


Figura 78 – Esquema da divisão das seções para cálculo da perda de carga (elaboração própria)

Para a análise do escoamento em cada seção a metodologia é a mesma, só diferenciando para o caso de dentro da coluna ou no anular. A única diferença entre o cálculo nas seções será as características geométricas de cada uma, como diâmetros interno e externo. Podemos observar na figura acima que a seção 5 não considera a altura da broca. Na tabela a seguir podemos verificar quais as características de cada seção.

Seção	OD	ID	Comprimento
1	-----	<i>Drill pipe</i>	$Z_{well} - (n_{hwdp} * L_{hwdp} + n_{dc} * L_{dc})$
2	-----	<i>Heavy-weight drill pipe</i>	$n_{hwdp} * L_{hwdp}$
3	-----	<i>Drill Collar</i>	$n_{dc} * L_{dc}$
5	Fase/Revestimento	<i>Drill Collar</i>	$n_{dc} * L_{dc}$
6	Fase/Revestimento	<i>Heavy-weight drill pipe</i>	$n_{hwdp} * L_{hwdp}$
7	Fase/Revestimento	<i>Drill pipe</i>	$Z_{well} - (n_{hwdp} * L_{hwdp} + n_{dc} * L_{dc}) - LDA$
8	Riser	<i>Drill pipe</i>	LDA

Tabela 8 – Características das seções do poço para cálculo da perda de carga

Como o bloco de hidráulica não tem como objetivo fornecer apenas a magnitude da perda de carga em cada seção, mas sim um perfil de pressão ao longo de toda a coluna e todo o espaço anular, o simulador discretizará cada uma das seções acima (com exceção da seção da broca) em dez pontos, onde serão calculadas as perdas de carga no trecho. Esses dez pontos estarão equidistantes uns dos outros e, no caso das seções 1 e 7, eles deverão ser recalculados a cada interação no simulador.

Para cada ponto de cálculo da perda de carga, discretizado como colocado acima, devemos estimar a perda de carga dP_f/dz , que, como sabemos, é função do modelo reológico do fluido.

A) Fluidos Newtonianos

Quando o fluido de perfuração simulado é newtoniano o cálculo da perda de carga deve ser obtida através da metodologia proposta na seção 3.2.2.5.A. Baseando-se nessa seção foi implementado no simulador uma função que calculará o gradiente da perda de carga no trecho da seção estudada, a partir do conjunto de informações carregada pelo usuário.

O funcionamento dessa função segue, basicamente, uma relação direta entre as variáveis. Porém, caso o escoamento seja turbulento, o fator de fricção que alimenta a Eq. de Fanning (Eq. 72) será dado pela solução da equação de Colebrook (Eq. 74), que é uma equação não linear e, como tal, não apresenta solução analítica. Nesse sentido foi utilizado o modelo numérico de Newton-Raphson, usando como estimativa inicial um valor da mesma ordem de grandeza do f . Nesse caso o modelo numérico implementado será do tipo:

$$f_i = f_{i-1} - \frac{F(f_{i-1})}{F'(f_{i-1})} \quad (123)$$

$$F(f_{i-1}) = 4 \log(Re\sqrt{f_{i-1}}) - 1/\sqrt{f_{i-1}} - 0,395 \quad (124)$$

$$F'(f_{i-1}) = -\frac{2}{Re f_{i-1}^2} + \frac{1}{2\sqrt{f_{i-1}}^3} \quad (125)$$

Nessas formulações temos que:

- $F(f_{i-1})$ função de Colebrook (Eq. 74), aplicada para o f_{i-1}
- $F'(f_{i-1})$ derivada da função de Colebrook, aplicada para f_{i-1}

O critério de convergência assumido para essa função foi da ordem de 10^{-5} .

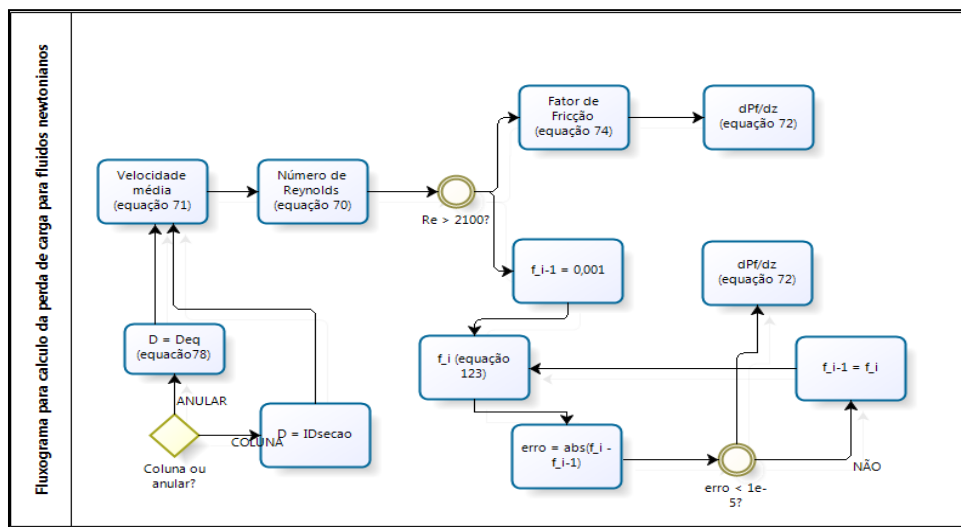


Figura 79 – Fluxograma para cálculo da perda de carga em fluido newtonianos (elaboração própria)

B) Fluidos Binghamianos

Para fluidos que seguem o modelo plástico de Bingham a metodologia de cálculo implementada é a proposta por Hanks *et al.* (1967), utilizando o número de Reynolds crítico como função do número adimensional de Hedstron. Para tal, assim como no caso do anterior, devemos aplicar o método número de Newton-Raphson para resolver o sistema de não linear composto pelas Eq. 81 e Eq. 82, que fornecerá o valor do número de Reynolds crítico para um dado He .

$$x_i = x_{i-1} - \frac{F(x_{i-1})}{F'(x_{i-1})} \quad (126)$$

$$F(x_i) = \frac{He}{16.800} (1 - x_{i-1})^3 \quad (127)$$

$$F'(x_i) = -\frac{3 He}{16.800} (1 - x_{i-1})^2 - 1 \quad (128)$$

Nessas formulações temos que:

- $F(x_{i-1})$ Eq. 81 aplicada para x_{i-1} ;
- $F'(x_{i-1})$ derivada da Eq. 81, aplicada para x_{i-1} ;
- x razão $\frac{\tau_0}{\tau_w}$

Nesse modelo foi utilizado como estimativa inicial uma razão igual a 1, e um critério de convergência da ordem de 10^{-2} . Após obtermos o valor da razão $\frac{\tau_0}{\tau_w}$, podemos obter o valor do Reynolds crítico diretamente através da Eq. 82.

Feito isso, caso o escoamento seja considerado turbulento devemos lançar mão do mesmo modelo para obtermos o fator de fricção que alimentará a equação de Fanning (Eq. 72).

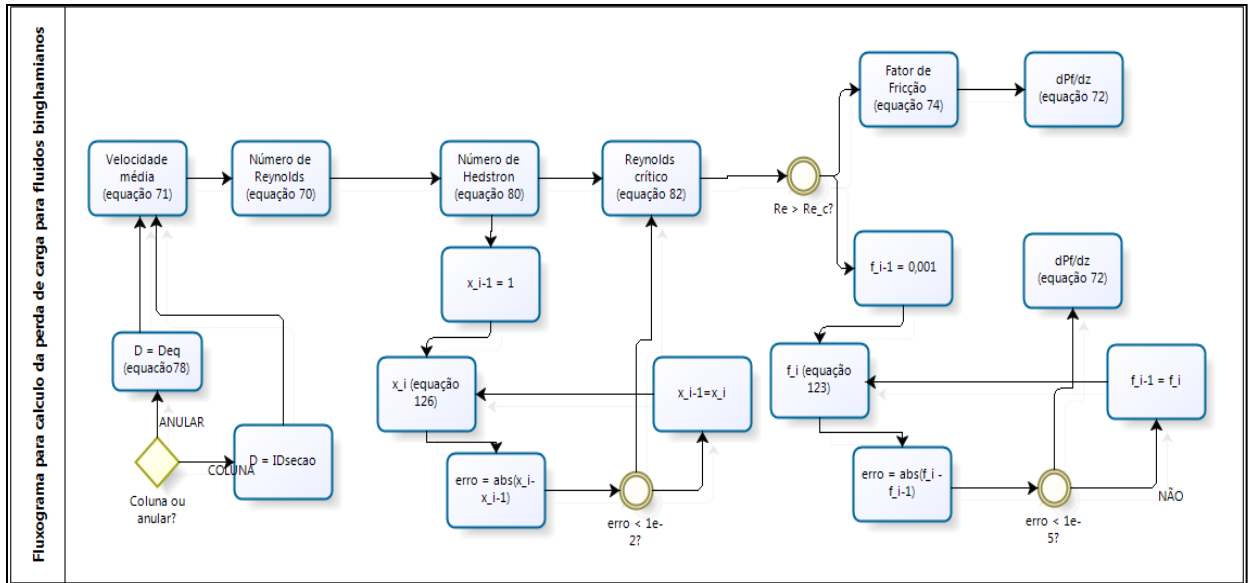


Figura 80 – Fluxograma para cálculo da perda carga para fluidos binghamianos (elaboração própria)

C) Fluidos Lei de Potências

No cálculo da perda de carga em fluidos cujo comportamento é regido pela lei de potências a metodologia implementada é a mesma frente apresentada na seção 3.2.2.5.C. Dessa forma devemos obter um número de Reynolds crítico a partir do gráfico apresentado na Figura 30, e definido o regime de escoamento, o fator de fricção que alimentará a equação de Fanning será dado pela Eq. 74, caso o mesmo seja laminar, ou pela equação de Doedje e Meltzner (Eq. 85), caso seja turbulento.

Para determinarmos numericamente o número de Reynolds crítico como função do índice n foi rodada uma regressão linear simples para verificar se era possível representar essa relação de maneira linear e, caso fosse possível, determinar qual a reta que melhor ajustava os valores curva.

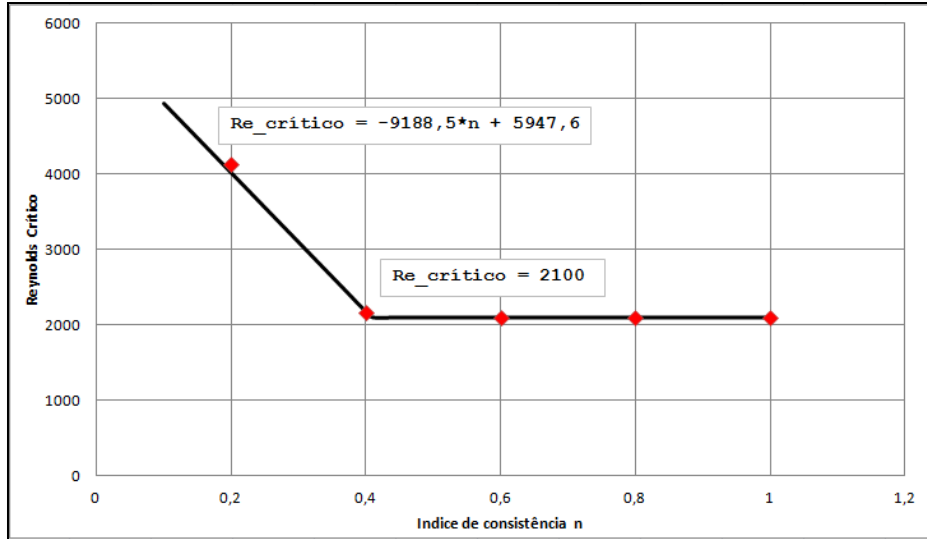


Figura 81 – Ajuste linear do Re crítico em função do parâmetro n (elaboração própria)

A curva ajustada apresentou um $R^2 = 0,995$, mostrando o bom ajuste com os dados extraídos da figura 30. Portanto, aos olhos do simulador, o número de Reynolds crítico será dado pela relação:

$$Re_{crítico} = -9188,5n + 5947,6, \quad se \ Re_{crítico} > 2100 \quad (129)$$

$$Re_{crítico} = 2100, \quad se \ Re_{crítico} < 2100 \quad (130)$$

Definido o valor de Reynolds crítico através das equações acima devemos aplicar, assim como para os demais fluidos estudados, o método numérico de Newton-Raphson para estimação do fator de atrito, quando o escoamento for considerado turbulento, porém, a equação que devemos solucionar passa a ser a equação de Doedge e Meltzner (Eq. 85).

$$f_i = f_{i-1} - \frac{F(f_{i-1})}{F'(f_{i-1})} \quad (131)$$

$$F(f_{i-1}) = \frac{4}{n^{0,75}} \log(Re \ f_{i-1}^{1-n/2}) - \frac{1}{\sqrt{f_{i-1}}} - \frac{0,395}{n^{1,2}} \quad (132)$$

$$F'(f_{i-1}) = \frac{1}{2f_{i-1}^{1,5}} + 4 \frac{1 - 0,5n}{f_{i-1} n^{0,75} \ln 10} \quad (133)$$

Nesse modelo foi utilizado como estimativa inicial igual a 0,001, e um critério de convergência da ordem de 10^{-5} .

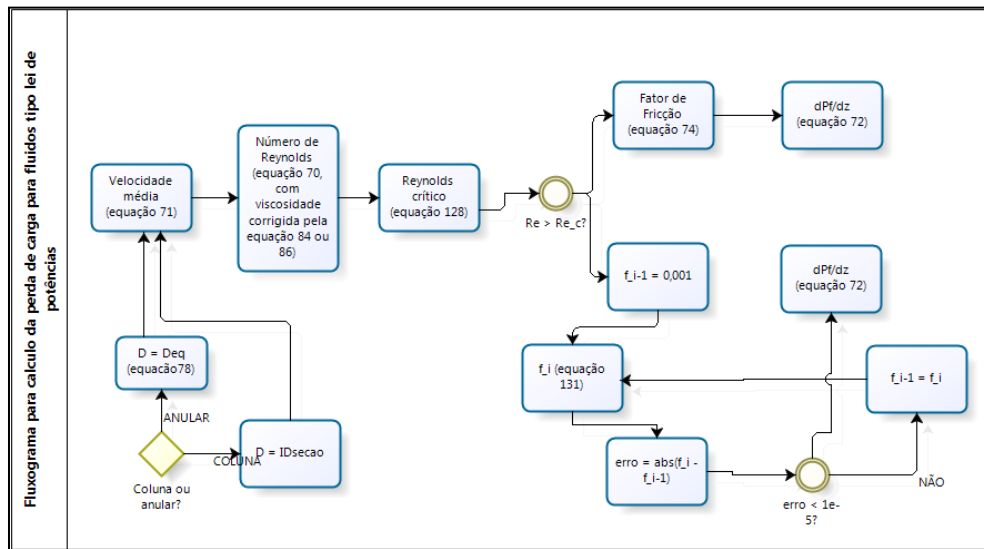


Figura 82 – Fluxograma para cálculo da perda de carga para fluidos de lei de potências (elaboração própria)

Finalmente, de posse do gradiente da perda de carga o bloco de hidráulica deverá calcular a perda de carga ao longo de cada trecho da seção como sendo:

$$\Delta P_{f_i} = \frac{dP_f}{dz} (Z_i - Z_{i-1}) \quad (134)$$

$$\Delta P_{f_{seção}} = \sum \Delta P_{f_i} \quad (135)$$

5.4.1.2. Perda de carga na broca

A perda de carga na broca segue a mesma metodologia proposta na seção 3.2.2.3 deste trabalho. Esse cálculo é feito de maneira direta e é independente do modelo reológico. Está será dada pela Eq. 60, utilizando os dados de entrada definidos pelo usuário conforme designado na seção 5.2.2.

5.4.1.3. Hidráulica da bomba

Conforme apresentado na seção 3.2.2.2 a pressão da bomba deve ser dimensionada de forma a garantir que o fluido vença todas as perdas de carga ao longo do poço e atinja a superfície com uma pressão pré-definida. Para o bloco de hidráulica implementado a pressão fornecida pela bomba será tal que o fluido atinja a superfície com pressão zero, ou seja, a pressão da bomba será dada por:

$$\Delta P_p = \sum_1^8 \Delta P_{f_{seção\ i}} \quad (136)$$

Nesse sentido a potência exigida ao motor da bomba será função da carga e da vazão fornecida ao escoamento, dada, em unidades de campo, pela Eq. 59.

5.4.1.4. Pressão ao longo do poço

Após calcularmos as componentes da perda de carga nas oito seções da coluna o simulador calculará a pressão em cada um dos pontos discretizados ao longo da coluna. Esse calculo será feito a partir do topo dos *drill pipes* na seção 1, e seguirá ao longo das demais seções até o topo da seção 8.

Genericamente o simulador enxergará a pressão ao longo da coluna através da Eq. 55, onde o termo relativo a perda de carga por fricção será definido pela posição em que estamos calculando a pressão. Caso o ponto de interesse esteja nos drill collars esse termos será igual a perda total acima do ponto, ou seja, $\Delta P_{f_{dp}} + \Delta P_{f_{hwd}} + \frac{dP_f}{dz} (Z_i - Z_{hwdp})$. Nesse sentido a pressão de fundo de poço, calculada através da coluna, será dada por:

$$BHP = \Delta P_p + P_h(Z_{well}) - \left[(\Delta P_f)_{dp} + (\Delta P_f)_{hwdp} + (\Delta P_f)_{dc} + (\Delta P_f)_b \right] \quad (137)$$

A partir daí podemos definir a pressão ao longo de todos os pontos discretizados no anular através da Eq. 56. A metodologia do cálculo da perda carga será igual aquela na coluna. A partir do *BHP* calculado acima podemos obter o *ECD* pela Eq. 110. Será permitido ao usuário acompanhar, no tempo de interesse, o comportamento do *ECD* frente aos gradientes geomecânicos do poço. Para tal basta apertar a tecla **x** em qualquer momento na simulação, como mostrado a seguir.

É importante observarmos que o cálculo da pressão ao longo do poço no simulador considera apenas o regime permanente de escoamento. Isso significa dizer que em momentos de alteração na vazão do escoamento (como paradas de circulação, por exemplo) o trecho transiente do escoamento não será considerado. Embora esta seja uma simplificação no modelo observações de campo mostram que uma forma de tornar o modelo considerado mais próximo da realidade é alterar a vazão de maneira gradual, como pequenos degraus.

Quanto a alteração de outros parâmetros de fluido, como densidade, é considerado que esta é *propagada* dentro do poço em um intervalo de tempo inferior ao *time step* utilizado. Essa medida também é consistente com a realidade, uma vez que o fluido é assumido como sendo incompressível.

5.4.2. Bloco de Profundidade por Tempo

O próximo bloco do módulo de simulação em tempo real que foi implementado foi o bloco de tempo por profundidade. Nesse bloco o simulador realizará, a cada interação, os cálculos relativos ao modelo de perfuração, como a taxa de penetração. A partir desse parâmetro o mesmo atualizará a profundidade do poço para um determinado instante de tempo.

Ao mesmo tempo outros fatores afetarão o perfil de profundidade por tempo, como, por exemplo, períodos de manobra e de assentamento de revestimento, como apresentado na seção 4.3.9.2. No entanto, ao contrário do módulo de planejamento de poço, esse bloco

permitira ao usuário iniciar o processo de manobra sempre que ele achar válido, assim como o processo de descida dos revestimentos, sem depender de um critério pré-determinado mas sim da experiência do usuário.

Dessa forma o bloco de profundidade por tempo será composto por três subrotinas que afetam diretamente a relação entre as duas variáveis: (i) modelo de desgaste, (ii) modelo de perfuração, (iii) manobras e (iv) assentamento de revestimento.

5.4.2.1. Modelo de Desgaste

O modelo de desgaste de broca seguirá a mesma metodologia utilizado no módulo de planejamento de poço, com a exceção de que os parâmetros que influenciam esse desgaste podem ser alterados de maneira dinâmica. Além disso, ao contrário do modelo implementado no módulo anterior, a broca poderá continuar atuando mesmo após o seu desgaste chegar a 100%, caso o usuário não inicie o processo de manobra. Isso não era permitido no módulo de planejamento de poço.

5.4.2.2. Modelo de Perfuração

A metodologia implementada no modelo de perfuração será a mesma implementada no módulo de planejamento de poço, salvo o fato de que, nesse módulo a definição dos parâmetros relativos a perfuração ocorre de maneira dinâmica, seja em outros blocos, como é o caso da hidráulica, seja por definição do usuário. Na tabela abaixo podemos identificar a origem dos parâmetros relevantes para o cálculo das oito relações funcionais do modelo.

Relação funcional	Parâmetro relevante	Origem
f_1	-----	-----
f_2	Z	Calculado em cada interação
f_3	Gp	Calculado no módulo de planejamento de poço
f_4	ECD	Calculado para cada interação
f_5	WOB	Usuário

f_6	N	Usuário
f_7	h	Calculado para cada interação
f_8	F_j	Calculado para cada interação

Tabela 9 – Origem da variáveis relevantes para o modelo de Bourgoyne-Young

Podemos notar que a atuação do usuário no modelo de perfuração se dá de forma ampla. Em algumas das relações funcionais essa atuação é direta (como no caso da atuação sobre o WOB ou o N) e em outros casos é indireta, como nas relações funcionais que dependem dos resultados da hidráulica. Essa atuação desempenha papel fundamental para determinação do ROP , uma vez que, como mostrado anteriormente, muitos dos dados trabalhados no modelo são cíclicos e, muitas vezes, atuam diretamente e indiretamente no mesmo parâmetro. Esse é o caso do peso sobre a broca e rotação, que, ao mesmo tempo em que contribui positivamente para o ROP também contribui positivamente para o desgaste que por sua vez atua vide relação negativa com o ROP . Ao alterar qualquer dois parâmetros em um tempo t essa rotina atualizará automaticamente o cenário de cálculo para o tempo posterior. Outros casos como esse podem ser diagnosticados analisando os modelos utilizados.

Para um determinado cenário de perfuração em um tempo t qualquer o simulador calculará todos os parâmetros relativos ao modelo de perfuração, como apresentado na tabela acima, e, aplicando as relações do modelo de Bourgoyne-Young e a Eq. 87 o ROP será atualizado para aquele tempo. No entanto esse ROP será utilizado para atualizar a profundidade do poço no instante seguinte de tempo. Essa metodologia é válida pois o simulador considera, como simplificação, que todo o trecho perfurado entre os tempos t_i e t_{i+1} é perfurado com o mesmo ROP . Essa consideração será tão mais próxima da realidade quanto menor for o intervalo temporal considerado. Já a profundidade total será dado pelo somatório da profundidades parciais, ou seja:

$$\Delta Z_i = ROP_i(t_i - t_{i-1}) \quad (138)$$

$$Z(t) = \sum_0^i \Delta Z_i \quad (139)$$

5.4.2.3. Modelo de Manobras

Ao contrário do módulo de planejamento de poço durante a simulação em tempo real o momento das manobras é definido unicamente pelo usuário. Isso significa dizer que o mesmo pode adiar uma manobra, caso considere que não valer a pena trocar a broca naquele instante, ou adiantar a troca, caso deseje um incremento imediato no *ROP*. Em ambos os casos temos uma perda e um ganho associado, cabendo a experiência do usuário definir qual seria a melhor estratégia. A forma de acionar o módulo de manobra, bem como todas as funcionalidades desse módulo, será apresentada na seção 5.6.

Quando o usuário definir por iniciar uma manobra a circulação será interrompida e a vazão e a taxa de penetração serão levadas a zero. Conseqüentemente a pressão de bombeio e a potência da bomba serão zeradas, bem como a perda de carga por fricção e a força hidráulica sobre a formação. Assim sendo o *BHP* será reduzido a pressão hidrostática, assim como o *ECD* será reduzido a densidade do fluido.

O tempo de operação, no entanto, seguirá a mesma metodologia apresentada no módulo anterior, onde considerava-se que o tempo para realização da uma manobra era proporcional ao tempo de retirada de uma seção e a profundidade total da coluna, e considerava-se desprezível o tempo entre a retirada de uma broca e o reinício da descida da coluna (Eq.112), mantendo-se a profundidade constante nesse tempo.

É importante notar que dentro dessa subrotina de manobra estamos considerando que, durante a retirada da coluna o nível de lama no poço é mantido constante, logo, parâmetros como volume de lama no poço e *BHP* são constantes da mesma forma.

Entretanto durante a manobra o peso sobre o gancho, ao contrário de todos os demais parâmetros, é continuamente alterado. Isso deve ao fato de estarmos retirando sucessivamente seções da coluna, o que torna o peso sustentado pelo gancho menor. Portanto

seria esperado que antes do início da manobra, o peso sobre o gancho seja igual ao peso dado pela Eq. 140 abaixo.

$$WOH_i = [(nwL)_{dp} + (nwL)_{hwdp} + (nwL)_{dc}] \alpha - WOB \quad (140)$$

No início da manobra, entretanto, a coluna perderá contato com o fundo, o que significa que o peso medido será apenas o peso da coluna suspensa, sem o termo do *WOB*. Quanto a coluna for retirada, ou seja, ao fim do tempo $t_t = \left(\frac{t_s}{l_s}\right)Z$ o peso deverá ser zero, retornando ao peso original ao fim do tempo $t = 2t_t$. Dessa forma foi implementada nesse bloco um modelo de variação do *WOH* durante a manobra com perfil linear com o tempo, garantindo as condições de contorno citadas acima. Dessa forma, durante o içamento, o peso sobre a broca será descrito por:

$$WOH(t) = (WOH_i + WOB) \left[\frac{Z - (\bar{l}_s/\bar{t}_s)t}{Z} \right], \quad t \leq t_t \quad (141)$$

Como podemos ver, quando $t = 0$ o peso sobre a broca será igual ao peso da coluna suspensa, como esperado. Quando o tempo foi igual a t_t teremos que o *WOH* tenderá a zero, também como esperado.

Para a descida da coluna o mesmo modelo será aplicado, com algumas pequenas alterações. Nesse caso:

$$WOH(t) = (WOH_i + WOB) \left[\frac{(\bar{l}_s/\bar{t}_s)(t - t_t)}{Z} \right], \quad (142)$$

$$t_t < t \leq 2t_t$$

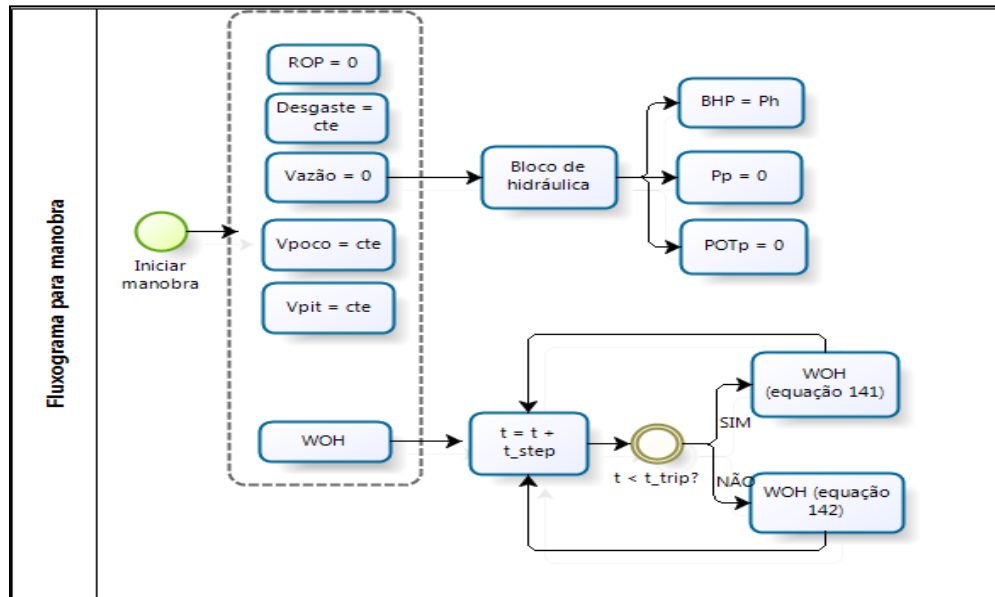


Figura 83 – Fluxograma para manobra no módulo de simulação em tempo real (elaboração própria)

5.4.2.4. Modelo de Assentamento de Revestimentos

Além de definir o momento da troca de brocas, resultando no procedimento de manobras o usuário também terá a opção de assentar o revestimento em qualquer ponto do poço, podendo ou não respeitar as profundidades definidas pelo projeto no módulo anterior. Quando esse procedimento é acionado o poço não avançará até o fim do assentamento.

Ao optar por essa opção, assim como na opção de manobra, a circulação do poço será interrompida, e todos os parâmetros de perfuração serão tratados da mesma maneira que naquele bloco. Vale observar que, aos olhos do simulador o processo de assentamento dos revestimentos nada mais é do que uma etapa de manobra com tempo definido pela Eq. 113, onde o parâmetro peso sobre o gancho (*WOH*) será modificado apenas no início e no fim desse tempo. Em outras palavras, teremos que o tempo do assentamento é distribuído da seguinte forma:

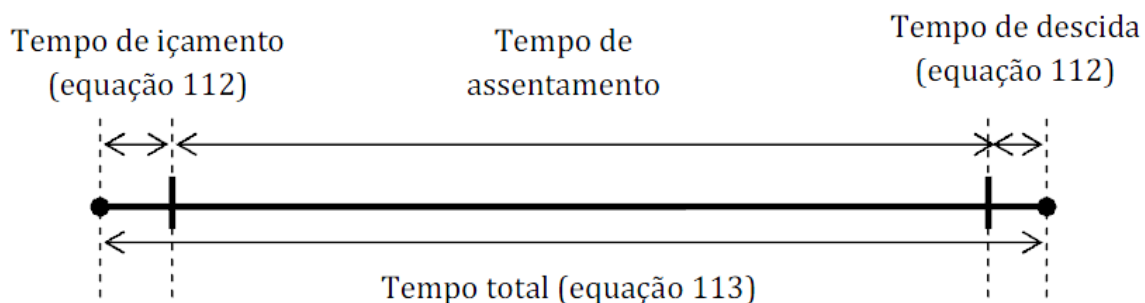


Figura 84 – Esquema de distribuição do tempo no bloco de assentamento dos revestimentos (elaboração própria)

Dessa forma a estimação do peso sobre o gancho seguirá o mesmo modelo aplicado na manobra, porém, para o intervalo de tempo entre $t_t < t \leq t_{assentamento}$ esse parâmetro deverá marcar zero. Observem que há uma grande simplificação aqui, pois o peso sobre a broca nesse trecho deveria medir o peso do revestimento, porém, como consideramos que o tempo intermédio entre tempo de içamento e o tempo de descida compreende não só a descida do revestimento, mas também a descida do cimento, pega do cimento, avaliação da qualidade da cimentação entre outras foi optado por manter o *WOH* igual a zero nesse intervalo.

Ao contrário do bloco anterior, quando o usuário opta por assentar os revestimentos o simulador abrirá uma tela onde este deverá carregar as informações relativas ao revestimento em questão, como podemos observar na figura a seguir.

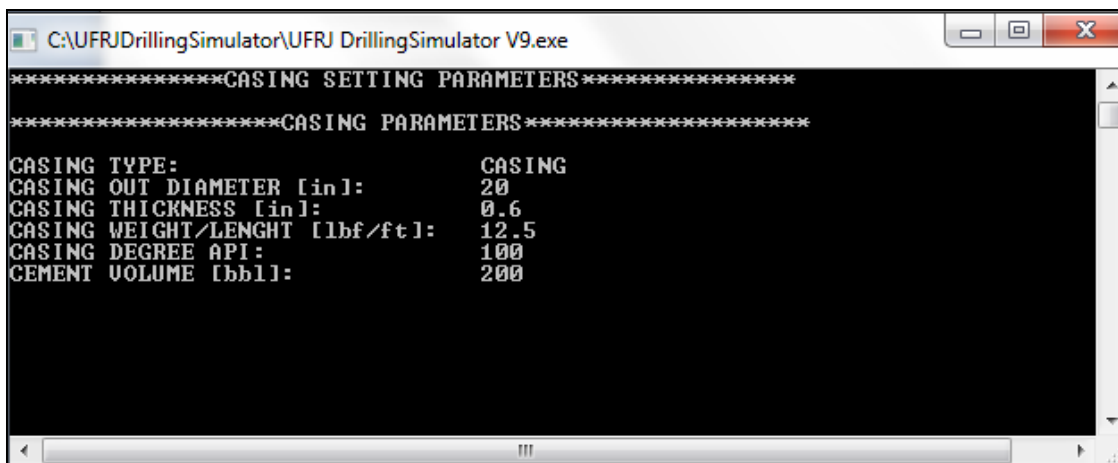


Figura 85 – Tela de carregamento dos dados do revestimento

Esses dados são importantes para compor um banco de dados acerca de cada um dos revestimentos assentados. Com isso o simulador poderia re-projetar o poço ou mesmo estudar a física da coluna em tempo real através de métodos numéricos. Essas módulos não foram implementados no simulador ainda.

5.4.3. Bloco de Estabilidade Poço

Sabemos do capítulo 2 deste trabalho que quando a pressão de fundo não está compreendida dentro da janela operacional do poço algum evento indesejável ocorrerá. Caso a pressão venha a ser inferior a pressão de poros e a formação tenha características que permitam o fluxo teremos a ocorrência de um kick no poço, que pode evoluir para um blowout eventual. Caso a pressão seja superior a pressão de propagação de fratura o fluido tenderá a percolar as fraturas induzidas, reduzindo o nível de lama no poço, no evento que chamamos de perda de circulação. No contexto do simulador as três informações relevantes para uma análise simplificada de estabilidade de poros (apenas considerando a metodologia apresentada na seção 2.2) são calculadas dinamicamente ou acessadas através de ponteiros para dados calculados anteriormente. São elas: a pressão de fundo de poço, a pressão de poros e a pressão de fratura.

De posse desses três parâmetros para cada interação do simulador é possível, através de uma abordagem simplificada, definir quando um desses eventos ocorrerá e, a partir daí, qual

impacto que terá sobre os demais parâmetros da perfuração. Essa funcionalidade permitirá ao usuário verificar, por meio de parâmetros específicos e sensíveis a tais eventos, quanto a perfuração está em risco e assim tomar decisões que venham a mitigar ou mesmo solucionar o problema. Para tal foi implementado computacionalmente os dois riscos de ordem geomecânica citados acima, bem como a funções que o usuário poderá acessar para corrigir tal evento.

5.4.3.1. Modelagem da ocorrência de perdas de circulação

Quando a pressão de fundo de poço for superior a pressão de propagação de fraturas, calculada ainda no módulo de planejamento de poço, ocorrerá o fraturamento da formação. Essa é a premissa fundamental que rege a modelagem das perdas no simulador.

A idéia básica consiste em o usuário fornecer ao simulador a área das fraturas ao redor de poço, que será considerada constante, a nível de simplificação. Daí, conforme o poço avança e um novo trecho ΔZ é perfurado, caso a condição de perdas seja verificada o simulador considera que todo esse trecho estará fraturado e o volume das perdas de fluido será igual ao volume das fraturas. Caso o poço continue avançando sem que o peso de fluido seja corrigido o novo trecho perfurado será fraturado, aumentando ainda mais o volume das perdas de fluido.

Vale observar também que a análise do fraturamento é realizada apenas no novo trecho perfurado. Isso significa que, caso o poço esteja sendo perfurado com um densidade inferior ao gradiente de fratura no fundo, porém, em algum ponto superior não revestido essa densidade é superior ao gradiente de fratura naquele ponto o simulador não será capaz de enxergar perdas de circulação nesta profundidade.

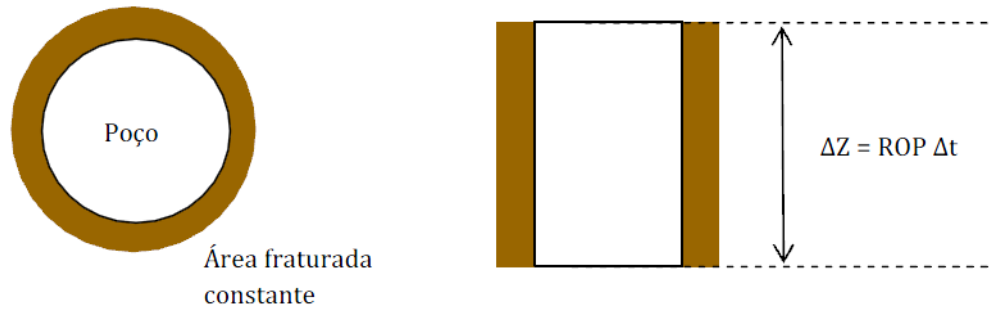


Figura 86 – Diagrama esquemático da zona fraturada para o simulador (elaboração própria)

$$(V_{fraturas})_i = A_{fraturas} \Delta Z_i \quad (143)$$

$$V_{perdas} = \sum (V_{fraturas})_i \quad (144)$$

Nesse sentido, conforme o volume das perdas é aumentado o balanço de massa dentro do poço é afetado, de forma que, caso não haja uma correção do usuário diretamente no volume de lama no poço este terá o seu nível reduzido. O volume de lama no poço, quando não houver perdas, será diretamente proporcional ao avanço do poço e será dado pela equação a seguir:

$$V_{well} = V_{coluna} + V_{anular} \quad (145)$$

$$V_{coluna} = \frac{\pi}{4} (ID_{dp}^2 L_{dp} + ID_{hwdp}^2 L_{hwdp} + ID_{dc}^2 L_{dc}) \quad (146)$$

$$V_{anular} = \frac{\pi}{4} [(OD_{well}^2 - OD_{dp}^2) L_{dp} + (OD_{well}^2 - OD_{hwdp}^2) L_{hwdp} + (OD_{well}^2 - OD_{dc}^2) L_{dc}] \quad (147)$$

Nessas formulações temos que:

- OD_{well} diâmetro da parede do poço, podendo ser igual ao diâmetro da fase, diâmetro do revestimento ou diâmetro do riser.

Todavia, caso seja verificado que o poço terá perdas esse volume de lama será corrigido, sendo igual a:

$$V_{well} = V_{coluna} + V_{anular} - V_{perdas} \quad (148)$$

Com isso o nível do poço também será alterado, o que acarretaria na queda do *BHP*, podendo atingir níveis críticos para o controle de poço. Esse nível, considerado que esse queda não atingiria o comprimento do riser, seria dado por:

$$Z_0 = \frac{V_{perdas}}{\frac{\pi}{4} (ID_{riser}^2 - OD_{dp}^2)} \quad (149)$$

Para corrigir os problemas causados pelas perdas de circulação foi implementada a funcionalidade de completar o volume do poço. Nessa opção basta o usuário defini-la que o nível do poço será instantaneamente corrigido. Porém é importante observar que para que essa medida seja efetiva é necessário que o peso do fluido de perfuração utilizado seja corrigido, uma vez que se não o for a pressão de fundo permanecerá superior a pressão de fraturas e o problema se repetirá.

A modelagem do fraturamento implementado nesse trabalho é extremamente simplificada, uma vez que não é escopo do mesmo desenvolver tal modelagem. Para trabalhos futuros tal caracterização poderia ser desenvolvida, levando em conta modelos de geometria de fratura e velocidade de propagação.

5.4.3.2. Modelagem da ocorrência de *kicks*

Caso a pressão de fundo de poço for menor que a pressão de poros da formação e a litologia perfurada naquele ponto fornecer condições para fluxo de gás podemos afirmar que haverá um influxo de gás para dentro do poço. Essa é a premissa que rege a ocorrência ou não de *kicks* no poço, salvo o fato de que as condições geológicas necessárias para ocorrência

será interpretado apenas como estar perfurando um arenito, ou seja, caso no ponto onde o *BHP* for inferior a pressão de poros e estivermos perfurando um folhelho não haverá influxo, porém, caso a formação seja arenítica, haverá o influxo.

O volume de gás que entrará no poço a cada interação em que for verificada a condição para influxo será modelada pela solução da equação da difusividade hidráulica em geometria radial, para gases ideais em regime permanente. Dessa forma temos que, em unidades de campo, a vazão de gás para dentro do poço será dada por: (Rosa *et al.* 2006)

$$q = (39,763 \cdot 10^{-6})x \frac{kh (P_{pr}^2 - BHP^2)}{\mu \cdot 2 \ln \left(\frac{r_e}{r_w} \right)} \quad (150)$$

Nessas formulações temos que:

- k permeabilidade do reservatório, em *md*;
- μ viscosidade do gás, em *cp*;
- h espessura da formação permeável perfurada, em *ft*;
- r_e limite radial do reservatório, em *in*;
- q vazão do gás na parede do poço, em $10^3 \text{ ft}^3/d$.

Daí o volume de gás no influxo em barris, será dado pela integração da vazão no tempo, ou seja, para um determinado intervalo de tempo Δt , em horas, será dado por:

$$V_{kick} = (2,9525 \cdot 10^{-10})x \frac{kh (P_p^2 - BHP^2)}{\mu \cdot 2 \ln \left(\frac{r_e}{r_w} \right)} \Delta t \quad (151)$$

No contexto do simulador o volume de gás, após entrar no poço, irá migrar como uma bolha ao longo do anular, promovendo um desequilíbrio no balanço de massa dentro poço, já que passará a entrar fluido a partir de uma segunda fonte não considerada. Dessa forma haverá um retorno maior de fluido até a superfície, que será verificado nos tanques.

Em uma condição de perfuração sem *kick* o volume nos tanques será resultado do balanço de massa no poço ao longo do tempo, logo será diretamente proporcional ao *ROP* instantâneo.

$$V_{pit} = V_{pit_0} - V_{well} \quad (152)$$

$$(V_{pit})_i = (V_{pit})_{i-1} - \frac{\pi}{4} [ID_{dc}^2 + (OD_{fase}^2 - OD_{dc}^2)] (ROP \Delta t) \quad (153)$$

Caso ocorra um *kick* será considerado que todo esse volume de gás deslocará igual volume de lama até o tanque, ou seja:

$$(V_{pit})_i = (V_{pit})_{i-1} - \frac{\pi}{4} [ID_{dc}^2 + (OD_{fase}^2 - OD_{dc}^2)] ROP \Delta t + V_{kick} \quad (154)$$

Nesse caso caberá ao usuário identificar a ocorrência do *kick* e atuar de forma corretiva, ou seja, acionando a funcionalidade de circulação do *kick*. Dessa forma estamos interessados apenas em corrigir o influxo de gás no poço e avaliar a perícia do usuário, e não modelarmos toda a circulação *kick*. Quando acionamos tal funcionalidade o programa acrescentará ao tempo de perfuração um tempo de controle de poço, carregado no arquivo de entrada principal desse módulo. É importante que o operador também faça a correção do peso de lama no poço, pois, caso contrário, o problema de *kick* se repetirá.

Vale notar que a implementação da circulação de *kick* é muito mais complexa do que a abordada nesse trabalho, no entanto essa modelagem não é o escopo atual. Todavia a estrutura computacional integrada desta funcionalidade para que uma implementação mais refinada seja feita está pronta.

5.5. Atuação do Usuário na Simulação

Nas seções anteriores foi dito continuamente que o usuário poderia acionar uma ou outra funcionalidade do programa, de acordo com seu interesse, alterar de maneira dinâmica os parâmetros de entrada da perfuração e assim acompanhar os resultados da atividade como um todo. Já foi apresentado como o simulador atua e quais os modelos que regem seu

funcionamento, porém os caminhos através dos quais o usuário pode atuar no seu próprio modelo de perfuração ainda será apresentada nessa seção.

Como apresentado nas seções anteriores o simulador foi desenvolvido com o intuito de permitir ao usuário atuar em qualquer momento da perfuração, com a maior gama de possibilidades possível. Tal gama de possibilidades de atuação é composta por: (i) alteração nos parâmetros de fluido, (ii) alteração na composição da coluna, (iii) alteração nos parâmetros da broca, (iv) manobra, (v) assentamento de revestimentos, (iv) completar nível do poço e (vii) circular o *kick*. Para cada uma dessas funcionalidades existem atalhos no teclado a partir de onde o usuário poderá atuar na simulação de maneira simples. Uma lista desses atalhos, bem como os caminhos através dos quais o operador pode atuar no sistema encontram-se na tabela a seguir:

Atuação	Atalho no teclado	Observações
Ir para tela dos parâmetros de fluido de perfuração	Tecla f	-----
Ir para tela da composição da coluna de perfuração	Tecla t	Só deve ser acionado quando for feita a retirada da coluna. O usuário pode alterar só o BHA ou alterar também o DP
Ir para tela dos parâmetros da broca	Tecla b	Se a broca for classificada como velha o único parâmetro que poderá ser trocado será o N. As demais brocas permitem outras alterações
Iniciar manobra	Tecla b	O atalho levará para carregar dados da broca nova. A manobra será iniciada caso seja escolhida como tipo de broca uma broca nova
Iniciar assentamento de revestimento	Tecla r	Deve-se alterar parâmetros da broca e dos fluidos antes de re-iniciar a simulação
Completar nível de lama no poço	Tecla c	Usuário será transferido para tela de dados do fluido para ajuste

Circular o <i>kick</i>	Tecla w	Usuário será transferido para tela de dados do fluido para ajuste
Ir para tela inicial	Tecla m	-----
Pausar simulação	Tecla p	Deve-se apartar a tecla s para retornar a simulação
Gráfico ECD na janela operacional	Tecla x	O gráfico aparece na tela sem interromper a simulação
Gráfico tempo x profundidade + ROP	Tecla z	O gráfico aparece na tela sem interromper a simulação

Tabela 10 – Caminhos para atuação do usuário durante a simulação

Após atuar no modelo através de quaisquer dos caminhos apresentados acima, com exceção dos dois últimos, o usuário será direcionado até a tela inicial do módulo de simulação. Nessa tela ele poderá definir se mais algum parâmetro deverá ser alterado e, quando satisfeito, basta reiniciar a simulação ao optar pela opção 7 (RETORNAR A SIMULAÇÃO). No ínterim em que tais medidas estão sendo tomadas a simulação estará pausada. Esta será reiniciada já com os novos parâmetros, na mesma profundidade que foi interrompida para atuação do usuário.

5.6. Resultados Fornecidos pelo Módulo

Durante a simulação diversas interações são executadas no domínio do tempo e, para cada uma delas, toda uma metodologia, composta por diversos blocos internos e vários modelos, é aplicada, gerando como resultado uma gama de parâmetros de saída que são resultado da aplicação, nas condições de poço, das variáveis definidas para compor o cenário da perfuração.

De posse desse conjunto de dados estimados ao longo de toda a perfuração diversos gráficos e análises podem ser feitas, contrapondo diferentes parâmetros em diferentes momentos no tempo. Porém, como a implementação gráfica desse simulador não foi trivial optou-se por implementar apenas a funcionalidade de gerar dois gráficos em tempo real para fornecer ao usuário a idéia, ao longo do tempo, do que está acontecendo no poço: (i) diagrama tempo x profundidade e (ii) diagrama de segurança de poço.

Ambos os gráficos podem ser gerados em qualquer momento no poço, porém estes são gerados estaticamente, ou seja, sua atualização não é dinâmica. Após o gráfico ser gerado, para observarmos em um novo momento no tempo devemos solicitar novamente ao programa.

O diagrama de tempo por profundidade é interessante pois permite contrapor os resultados obtidos nesse módulo e aqueles obtidos no módulo de planejamento de poço, de forma a avaliar o quão melhor ou pior foi a estratégia adotado, no que diz respeito ao tempo da perfuração e a taxa de penetração.

Já o diagrama de segurança de poço também é importante, pois permite que o usuário acompanhe ao longo do tempo se a densidade e o *ECD* utilizados estão dentro da janela operacional permitida e ajuda o processo decisório de, por exemplo, aumentar o peso de lama para evitar influxos, a possibilidade de aumentar a vazão, bem como o *ECD*, sem que se tenha problemas de fraturamento em trechos mais rasos não revestidos, ponto para assentar o revestimento, entre outros.

Além desses diagramas, que serão apresentados no próximo capítulo, o módulo também fornecerá um relatório de perfuração que conterà dados atualizados a cada nova profundidade atingida no poço. Esse relatório foi programado para conter informações de parâmetros de entrada ajustados ao longo da simulação, informar quando uma manobra foi acionada, quando um revestimento foi assentado etc. Caso a pressão de fundo seja superior a 95% da pressão de propagação de fratura o relatório exibirá uma mensagem com esse informação. O mesmo vale para quando a pressão for inferior a 105% da pressão de poros. Isso servirá como forma de alerta ao usuário. Quando o desgaste da broca atingir 90% uma mensagem será impressa no relatório. Caso nenhum desses alarmes for impresso no relatório o mesmo exibirá apenas o tempo, a profundidade e a litologia perfurada naquela interação do simulador.

O módulo de simulação em tempo real também fornecerá um arquivo texto de saída contendo todos do dados de saída calculados ao longo da perfuração. No arquivo UFRJDRILLINGSIMULATOR/OUTPUTS/OUTPUTREALTIME_RESULTS.TXT temos, discretizados por tempo, as seguintes informações, em ordem:

- i. Ação (perfurando, manobra ou assentamento dos revestimentos)
- ii. Tempo acumulado de perfuração, em h
- iii. Profundidade total do poço, em m
- iv. Pressão de fundo de poço, em psi
- v. Densidade equivalente de circulação, em ppg
- vi. Pressão de bombeio, em psi
- vii. Potência da bomba, em HP
- viii. Taxa de penetração, em m/h
- ix. Vazão de escoamento, em gal/min
- x. Peso sobre o gancho, em $klbf$
- xi. Volume de lama no poço, em bbl
- xii. Desgaste da broca, adimensional
- xiii. Volume nos tanques de lama, em bbl
- xiv. Volume acumulado das perdas de circulação, em bbl
- xv. Volume da perda de circulação para aquele trecho perfurado, em bbl
- xvi. Volume acumulado de kick, em bbl
- xvii. Gradiente de poros para aquela profundidade, em ppg
- xviii. Gradiente de fratura para aquela profundidade, em ppg
- xix. Litologia para aquela profundidade

Outro arquivo de saída fornecido pelo simulador contém dados referentes ao perfil de pressão ao longo do poço, tanto na coluna quanto no anular. Esse arquivo é composto por três colunas, onde a primeira corresponde a profundidade em que estamos calculando o perfil, a segunda corresponde a pressão dentro da coluna e a terceira representa a pressão no anular. Entretanto os resultados da terceira coluna acompanham o fluxo no anular, ou seja, os resultados foram obtidos e impressos da profundidade objetivo do poço até a superfície.

Por fim o simulador também apresentará na tela do programa diversas variáveis de controle que poderão ser acompanhadas de forma dinâmica. Nessa tela não estarão presentes todas variáveis listadas acima, mas é esperado que o usuário seja capaz de acompanhar o desenvolvimento da perfuração através destas informações.

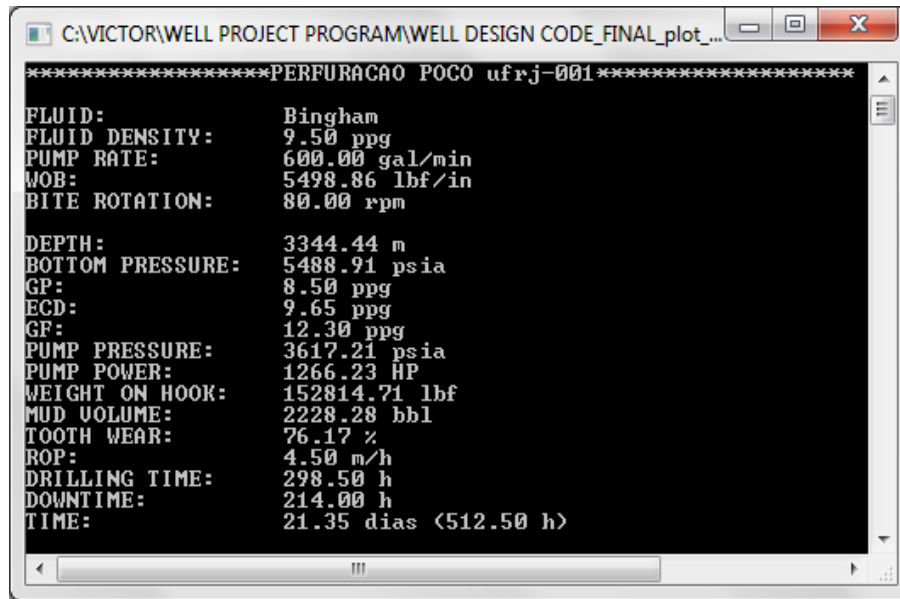


Figura 87 – Tela da simulação, com diversos parâmetros de acompanhamento do poço

Todos os diagramas, arquivos de saída, perfis e resultados de uma maneira geral serão apresentados no capítulo 6.

6. Estudo de Caso, Resultados e Discussões utilizando o Simulador Proposto

Finalmente, após abordamos os modelos teóricos e a metodologia técnica por trás do desenvolvimento do simulador, bem como algumas diretrizes básicas para a sua utilização iremos aplicar o programa desenvolvido, em seus dois módulos, a um caso sintético de perfuração, como todos os dados extraídos ou da literatura ou fornecidos pelo corpo docente do curso ou simplesmente criados, de forma que fossem coerentes ao cenário.

Esse estudo de caso será composto por uma seção descritiva, onde apresentaremos o problema em questão, bem como as variáveis envolvidas, simplificações, entre outras; uma seção de planejamento de poço, onde todos os resultados apresentados ao longo do capítulo 4 deverão ser apresentados; e uma seção de simulação, onde o autor, fazendo as vezes do usuário do programa, criará diferentes cenários de perfuração, como forma a avaliar os resultados fornecidos pelo simulador. Ao longo do capítulo e baseado nos resultados fornecidos pelo programa supracitado ao caso estudado discutiremos a aplicabilidade do simulador, bem como sua capacidade de gerar resultados condizentes com os modelos implementados nos capítulos 4 e 5 deste trabalho.

6.1. Cenário Sintético Proposto para Perfuração

6.1.1. Informações Básicas

Nesse capítulo estudaremos um cenário de perfuração de um poço de desenvolvimento PEGASUS, em ambiente *offshore*. O poço perfurado será vertical, e terá uma lâmina d'água da ordem de 1000 m de profundidade e objetivo final de profundidade 5500 m, contabilizados a partir da mesa rotativa. Não consideraremos o *air gap* para esse cenário, podendo ser desprezível frente os demais parâmetros do problema.

Foi disponibilizado, a partir de correlações estratigráficas com dados de *mud logging* de poços de correlação a coluna estratigráfica esperada para o poço em questão. Nesse cenário estamos considerando que o poço encontrará apenas formações clásticas, particularmente

arenitos consolidados (de forma a reduzir a possibilidade de ocorrência de colapso superior) e folhelhos não reativos, como ilitas ou calcalinitas.

O perfil que se tem disponibilidade para estudar e, a partir daí, gerar o projeto de poço, é um perfil de tempo de trânsito (DT). Esse perfil é produto de um cubo sísmico *pós-stack* gerado após respectivo processamento e inversão dos dados adquiridos por um navio sísmico. O refinamento dos dados de DT ao longo da trajetória do poço é da ordem de 50 m. Além disso, durante a aquisição, um trecho superficial de 200 m não obteve nível de ruído capaz de ser manipulando no processamento, assim este foi descartado da análise. Portanto o perfil de DT terá início na profundidade de 1200 m, estendendo-se até o objetivo do poço, em 5500 m. O gráfico mostrado na Figura 88 apresenta tanto os dados sônicos utilizados no projeto quanto a coluna estratigráfica esperada ao longo da perfuração.

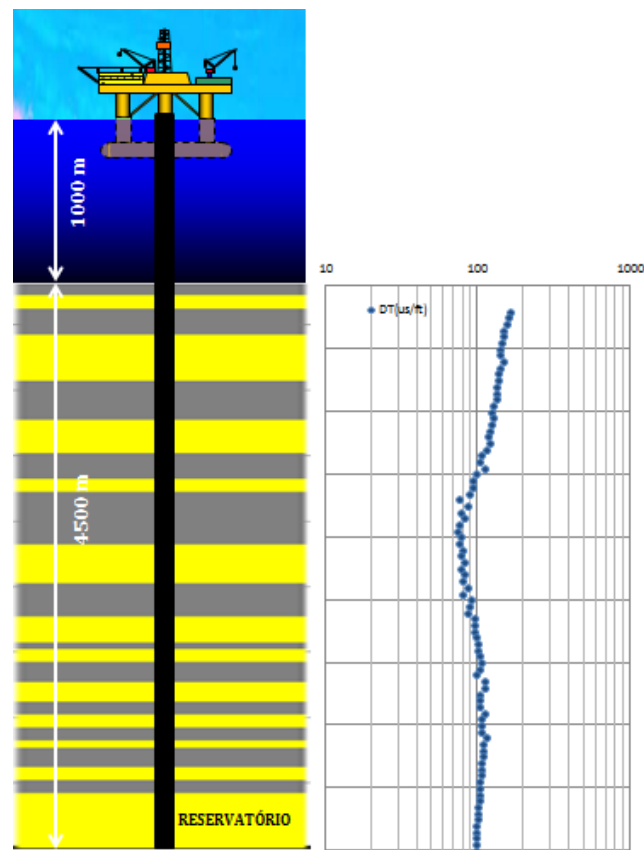


Figura 88 – Diagrama esquemático do poço, contendo as informações originais para projeto (elaboração própria)

O poço não apresenta um perfil de densidade completo, de forma que algum método de estimação deva ser adotado. No entanto foi fornecido os dados de densidades no trecho que compreende o reservatório. Esses dados poderão ser utilizados para calibrar os parâmetros do modelo de estimação escolhido. Além disso não há perfil de geopressões calculados previamente ao estudo no simulador.

O poço será projetado para um revestimento condutor de 30” composto por 6 seções de 12 m cada, e deverá ser jateado na superfície. Além disso após análise da geofísica de subsuperfície não houve a detecção de aquíferos subterrâneos ou zonas de gás raso na primeira lente de areia. Portanto decidiu-se por alojar o revestimento de superfície na profundidade de 500 m abaixo do leito marinho. A partir desse ponto será instalado o BOP *stack*, bem como o riser de perfuração. A pressão de trabalho do BOP ser verificada pelo programa, enquanto que o riser instalado será convencional, de diâmetro interno definido em 19 polegadas.

O ambiente de perfuração é rico em gás sulfídrico⁷, que limita a escolha do tipo de aço para os revestimentos. Segundo Bourgoyne *et al.* (1986) aços tipo L-80 e C-90 apresentam resultados satisfatórios quanto a resistência a corrosão por ácido sulfídrico. Portanto todos os revestimentos que serão usados devem ser compostos ou por aço tipo L-80 ou C-90.

O cimento utilizado para o assentamento dos revestimentos terá densidade constante e igual a 16 *ppg*. Caso haja influxo de gás no poço considera-se que este migrará como uma bolha única pelo anular, com densidade 2,0 *ppg* e viscosidade igual a 0,0012 *cp*. Já o aço utilizado nos revestimentos e na coluna terá densidade de 65,5 *ppg*.

6.1.2. Testes no Poço de Correlação

No modelo considerado utilizaremos dados de dois testes de poço para ajudar a calibrar os resultados da simulação: o *Repeated Formation Test* (RFT) e o *Leak-of-Test* (LOT).

Conforme já apresentado na seção 2.2 esses testes são aplicados respectivamente para calibrar os resultados obtidos no modelo de estimação do gradiente de poros e do gradiente de propagação de fraturas. O RFT foi realizado na região do reservatório, como forma de

⁷ Extremamente tóxico e corrosivo, de fórmula química H₂S

medir a pressão de poros na parede do poço. Já o LOT foi tomado abaixo da sapata dos revestimentos assentados. Para ambos podemos tomarmos tantos dados quanto forem realizados tais testes, mas no cenário em questão o simulador trabalhará com apenas dois dados de RFT e três de LOT. As tabelas Tabela 11 e Tabela 12 apresentam os dados desses testes que serão utilizados na simulação.

Profundidade	Gradiente medido
5200 m	15,3 <i>ppg</i>
5400 m	15,55 <i>ppg</i>

Tabela 11 – Dados de RFT obtidos no poço de correlação

Profundidade	Gradiente medido
1500 m	9,65 <i>ppg</i>
3000 m	12,2 <i>ppg</i>
4500 m	15,6 <i>ppg</i>

Tabela 12 – Dados de LOT obtidos no poço de correlação

6.1.3. Dados do Poço de Correlação

O poço cujos dados estarão alimentando o simulador é um poço localizado na mesma bacia, a uma distância arbitrária do nosso poço de estudo, apresentando a mesma lâmina d'água do poço em questão. Já a profundidade final deste, referenciada a altura da mesa rotativa, será de 5200 m, logo mais raso que o poço em estudo.

Esse poço fornecerá ao simulador as informações relativas ao cenário de perfuração de cada broca, que denominaremos de *bit record*, e as informações acerca dos revestimentos assentados, como profundidade de assentamento e tempo. Tais informações alimentarão o simulador de maneira direta e serão utilizadas para estimar o tempo total de perfuração, como mostrado na seção 4.3.9.

Consideramos que esse poço foi perfurado sem nenhum evento indesejado, com a perfuração correndo perfeitamente dentro do planejado. Este foi projetado para ter um total

de sete revestimentos, sendo todos pendurados na cabeça do poço, ou seja, sem haver *liners*. A tabela a seguir apresenta a profundidade em que os revestimentos foram assentados, bem como o intervalo de tempo, em horas, que todo o procedimento levou para ser finalizado.

Tipo de Revestimento	Profundidade de assentamento	Tempo até o reinício da perfuração
Condutor	1060 m	200 ⁸ h
Superfície	1500 m	20 h
Intermediário	2800 m	35 h
Intermediário	3650 m	45 h
Intermediário	4300 m	50 h
Intermediário	4750 m	53 h
Produção	5200 m	60 h

Tabela 13 – Dados referentes ao assentamento dos revestimentos no poço de correlação

No que diz respeito aos dados de broca o poço de correlação foi perfurado utilizando-se um total de 15 brocas, todas tricônicas, com jatos de 12/32, 12/32 e 16/32 polegadas. Para os trechos mais superficiais (profundidade inferiores a 3400 m) foram aplicadas brocas classe 1-3 segundo classificação da IADC pré-1987⁹. Já para trechos mais profundos (profundidade superiores a 3400 m) foi utilizada broca de classe 3-1. Para definição dessas classes de broca estamos considerando que a formação acima da profundidade limite de 3400 m se comporta como macia ou média dureza, enquanto que as rochas mais profundas se comportariam como sendo duras e semi-abrasivas (Bourgoyne et al. 1986). Vale observar que a profundidade limite foi escolhida de maneira puramente arbitrária.

As brocas de classe 1-3 apresentam, como parâmetros de desgaste H_1 , H_2 e peso por diâmetro máximo iguais a, respectivamente, 1,84, 6 e 8000 *lbf/in*, enquanto as brocas de classe 3-1 tem os parâmetros acima iguais a 1,7, 3 e 10000 *lbf/in*, respectivamente.

⁸ Contabiliza o tempo total desde a chegada da sonda no porto até o início da perfuração da segunda fase

⁹ Em 1987 a IADC passou a utilizar um sistema de classificação de broca utilizando 4 caracteres para brocas com partes móveis (Halliburton, 2008)

Independentemente do estado de desgaste as brocas foram trocadas em cada descida de revestimento. O peso sobre a broca e a rotação são considerados constantes ao longo do intervalo de rocha perfurado, variando entre 4000 e 6000 *lbf/in* e entre 70 e 80 *rpm*, respectivamente.

6.1.4. Dados Discretizados por Litologia

Conforme explicado na seção 4.3.9 a constantes de abrasividade, bem como os parâmetros do modelo de Bourgoyne-Young e a tensão mínima para promover o fratura da rocha são propriedades da litologia, se diferenciando conforme se diferencia a formação perfurada. Com exceção da tensão mínima de fraturamento, os demais parâmetros não são usualmente obtidos de maneira direta, demandando análises estatísticas para sua determinação.

Como sabido a coluna estratigráfica do poço em questão é composta apenas por intercalações de arenitos e folhelhos. Nesse sentido nosso modelos considerará, como forma de simplificação, que os parâmetros em questão não serão alterados por fatores como compactação por exemplo, sendo constantes todas as lentes daquela determinada formação.

Sendo assim as litologias forma estudadas, a partir de dados do poço de correlação, e observou-se que as areias, muito consolidadas, forneciam maior resistência a perfuração, apresentando uma constante de abrasividade menor (o que significa uma abrasividade absoluta maior), bem como maior tensão mínima de fraturamento, em relação aos folhelhos. Além disso o parâmetro a_1 do modelo de perfuração para as areias será menor que para os folhelhos, pois foi observado que, ao atingir tais formações a rocha tinha sua taxa de penetração reduzida bruscamente. Já os demais parâmetros serão considerados constantes, a nível de simplificação.

A Tabela 14 apresenta os parâmetros que serão utilizados para a simulação da perfuração do poço estudado.

Litologia	τ_H	WOB_t	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
-----------	----------	---------	-------	-------	-------	-------	-------	-------	-------	-------

Folhelho	100	0	1,5	7e-5	5e-6	3e-5	1	0,5	0,5	0,5
Arenito	90	1,5	1,45	7e-5	5e-6	3e-5	1	0,5	0,5	0,5

Tabela 14 – Dados relativos a cada litologia da coluna estratigráfica

6.1.5. Custos Associados a Perfuração

Todos os custos associados a perfuração do poço PEGASUS foram arbitrados, sendo considerados apenas os custos levantados na seção 4.3.10.

Embora tais valores tenham sido arbitrados foram assumidos valores tais que o custo total da perfuração não fugisse da realidade, considerando um valor médio diário da perfuração de um poço da ordem de US\$ 1.000.000, sendo a perfuração realizada por uma sonda de aluguel diário médio de US\$ 300.000. A tabela a seguir apresenta os custos de cada uma das rubricas de custo levantadas na tabela 6.

Rubrica dos Custos	Valores (US\$)	Rubrica dos Custos	Valores (US\$/dia)
Fixos		Variáveis	
Controle de qualidade	65.400	Helicóptero	10.500
Posicionamento da sonda	90.000	Base de apoio	60.000
Equipamentos da wellhead	75.000	Barcos de apoio	78.000
Perfis elétricos	3.000.000	Gerência de resíduos	6.000
Teste de formação	1.650.000	Previsão Meteorológica	450
Serviços de HSE	390.000	Descida de revestimento	10.200
Seguro	216.000	Cimentação	10.800
Serviço Legal	165.000	ROV	12.000
Cabeça de poço	1.050.000	Equipamentos de perfuração	6.300
Revestimento	1.650.000	Equipamentos de pescaria	1.650
Cimento	300/ <i>bbl</i>	Mud logging	12.000
Materiais para fluido	1.800.00	MWD, LWD, etc.	45.000
Broca classe 1-3	30.000/broca	Telecomunicações	1.500

Broca classe 3-1	50.000/broca	Pessoal	105.000
		Diesel	96.000
		Aluguel da sonda	500.000

Tabela 15 – Dados de custos utilizados no cenário de perfuração

6.2. Projeto e Planejamento do Poço PEGASUS

O projeto do poço PEGASUS será desenvolvido através do simulador de poço UFRJ DRILLING SIMULATOR, em particular no seu primeiro módulo, utilizando a coletânea de dados e informações listadas na seção 6.1. Toda a metodologia apresentada no capítulo 4 será seguida, com o autor fazendo às vezes do usuário do programa.

6.2.1. Tratamento Inicial dos Dados

Os dados do perfil sônico, bem como os dados principais para o planejamento de poço foram carregados de maneira direta pelo programa, como podemos observar na figura abaixo.

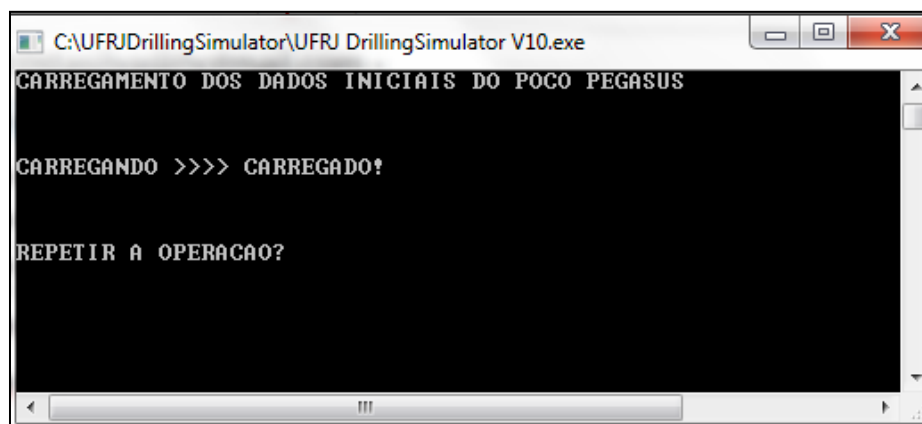


Figura 89 – Tela do simulador indicando que os dados foram carregados

Como o intervalo de profundidade em que os dados sísmicos estão discretizados são da ordem de 50 m utilizaremos um intervalo interpolado de 10 m para a re-formatação do perfil. Vale notar que 10 m satisfaz os requisitos do simulador, que exige um intervalo de interpolação inteiro e que seja divisor do intervalo original.

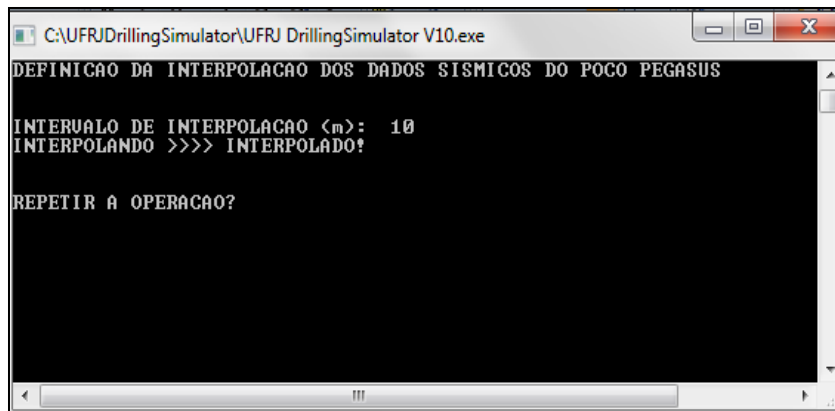


Figura 90 – Tela do simulador indicando que os perfil foi interpolado

Após a implementação da interpolação o perfil sônico será dado pela figura abaixo. Podemos observar que os resultados da interpolação ajustam perfeitamente com os dados originais do perfil, o que valida o bloco de interpolação desenvolvido. Vale lembrar que a interpolação que foi implementada no simulador foi linear, logo sua acurácia é diretamente *proporcional* a intervalo de dados original.

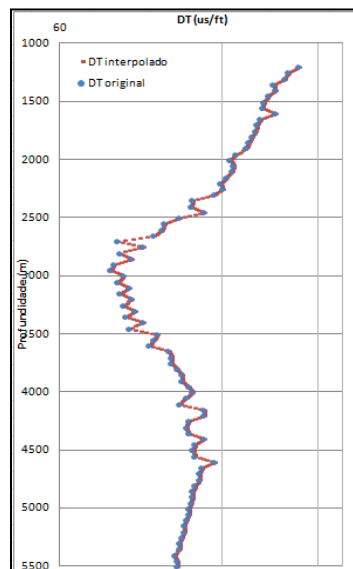


Figura 91 – Diagrama do perfil sísmico original e após a interpolação (elaboração própria, baseado nos resultados do simulador proposto)

6.2.2. Resultados quanto o cálculo da densidade do poço

Concluído o carregamento dos dados iniciais e interpolação do perfil sônico a próxima etapa do simulador é o carregamento ou estimativa do perfil de densidades.

Por ausência de um perfil de densidade completo ao longo da trajetória do poço o método escolhido pelo usuário para estimar a densidade será a aplicação do modelo de Gardner (Eq. 1). Entretanto, como um pequeno trecho do poço apresenta dados de densidade medidos utilizaremos esse perfil para calibrar o modelo utilizado.

Para calibrar o modelo de Gardner iniciou-se os testes com os parâmetros a e b valendo, respectivamente, 0,23 e 0,25, que são os valores típicos para o Golfo do México (Rocha et al. 2006). A figura a seguir foi gerada pelo próprio simulador.

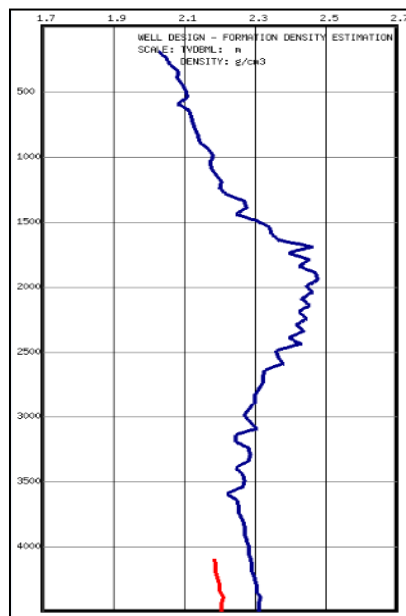


Figura 92 – Perfil de densidade estimada para $a = 0,23$ e $b = 0,25$ (resultado do simulador proposto)

Como esses assumidos para os parâmetros a e b sobrestimaram a densidade ao longo do poço é necessária a correção desses parâmetros visando ajuste. Os valores de 0,23 e 0,245 foram então testados e, como podemos ver na figura abaixo o ajuste foi satisfatório.

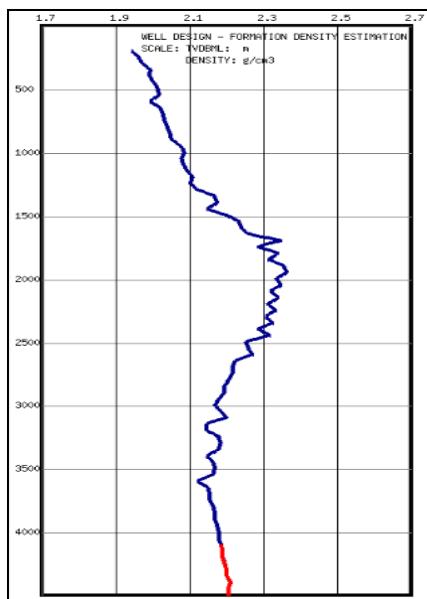


Figura 93 – Densidade estimada para $a = 0,23$ e $b = 0,245$ (resultado do simulador proposto)

6.2.3. Extrapolação do Perfil Sônico e da Massa Específica

Segundo as informações básicas do poço estudado existe um trecho superficial de 200 m onde o elevado nível de ruídos acarretou no descarte dos dados sísmicos daquele trecho e, conseqüentemente, não há informações de densidade calculada para essa zona.

Assim sendo o usuário optará por extrapolar a densidade nesse trecho a partir da equação de Traugott (Eq. 2), onde os parâmetros a , b e c serão obtidos pelo próprio simulador utilizando o método da minimização do erro, apresentado na seção 4.3.2.2. Como forma de garantir um melhor ajuste e continuidade com os trechos superficiais foi utilizado um limite para comparação igual a 100 m abaixo do trecho sem registro. A figura a seguir apresenta o ajuste da densidade extrapolada com a densidade calculada.

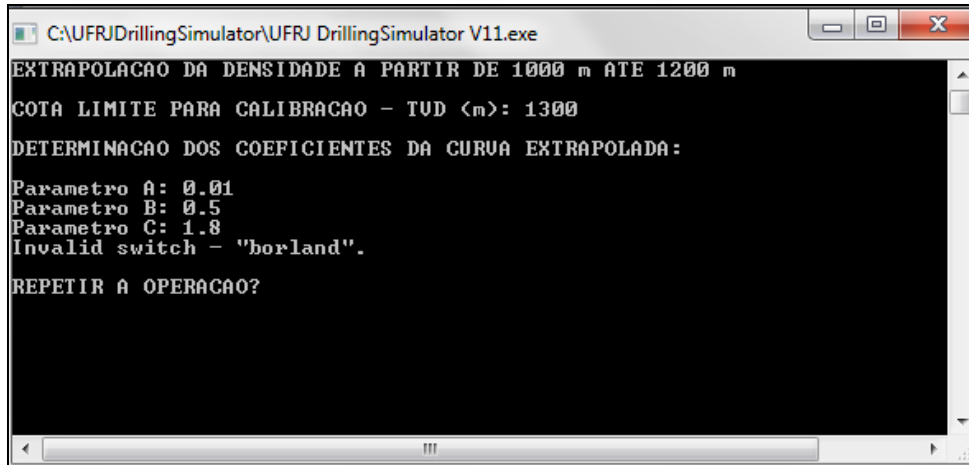


Figura 94 – Tela do simulador informando os parâmetros ajustados da curva de extrapolação de densidades

A figura abaixo apresenta o perfil da curva de Traugott utilizando os parâmetros a , b e c estimados pelo simulador e fornecidos acima. Vale observar o bom ajuste entre a densidade extrapolada e a calculada para as profundidades mais rasas do poço.

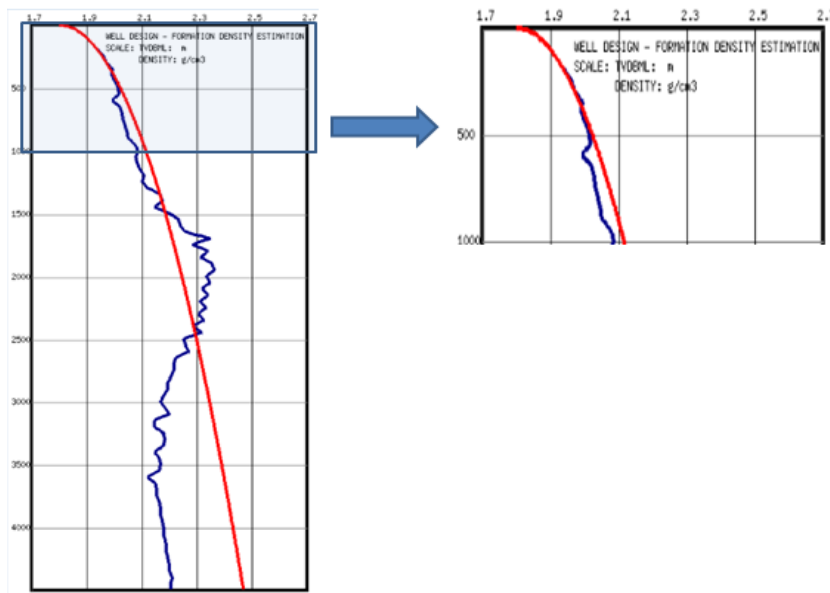


Figure 95 – Perfil da massa específica extrapolada (resultado do simulador *proposto*)

A partir dos dados da curva de densidade extrapolada poderemos obter os valores respectivos para o DT no trecho superficial, aplicando a equação inversa de Gardner (Eq.

101). Os parâmetros da equação de Gardner deverão ser os mesmos utilizados para estimar o perfil de densidades ao longo do poço, ou seja, $a = 0,23$ e $b = 0,245$. Os gráficos abaixo apresentam a relação entre DT e a densidade, através da Eq. inversa de Gardner.

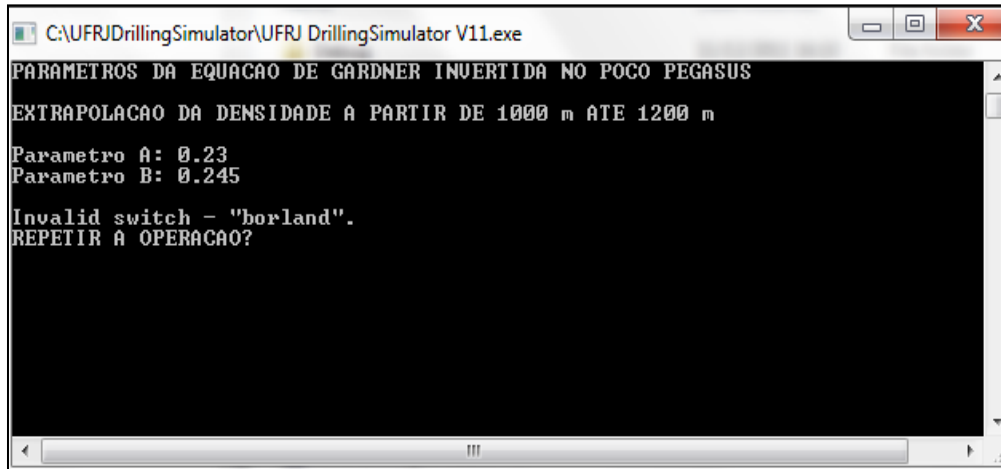


Figura 96 – Tela do simulador para carregamento dos parâmetros da equação invertida de Gardner

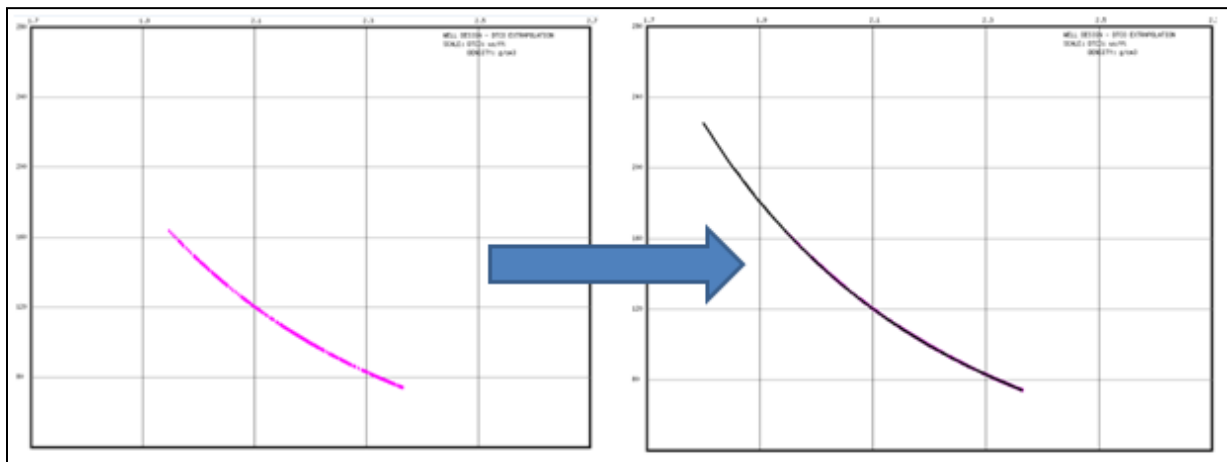


Figura 97 – Extrapolação do perfil sísmico através do modelo de Gardner (resultado do simulador proposto)

Os pontos em magenta da figura a esquerda representam os dados sísmicos originais do poço frente ao perfil de densidades. O comportamento contínuo dos pontos se deve ao fato de a relação entre os dois parâmetros ser definida a partir da equação de Gardner. Já a curva em preto da figura a direita representa o tempo de trânsito (DT) calculado a partir da Eq. 101 para as profundidades mais superficiais.

6.2.4. Estimação das Geopressões

Após a estimação e extrapolação da densidade e do perfil sônico ao longo de toda a trajetória do poço o simulador atuará estimando a tensão de sobrecarga, a pressão de poros e a pressão de propagação de fratura, conforme apresentado abaixo.

6.2.4.1. Estimação da Tensão e Gradiente de Sobrecarga

A tensão de sobrecarga será calculada de maneira direta através da Eq. 4 utilizando o perfil de densidade calculado anteriormente. Não há interferência do usuário nessa estimação. Os resultados da tensão de sobrecarga, bem como do gradiente, calculados e fornecidos pelo simulador se encontram nos gráficos abaixo.

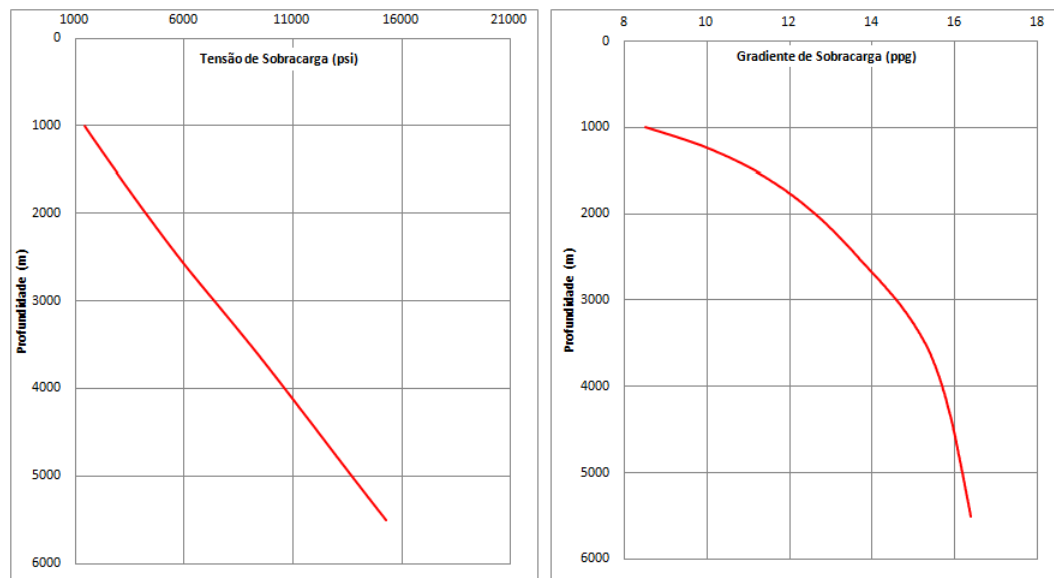


Figura 98 – Perfis de tensão e gradiente de sobrecarga ao longo do poço (elaboração própria baseada nos resultado do simulador proposto)

6.2.4.2. Estimação da Pressão e Gradiente de Poros

Para a estimação da pressão de poros será utilizado o modelo de Eaton, uma vez que estamos assumindo que não há zonas de sobrepressão originada por expansão de fluidos. Dessa forma para utilizar o modelo proposto por Eaton o simulador necessita definir qual a curva de tendência que será utilizada como propósito de definir a zona de compactação

normal ao longo do poço. Como já foi apresentado anteriormente o simulador é capaz de tratar com três maneiras de definição da curva de tendência, entretanto, para esse estudo, será utilizado o critério da calibração com os dados advindos do RFT do poço de correlação.

Foi definido como limites da zona de compactação normal as profundidades de 1200 e 2500 m. Isso significa dizer que o simulador testará apenas valores dentro desse trecho e que sejam folhelhos, já que esta é uma consideração inerente ao modelo de Eaton. A profundidade de 2500 m foi definida como limite máximo para o trecho de compactação normal, pois além dessa profundidade o *DT* reduz rapidamente, enunciando uma zona de compactação anormal.

A figuras abaixo apresenta os pontos que definirão a curva de tendência.

```
*****ZONA DE COMPACTACAO NORMAL*****
COTA SUPERIOR PARA CALIBRACAO - TUD (m): 1200
COTA INFERIOR PARA CALIBRACAO - TUD (m): 2500
*****GRAD. COMPACTACAO NORMAL*****
PONTOS DA TREND CURVE:
TUD 1: 1600 m
DTCO 1: 149 us/ft
TUD 2: 2150 m
DTCO 2: 121 us/ft
Invalid switch - "borland".
REPETIR A OPERACAO?
```

Figura 99 – Tela do simulador contendo pontos que definirão a curva de tendência

Já o gráfico abaixo apresenta o perfil sônico do poço bem como a curva de tendência definida pelos pontos marcados na figura acima. Vale observar que a escala de profundidade é *TVDBML* (*True Vertical Depth Bellow Mud Line*), logo a mesma está referenciada ao leito marinho e não a mesa rotativa.

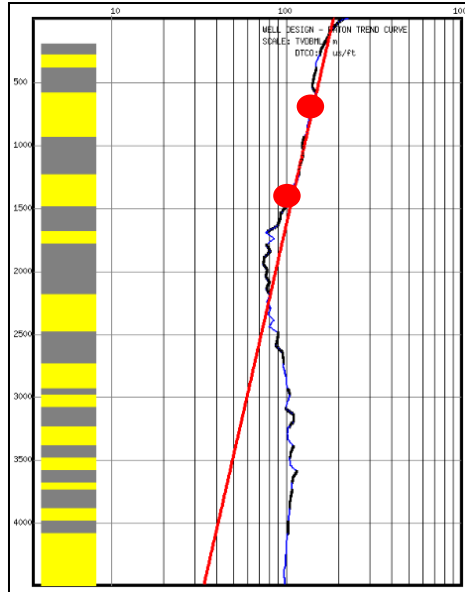


Figura 100 – Perfil sísmico do poço PEGASUS com a curva de tendência definida pelo simulador (resultados do simulador proposto)

Em seguida foi selecionada a opção de suavizar a curva a partir de uma média ponderada das cinco pressões estimadas em profundidade imediatamente acima de cada ponto na trajetória do poço. Os coeficientes de ponderação serão variam entre 5 e 1, como maior peso dado ao gradiente da profundidade de interesse, diminuindo continuamente conforme se afasta do ponto em questão. Assim o perfil de pressão e gradiente de poros, calculados pelo simulador, serão dados abaixo.

```

C:\UFRJDrillingSimulator\UFRJ DrillingSimulator V11.exe
SUAUizacao DO PERFIL DE GRADIENTE DE POROS NO POCO PEGASUS
DETERMINACAO DOS COEFICIENTES DE PONDERACAO
Gp_i:      5
Gp_<i-1>:  4
Gp_<i-2>:  3
Gp_<i-3>:  2
Gp_<i-4>:  1
REPETIR A OPERACAO?
  
```

Figura 101 – Coeficientes de ponderação para suavização da curva de pressão de poros

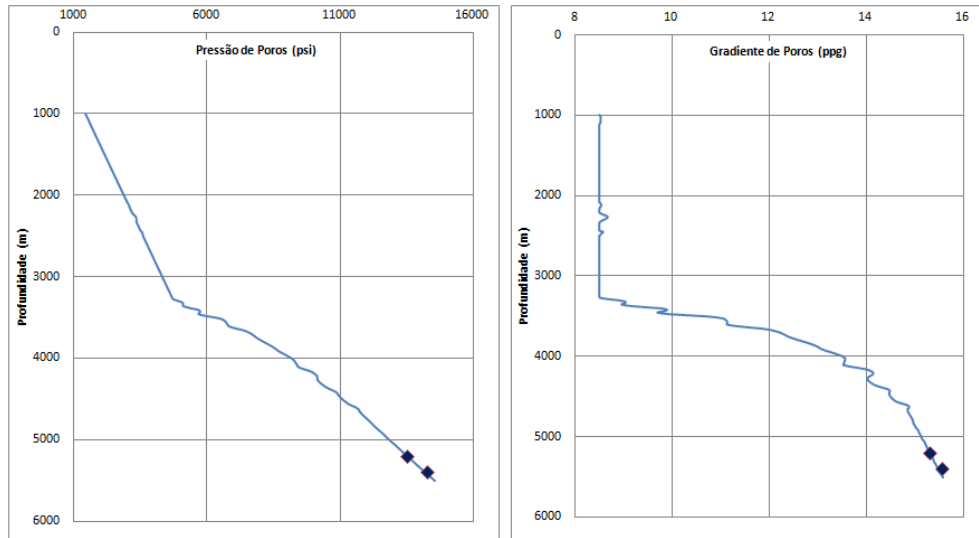


Figura 102 – Perfil de pressão e gradiente de poros calculados pelo simulador (elaboração própria baseada nos resultado do simulador proposto)

6.2.4.3. Estimação da Pressão e Gradiente de Fratura

O cálculo do gradiente de fratura será realizado de maneira direta, sem a atuação do usuário. Para tal o programa utilizará os dados de LOT do poço de correlação, apresentado na seção 6.1, aplicando na metodologia proposta na seção 4.3.3.3. A pressão de fratura, bem como o gradiente de fratura, já calibrados, são apresentados na Figura 103.

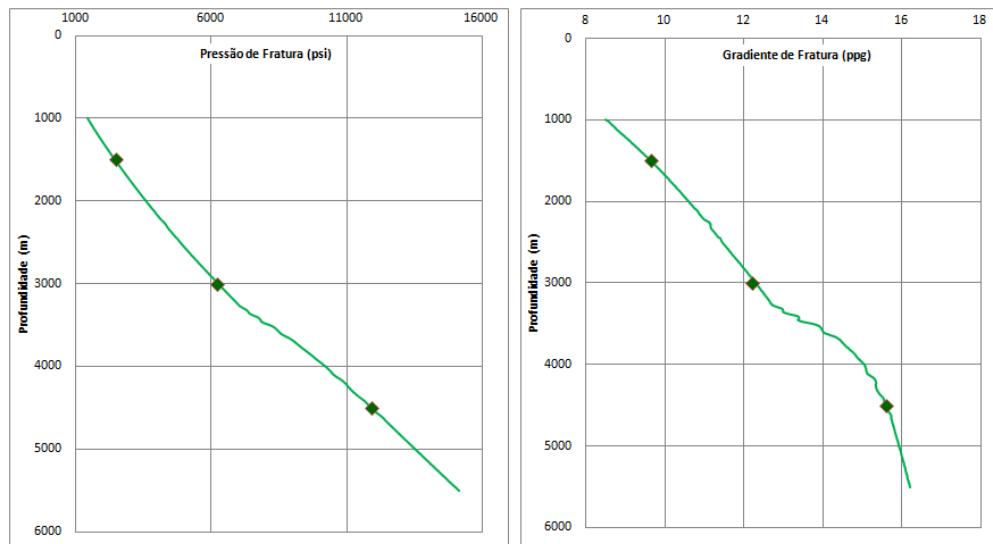


Figura 103 – Perfil de pressão e gradiente de fratura calculado pelo simulador (elaboração própria baseada nos resultado do simulador proposto)

6.2.5. Assentamento dos Revestimentos

Para a definição das profundidades de assentamento dos revestimentos será escolhido o critério baseado na janela operacional. Essa escolha é devido ao fato de o critério da margem de riser não ser funcional para o caso brasileiro, onde eventos que exijam uma desconexão de emergência do riser não são comuns como, por exemplo, no mar do norte. Dessa forma será escolhido a opção 7 na tela inicial do módulo de planejamento de poço.

A margem de erro associada ao gradiente de poros e de fratura será da ordem de 0,1 *ppg* e a profundidade de assentamento dos revestimento condutor e de superfície já foram previamente definidos como sendo igual a 1060 e 1500 m respectivamente.

A Figura 104 apresenta o esquema de assentamento dos revestimentos intermediários e de produção, que comporão o arranjo telescópico do poço juntamente com os revestimentos mais superficiais. A figura em questão é fornecida como produto do simulador.

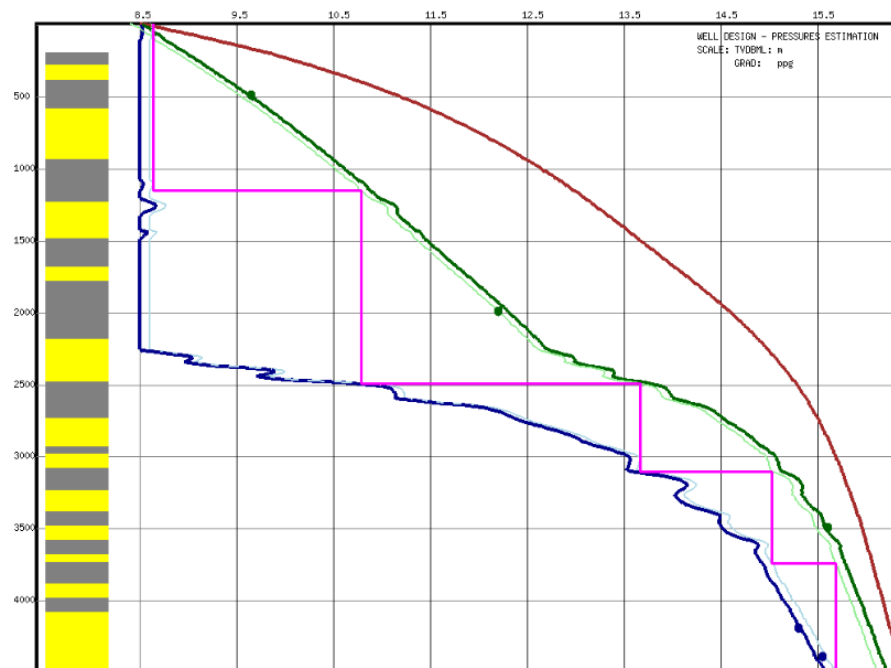


Figura 104 – Diagrama de geopressões e profundidade de assentamento dos revestimentos do poço estudado (resultados do simulador proposto)

Portanto, a partir dos resultados fornecidos pelo simulador podemos observar que o poço terá um total de sete revestimentos assentados, correspondendo a sete fases de perfuração, sendo quatro destes revestimentos intermediários. A tabela abaixo contém os dados relativos

ao assentamento dos revestimentos, como a respectiva profundidade e densidade de fluido considerada no critério para cada fase. Essas informações é resultado do simulador, sendo parte dos vários dados contidos no relatório final de planejamento, no arquivo de saída UFRJDRILLINGSIMULATOR/OUTPUTS/OUTPUTWELLPLANNING_RESULTS.TXT.

	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	Fase 6	Fase 7
Profundidade	1060 m	1500 m	2160 m	3500 m	4110 m	4750 m	5500 m
Densidade	8,5 ppg	8,5 ppg	8,63 ppg	10,60 ppg	13,67 ppg	15,01 ppg	15,68 ppg

Tabela 16 – Profundidade de assentamento e densidade para cada fase do poço estudado (elaboração própria baseada nos resultados do simulador proposto)

6.2.6. Geometria dos Revestimentos

A definição da geometria dos revestimentos consistirá em determinarmos os diâmetros e a espessura dessas tubulações que serão assentadas no poço. A definição dos diâmetros dos revestimentos serão proporcionados de maneira direta pelo simulador, a partir da coletânea de arranjos de diâmetros de fase e revestimento que são carregadas no programa, conforme apresentado na seção 4.3.5. Já a espessura de cada revestimento deverá ser definida no bloco de dimensionamento dos revestimentos, conforme apresentado na seção 4.3.6.

Consideraremos que todos os revestimentos assentados, com exceção do revestimento de produção, serão pendurados na cabeça do poço, se estendendo até a profundidade definida na Tabela 16. O revestimento produtor será um *liner*, que deverá ser pendurado pouco acima da sapata do último revestimento. Como o ambiente de perfuração é rico em gás sulfídrico o aço que será utilizado em todos os revestimentos será de grau L-80, com 80 *ksi* de limite de escoamento. A escolha entre os aços L-80 ou C-90 se deu de forma puramente arbitrária.

A imagem a seguir apresenta o arranjo de diâmetros do poço, bem como o carregamento dos dados relativos aos revestimentos.

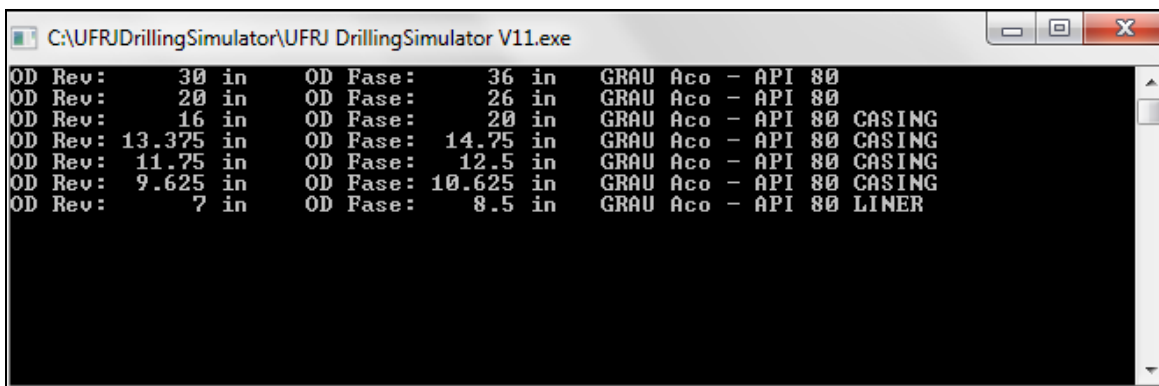


Figura 105 – Dados do arranjo de diâmetros e parâmetros do aço do revestimento

A tabela abaixo apresenta a geometria respectiva a cada fase do poço, contendo informações de diâmetro e espessura do revestimento e diâmetro da fase, bem como o grau do aço utilizado. Tais informações são saídas do simulador e comporão o relatório de planejamento juntamente com outras várias informações relativas a cada fase perfurada.

	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	Fase 6	Fase 7
Diâmetro da Fase	36 in	26 in	20 in	14 3/4 in	12 1/2 in	10 5/8 in	8 1/2 in
Diâmetro do Revestimento	30 in	20 in	16 in	13 3/8 in	11 3/4 in	9 5/8 in	7 in
Espessura do Revestimento	1,5 in	1,0 in	0,46 in	0,67 in	0,74 in	0,72 in	0,29 in

Tabela 17 – Resultados referentes a Geometria das fases de perfuração do poço estudado (elaboração própria baseada nos resultados do simulador proposto)

6.2.7. Desempenho da Perfuração

Após ter concluído o projeto do poço segundo a óptica do assentamento dos revestimentos e a geometria do poço o módulo de planejamento ainda é capaz de desenvolver o projeto de brocas e estimar o tempo de perfuração de cada fase, discretizado em *PT* e *NPT*. Além disso ele também fornecerá um diagrama tempo por profundidade ao longo da perfuração, bem como a taxa de penetração.

6.2.7.1. Projeto de Brocas

Segundo resultados do simulador o poço deverá ser perfurado com um total de 16 brocas, sendo 7 brocas de classe 1-3 e outras 9 brocas de classe 3-1. A tabela a seguir apresenta os dados relativos às brocas que serão utilizada para atingir a profundidade objetivo do poço. Essas informações estão contidas no relatório de planejamento, assim como as informações apresentadas nas seções anteriores.

Broca	Classe	ΔZ (m)	Z total (m)	Δt (h)	t total (h)
1	1-3	440	1500	23,22	254,88
2	1-3	660	2160	46,88	347,26
3	1-3	390	2550	82,24	436,29
4	1-3	260	2810	66,29	511,12
5	1-3	320	3130	62,53	583,15
6	1-3	160	3290	77,92	671,07
7	1-3	170	3460	64,27	745,85
8	3-1	40	3500	5,35	817,21
9	3-1	210	3710	66,34	891,88
10	3-1	350	4060	57,97	962,18
11	3-1	50	4110	4,9	1040,22
12	3-1	350	4460	70,85	1122,81
13	3-1	290	4750	40,53	1244,57
14	3-1	320	5070	56,41	1315,17
15	3-1	290	5360	50,4	1381,85
16	3-1	140	5500	19,99	1434,06

Tabela 18 – Brocas utilizadas na perfuração do poço estudado (elaboração própria baseada nos resultados do simulador proposto)

A partir da tabela acima podemos observar que nas profundidade de assentamento dos revestimentos (Tabela 16) ocorrerá a mudança de broca, como foi modelado na simulação. Podemos atribuir a esse fato a existência de duas brocas (8 e 11) cuja profundidade perfurada destoa das demais brocas utilizadas, perfurando apenas 40 e 50 m de rocha, respectivamente. Da mesma forma podemos observar que algumas brocas apresentaram um desempenho superior a outras, desenvolvendo taxas de perfuração superiores, como é o caso das brocas

utilizadas nas formações mais superficiais. Esse fato será evidenciado na próxima seção, onde apresentaremos o perfil de *ROP* ao longo da perfuração do poço estudado.

6.2.7.2. Estimação do Tempo de Perfuração

De acordo com os resultados da simulação para o planejamento do poço este levaria um total de 1438,54 horas (equivalente a aproximadamente 59 dias e 23 horas) para ser perfurado, já considerando o tempo para a última manobra da coluna. Aos olhos do simulador esse tempo total é discretizado em quatro itens: (i) perfurando, (ii) manobra, (assentando revestimentos) e (iv) deslocamento inicial da sonda. Nesse sentido esse tempo de aproximadamente 60 dias de perfuração será discretizado em:

Atividade	Tempo (h)	Fração (%)
Perfurando	796,10	55,34
Manobra	180,43	12,54
Assentando Revestimento	262,05	18,21
Deslocamento da Sonda	200 h	13,90
TEMPO TOTAL	1.438,54	100,00

Tabela 19 – Tempo total da perfuração do poço estudado (elaboração própria baseada nos resultados do simulador proposto)

Da mesma forma o simulador também discretizará o tempo de perfuração nas fases do poço, ou seja, para cada uma das 7 fases teremos discretizados os tempos *PT* e *NPT*, como podemos observar na tabela abaixo. Vale observar que essa tabela não contabiliza como *NPT* os tempos de assentamento de revestimento, que são considerados interfases.

	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	Fase 6	Fase 7
PT (h)	-----	23,22	46,88	358,61	129,21	111,27	126,80
NPT (h)	-----	4,53	6,56	56,91	36,08	27,97	48,38
Tempo Total (h)	-----	27,75	53,44	415,52	165,29	139,24	175,18

Tabela 20 – Tempo de perfuração de cada fase do poço estudado (elaboração própria baseada nos resultados do simulador proposto)

O relação entre tempo gasto e profundidade perfurada pode ser observada no diagrama tempo por profundidade, fornecido na Figura 106. Nessa imagem ficam evidenciados os trechos de avanço da broca, que correspondem ao pontos de derivada diferente de zero, e os trechos de parada de circulação, notados pelos patamares horizontais no gráfico. Tais patamares podem ser originados por manobras para troca de broca ou por assentamento das sapatas, sendo que, em condições normais de operação, os patamares oriundos das manobras tendem a ser menores do que os oriundos da descida dos revestimentos. A curva em azul escuro fornece a posição do poço ao logo do tempo (em dias).

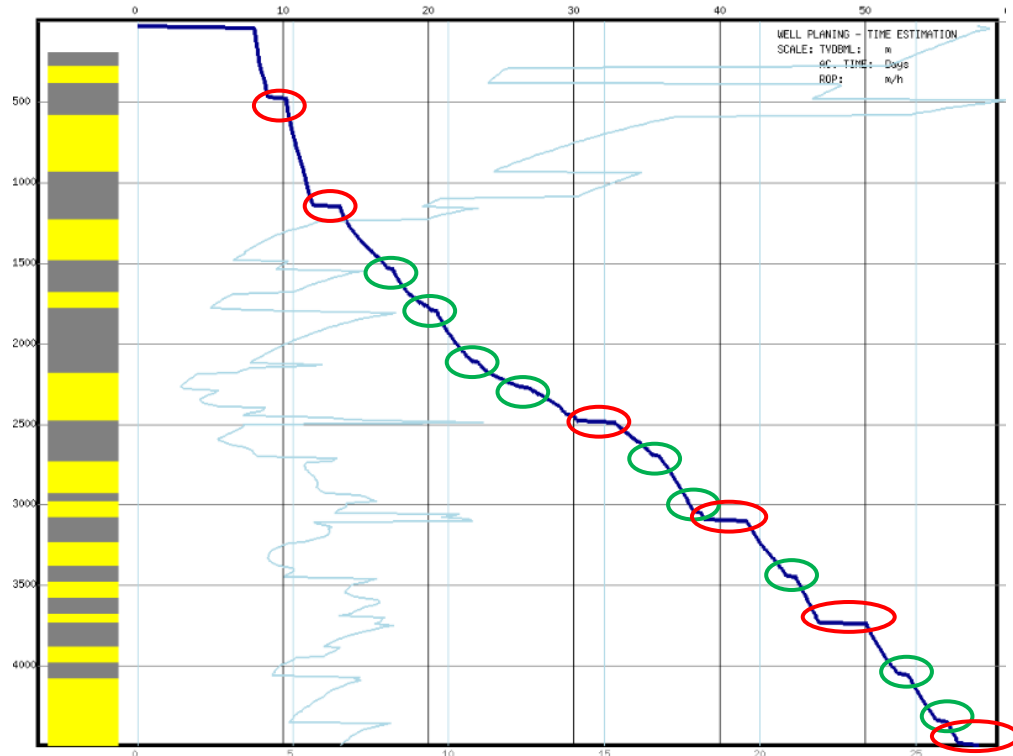


Figura 106 – Diagrama tempo por profundidade estimado para o poço estudado (resultado do simulador)

Na figura acima os patamares marcados em vermelho representam o tempo gasto no assentamento dos sete revestimentos do poço, conforme a tabela 16. Já os patamares marcados em verde representam os pontos de manobra, conforme a tabela 18. Podemos observar que existe um ajuste correto entre as informações fornecidas pelo relatório de perfuração e o diagrama tempo por profundidade acima, como a posição das manobras e das descidas de revestimentos. Da mesma forma podemos observar que a derivada da curva entre duas manobras não é linear, visto que esta depende do comportamento da taxa de penetração.

E esse comportamento é dado pela curva azul claro no diagrama da Figura 106. Esta deverá ser lida no eixo horizontal inferior do gráfico, que apresenta uma variação de 0 até 30 m/h . É notório que o comportamento da taxa de penetração é bastante irregular ao longo do poço, como vários pontos inflexão e de descontinuidade. Isso é explicado pela diversidade de parâmetros que regem o *ROP*, segundo o modelo de Bourgoyne-Young. Qualquer alteração em um desses parâmetros provoca uma alteração do comportamento contínuo da taxa de penetração.

Além disso podemos observar na figura abaixo a influência direta da litologia sobre a taxa de penetração. Ao atingir formações areníticas a taxa de penetração cai violentamente, caracterizando uma descontinuidade no gráfico. Em contrapartida, ao atingir novamente os folhelhos o *ROP* aumenta, caracterizando uma nova descontinuidade. Tal fato era esperado, já o modelo que alimentou o simulador previa um desgaste maior e um *ROP* menor nas areais.

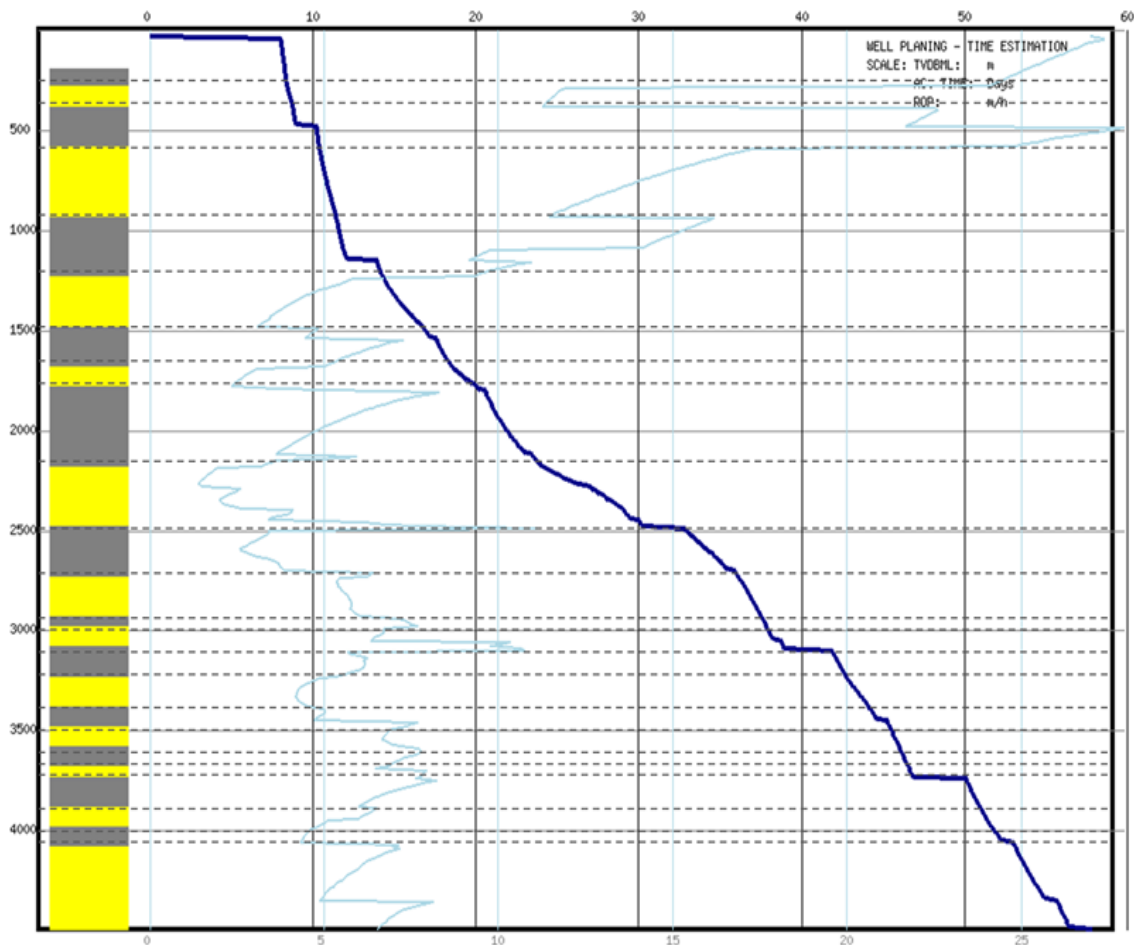


Figura 107 –ROP ao longo do poço, marcando a influência da litologia (resultados do simulador proposto, adaptado)

Nessa figura as linhas pontilhadas correspondem aos limites das camadas de uma determinada formação. Assim fica notório o comportamento do ROP ao mudar de litologia. Outro fator que podemos analisar a influência na taxa de penetração é a mudança de broca. Nos pontos em que alguma manobra ocorre o desgaste daquela broca, que anteriormente era de 100%, será reduzido a zero, já que a broca é nova, o que provocará um aumento natural na taxa de penetração. A figura abaixo elucida essa influência.

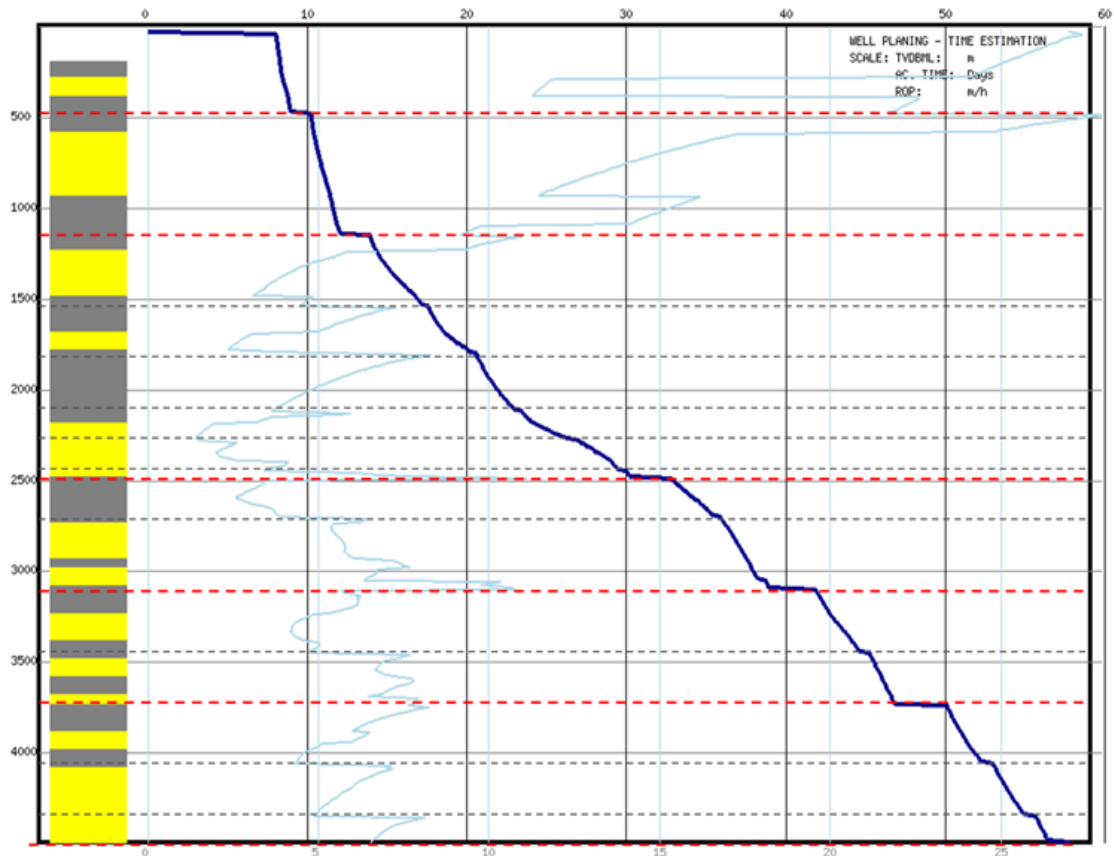


Figura 108 –ROP ao longo do poço, marcando a influência da troca de brocas (resultados do simulador proposto, adaptado)

Nesse gráfico as linhas pontilhadas em cinza representam as profundidades onde alguma manobra ocorreu, sem caracterizar o assentamento de revestimentos. Já as linhas pontilhadas demarcadas em vermelho representam as profundidades onde houve descida de revestimentos. Em ambos os casos somos capazes de observar um aumento repentino da taxa de penetração, o que é coerente, dado que estamos mudando a broca. Esse descontinuidade no gráfico pode ser maior ou menor, dependendo dos demais fatores que atuam sobre o ROP. Podemos que cada patamar horizontal apresenta uma linha, totalizando 16 linhas pontilhadas sobre o gráfico, que, por sua vez, representam as 16 brocas utilizadas no poço. Da mesma forma estão apresentadas 6 linhas vermelhas, o que representam os 6 revestimentos do poço, já que o condutor não está representado, uma vez que este será jateado.

É interessante estudar também os resultados do ROP ao longo do poço frente outros parâmetros como, por exemplo, o gradiente de poros, que atua aumentando o overbalance de pressão no fundo, levando a perfuração rúptil, que é mais rápida de acontecer que a dúctil (vide seção 3.3.2). Além disso esse aumento evidencia zonas de subcompactação, que são mais fáceis de serem perfuradas.

No nosso poço a zona de sobrepressão inicia-se a profundidade de 3270 m. A partir desse ponto podemos observar que dentro da mesma litologia e para uma mesma broca o ROP passa a ser crescente, ao contrário do que acontecia para profundidades mais rasas. Termoa atingido o topo de uma zona de subcompactação pode explicar tal fenômeno. A figura a seguir elucida tal fato, marcando em vermelho o trecho de ROP crescente.

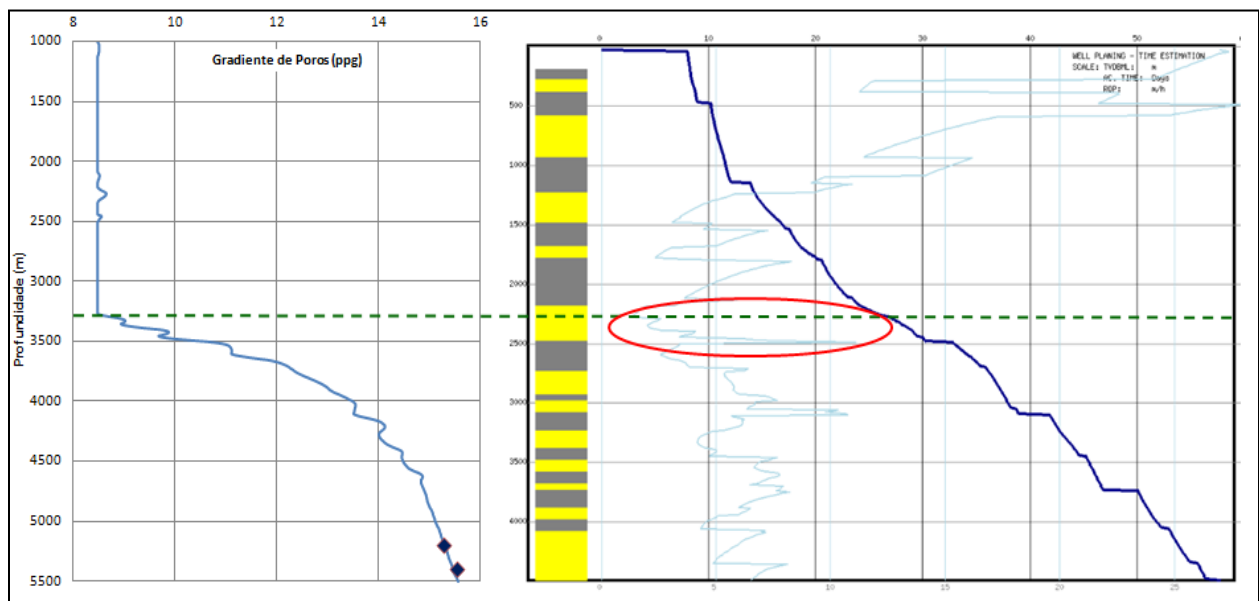


Figura 109 –ROP ao longo do poço, marcando a influência do gradiente de poros (resultados do simulador proposto, adaptado)

6.2.8. Estimação do Custo da Perfuração

Segundo os resultados do simulador o custo total da perfuração do poço em questão foi da ordem de US\$ 58,27 milhões. Como é sabido esse custo é diretamente relacionado ao tempo da perfuração, como ficará elucidado na Figura 110, aonde podemos observar claramente o avanço do custo com o avanço do tempo. Entretanto também podemos observar

que mesmo antes do início da perfuração o custo associado a ela já atinge cerca de US\$ 8,0 milhões. Esse custo inicial se deve as rubricas de custo que foram enunciadas na Tabela 6 como cobradas no início da atividade. Avaliando apenas o custo do aluguel de sonda ao longo dos quase 60 dias de perfuração teríamos um custo de US\$ 30 milhões, que representa mais de 50% do custo total com o poço. Vale notar o perfil do custo ao longo da profundidade segue aproximadamente o perfil de tempo, já que as duas variáveis estão intimamente relacionadas.

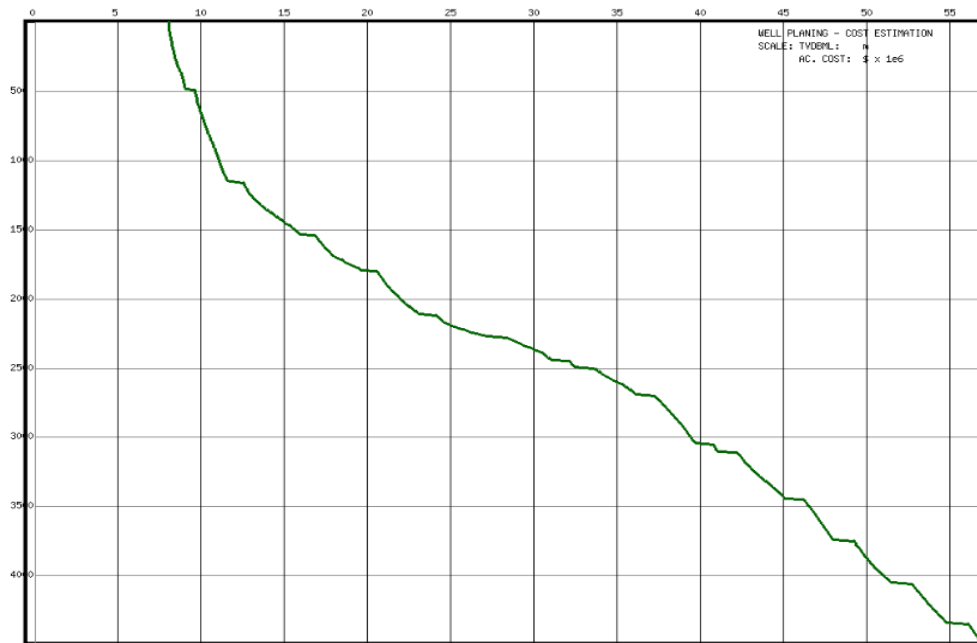


Figura 110 – Diagrama custo por profundidade estimado para o poço estudado (resultados do simulador proposto)

6.2.9. Demais Resultados do Planejamento

Como sabido o módulo de planejamento de poço também é capaz de dimensionar o volume de lama a ser aplicado para cada fase do poço, bem como dimensionar o BOP a ser utilizado. Dessa forma, segundo os resultados da simulação, o volume e a coluna de cimento a serem aplicadas em cada fase do poço serão:

	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	Fase 6	Fase 7
Volume de cimento (bbl)	-----	426,04	367,31	98,64	46,39	51,65	59,31
Altura da coluna (m)	-----	500	800	800	800	800	800

Tabela 21 – Dimensionamento da coluna de cimento para as fases do poço estudado (elaboração própria baseada nos resultados do simulador proposto)

Como a fase 1 (revestimento condutor) é jateada não teremos injeção de cimento na mesma, ao contrário das demais fases, cujo dimensionamento da coluna de cimento seguirá a metodologia *proposta* na seção 4.3.8.

Além disso, segundo os resultados do simulador, o BOP adequado para desenvolvermos a perfuração desse poço será o de pressão de trabalho igual a 15000 *psi*.

As figuras abaixo apresentam dois perfis do poço perfurado, uma vista frontal e o outra vista superior. Ambos os diagramas esquemáticos são produtos do simulador, onde podemos ver o os revestimentos assentados, os liners pendurados na base do último revestimento, entre outros. Interessante notar que o algoritmo de desenho foi desenvolvido de forma que, caso haja o assentamento de até dois liners seguidos ambos serão assentados na base do último revestimento pendurado na cabeça do poço.

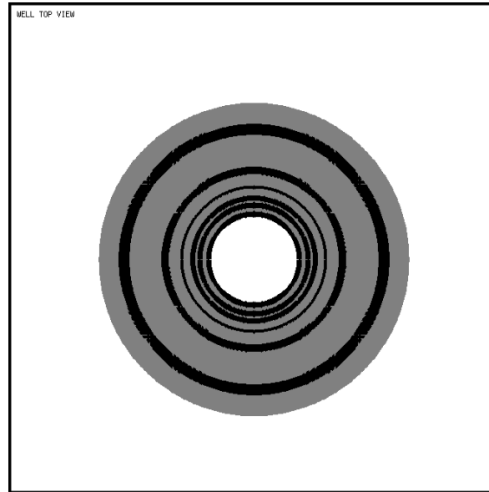


Figura 111 – Vista superior do poço projetado (resultados do simulador proposto)

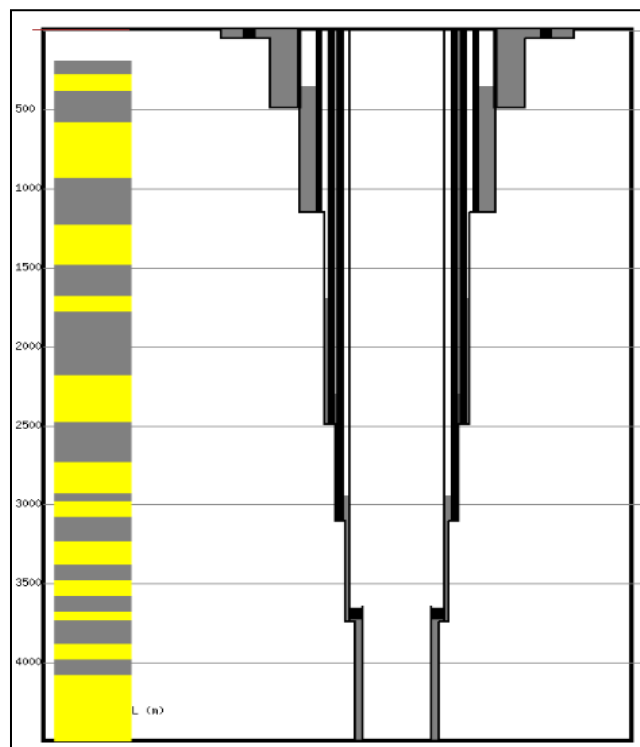


Figura 112 – Vista frontal do poço projetado (resultados do simulador proposto)

6.3. Simulação em Tempo Real da Perfuração do Poço PEGASUS

Finalmente podemos implementar a simulação em tempo real da perfuração do poço estudado, utilizando o simulador desenvolvido. O objetivo dessa simulação será implementar o projeto de poço desenvolvido pelo modulo anterior, porem introduzindo um cenário de perfuração particular.

Como a simulação é dinâmica o usuário tem acesso livre para alterar o cenário o perfuração sempre que achar conveniente e, como isso, poderá avaliar através das diversas saídas do simulador se a sua estratégia está adequada ou não. Dependendo da atuação do usuário pode-se evitar situações de influxos ou de perdas, ou mesmo atingir taxas de penetração que reduzam o tempo total da perfuração. Também pode-se adotar uma estratégia que reduza o desgaste das brocas e, conseqüentemente, o tempo total com manobras, porém a custo de uma redução no *ROP*.

6.3.1. Regras para Cenarizar a Simulação

Uma vez que trabalhamos com um número grande de graus de liberdade nessa etapa, é importante estabelecer algumas regras que nortearão a simulação como um todo.

- O objetivo é simular o poço projetado no módulo de planejamento, respeitando o número de fases e a geometrias destas;
- Só serão utilizadas as brocas classe 1-3 e 3-1, da mesma forma que foi definido para o planejamento. O diâmetro dos jatos serão constantes e iguais a $12/32$ in, $12/32$ in e $16/32$ in, respectivamente.
- A rotação da broca poderá variar livremente. Para esse simulação a variação foi entre 60 e 90 rpm;
- A configuração das seções da coluna foram consideradas constantes ao longo de todo o poço, sendo:
 - Drill Pipes: $OD = 5,0$ in, $\delta = 0,35$ in, $L = 30$ m/seção, $w = 12$ lbf/ft;
 - Heavy Weight Drill Pipes: $OD = 6,0$ in, $\delta = 1,0$ in, $L = 12$ m/seção, $w = 120$ lbf/ft;

- Drill Collars: $OD = 6,5 \text{ in}$, $\delta = 1,25 \text{ in}$, $L = 13 \text{ m/seção}$, $w = 150 \text{ lbf/ft}$;
- O número de seções do *BHA* poderá variar livremente, desde que já haja mudança de *BHA* quando houver manobra por alguma razão. Outro limitante é que, por questões de segurança operacional com a broca, o peso sobre a broca não ultrapasse 70% do peso máximo que a broca utilizada consegue sustentar.
- O fluido utilizado deverá respeitar a densidade definida no projeto de poço anterior. Para perfuração da fase 2, que é utilizada água do mar, esse fluido foi newtoniano, de viscosidade 30 *cp*. Já para os demais casos o fluido utilizado será sempre binghamiano, de viscosidade plástica 19 *cp* e tensão de escoamento de 1500 *lbf/ft²*.
- A vazão também poderá ser alterada livremente. No entanto é importante que ela seja tal que o *ECD* não ultrapasse o gradiente de fratura naquele ponto.

6.3.2. Resultados da Simulação em Tempo Real da Perfuração

Conforme apresentado na seção 5.6 o módulo de perfuração fornece um conjunto grande de dados referentes a perfuração em um determinado tempo e profundidade. A partir desses dados foram gerados diversos gráficos que tem por objetivo elucidar os resultados da perfuração, e verificar que aquilo que foi gerado pelo simulador está condizente com o que foi modelado. Além disso, poderemos ter uma real noção de como a estratégia desenvolvida para simulação desse poço foi mais bem sucedida que o poço planejado para assim revermos o modelo de planejamento desenvolvido no módulo anterior, de forma a acelerar e tornar mais rápida e barata a perfuração como um todo.

Ao fim da simulação um total de 49.968 dados, referentes a 18 parâmetros de perfuração (vide seção 5.6), foram calculados e fornecidos pelo simulador. Essa coletânea de dados abrange tanto a perfuração em si, quanto as etapas de assentamento dos revestimentos e as manobras.

A simulação foi desenvolvida com um intervalo temporal de 30 min sendo acelerada 100 vezes para o acompanhamento dos parâmetros de perfuração na tela do simulador (Figura 87). Os trechos de manobra e revestimento não serão exibidos na tela, mas serão avaliados a cada 20 min. A simulação foi executada em um computador pessoal, com apenas

dois processadores, onde, para o cálculo e exportação de todos os 18 parâmetros em um dado tempo não se levou mais que 100 μs .

6.3.2.1. Assentamento e Geometria dos Revestimentos

Conforme falado anteriormente o objetivo dessa simulação é implementar o poço projetado e que significaria dizer que todas as profundidades, densidades e respectivas geometrias das fases definidas no projeto deveriam ser respeitadas. Entretanto, por falha voluntária ou involuntária do usuário podemos tais parâmetros implementados na simulação não são exatamente os mesmos projetados anteriormente. Tais parâmetros se encontram na tabela abaixo.

	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	Fase 6	Fase 7
Profundidade	1060 <i>m</i>	1552 <i>m</i>	2106 <i>m</i>	3500 <i>m</i>	4287 <i>m</i>	4755 <i>m</i>	5500 <i>m</i>
Densidade	8,5 <i>ppg</i>	8,5 <i>ppg</i>	8,65 <i>ppg</i>	10,70 <i>ppg</i>	13,67 <i>ppg</i>	15,03 <i>ppg</i>	15,50 <i>ppg</i>
Diâmetro da Fase	36 <i>in</i>	26 <i>in</i>	20 <i>in</i>	14 3/4 <i>in</i>	12 1/2 <i>in</i>	10 5/8 <i>in</i>	8 1/2 <i>in</i>
Diâmetro do Revestimento	30 <i>in</i>	20 <i>in</i>	16 <i>in</i>	13 3/8 <i>in</i>	11 3/4 <i>in</i>	9 5/8 <i>in</i>	7 <i>in</i>
Espessura do Revestimento	1,5 <i>in</i>	1,0 <i>in</i>	0,46 <i>in</i>	0,67 <i>in</i>	0,74 <i>in</i>	0,72 <i>in</i>	0,29 <i>in</i>

Tabela 22 – Características gerais das fases do poço simulado (elaboração própria baseada nos resultados do simulador proposto)

6.3.2.2. Desempenho da Perfuração

Segundo os resultados da simulação em tempo real serão utilizadas um total de 21 brocas, sendo 8 brocas tipo 1-3 e 13 brocas tipo 3-1. Isso representaria 1 broca classe 1-3 e 4 brocas 3-1 a mais que o planejado. Isso significaria um acréscimo de US\$ 1,4 milhão frente

ao estimado no planejamento, apenas vislumbrando o excedente de brocas. O gráfico abaixo representa o desgaste das brocas ao longo do tempo de perfuração.

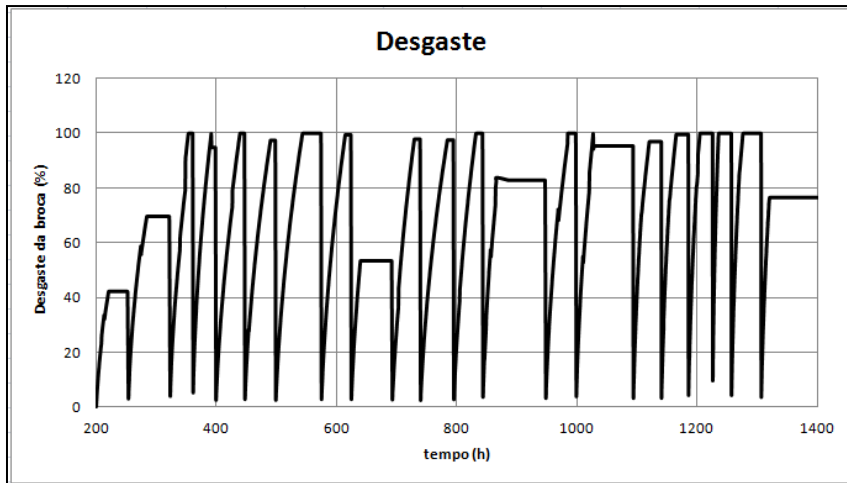


Figura 113 – Perfil de desgaste das 21 brocas utilizadas no poço (elaboração própria baseada nos resultados do simulador proposto)

Entretanto o simulador previu que, com a estratégia desenvolvida, o poço estudado seria perfurado com um tempo total de 1400 h, ou aproximadamente 58 dias e 8 horas. Isso significa dizer que com a abordagem testada haveria uma redução no tempo da perfuração de 38,54 horas, ou 2,7%. Na tabela abaixo podemos observar a discretização do tempo de perfuração desse poço, comparando os resultados obtidos com aqueles fornecidos pelo módulo de planejamento.

Atividade	PLANEJAMENTO DO POÇO		SIMULAÇÃO DO POÇO	
	Tempo (h)	Fração (%)	Tempo (h)	Fração (%)
Perfurando	796,10	55,34	662	47,29
Manobra	180,43	12,54	193,55	13,83
Assentando Revestimento	262,05	18,21	324,45	23,18
Deslocamento da Sonda	200	13,90	200	14,29
Circulação de Kick	-----	-----	20	1,429
TEMPO TOTAL	1.438,54	100,00	1.400,00	100,00

Tabela 23 – Comparação de desempenho dos poços simulado e planejado (elaboração própria baseada nos resultados do simulador proposto)

Analisando comparativamente os resultados da perfuração do poço dado pelo planejamento e pela simulação podemos notar que a contribuição no *PT* para o tempo total da perfuração foi consideravelmente menor no segundo caso. De fato, a partir desses dados podemos estimar que a taxa de perfuração média do poço estimada pelo módulo do planejamento foi 17% menor que a taxa média simulada (5,52 *m/h* contra 6,64 *m/h*).

Entretanto esse aumento considerável na taxa de penetração teve um custo quando analisamos o *NPT* do poço simulado. Este passou de 642,44 h, no poço planejado, para 738 h, no poço simulado, representando um acréscimo de 14,8%. O aumento no tempo total de manobra se justifica pelo aumento no número de brocas utilizadas, e, conseqüentemente, no número de manobras realizadas. O aumento no tempo de assentamento dos revestimentos se justifica pelo fato de, ao todo, os revestimentos nesse poço terem sido assentados 180 m abaixo do poço projetado. Além disso, nesse poço ainda surgiu a rubrica de tempo gasto para circulação de *kick*, visto que o cenário da perfuração foi tal que induzisse um influxo.

A economia de tempo da ordem de 38,54 horas representaria para a campanha de perfuração uma redução da ordem de US\$ 1,65 milhão, com os custos considerados nesse modelo.

A Figura 114 abaixo apresenta o diagrama tempo por profundidade comparando o poço simulado com o projetado, onde podemos ver em vermelho o perfil de profundidade por tempo da simulação em tempo real. Já a curva em laranja representa o *ROP* estimado para o poço simulado. É notório o melhor desempenho da estratégia testada na simulação, com *ROPs* usadamente superiores, embora, para alguns trechos do poço, a taxa estimada no planejamento ainda seja superior. Nesse gráfico também é possível notar a influência dos parâmetros avaliados na seção 6.2 sobre o *ROP*, bem como o avanço do poço, que se afasta ou se aproxima mais da curva planejada dependendo do *ROP*. Entretanto podemos notar mais patamares horizontais, indicando um *NPT* maior, como mostrado na tabela acima.

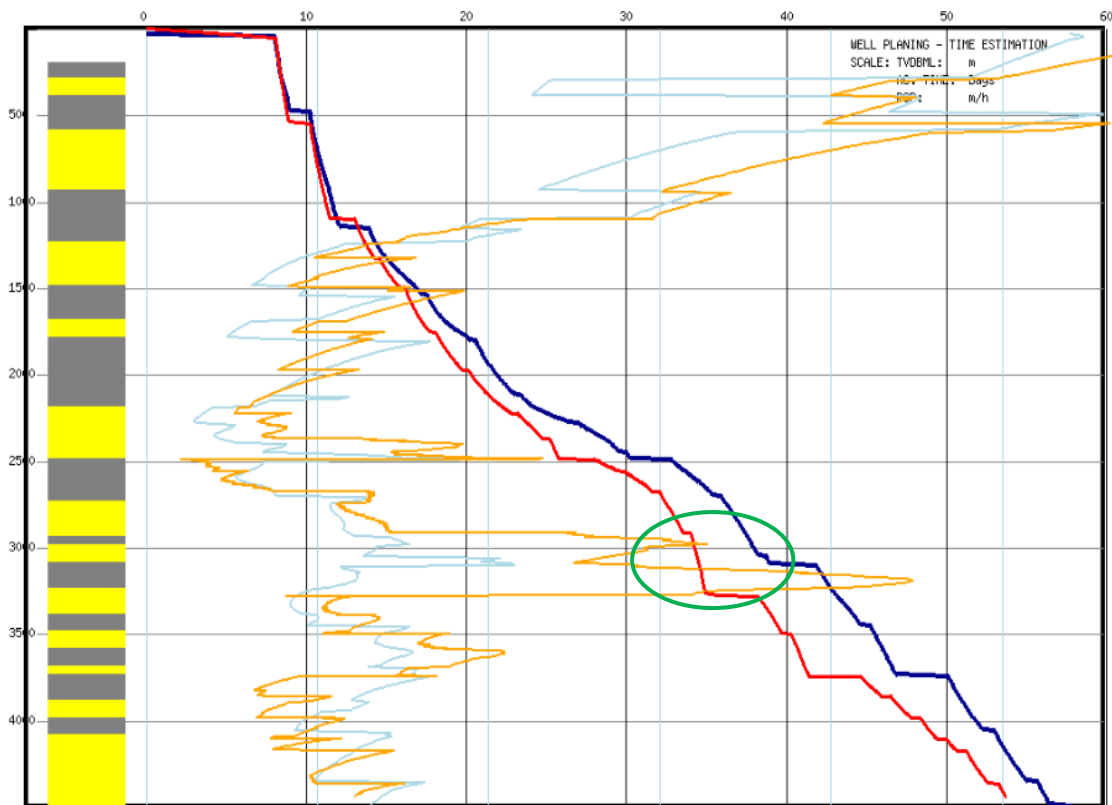


Figura 114 – Diagrama tempo por profundidade do poço simulado frente ao poço projetado (resultados do simulador proposto)

Dentre os diversos pontos passíveis de análise dentro do diagrama acima o trecho marcado em verde é interessante uma vez que nota-se um atraso na descida do revestimento intermediário da fase 5. Esse retardo na descida do revestimento levou o poço a entrar em

estado de perfuração *underbalance*¹⁰, como poderemos ver mais adiante. Dessa forma o diferencial de pressão avaliado no modelo de Bourgoyne-Young atingiu valores positivos, o que levou a perfuração a ser executada de maneira rúptil ainda mais evidente. Nesse sentido o poço sentiu um aumento significativo do *ROP* nesse trecho, atingindo cerca de 22 m/h. Consequentemente notamos que o avanço do poço ficou ainda mais protuberante, afastando-se da curva planejada.

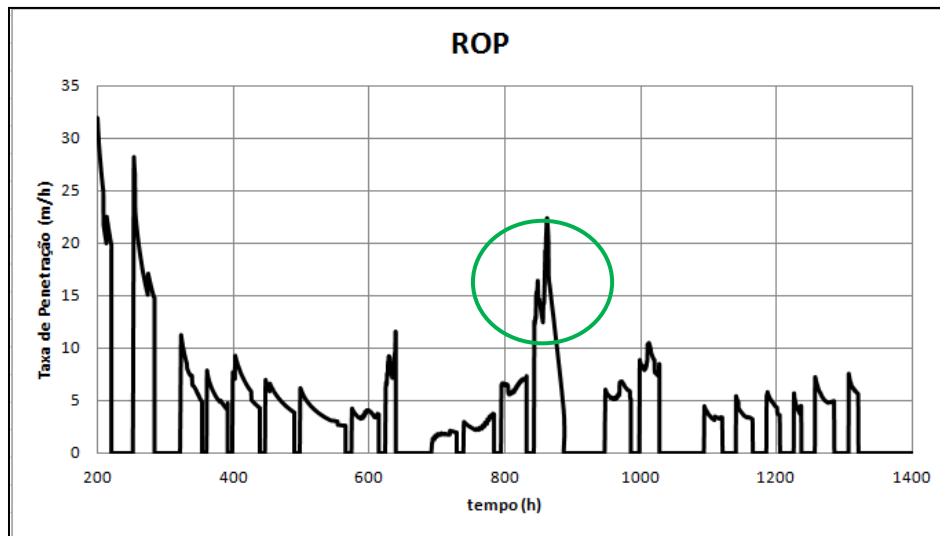


Figura 115 – Perfil do *ROP* ao longo do tempo de perfuração (elaboração própria baseada nos resultados do simulador proposto)

6.3.2.3. Estabilidade do Poço

Como sabido o módulo de simulação em tempo real foi desenvolvido para ser sensível a cenários de balanço de pressão que foge da janela operacional do poço. Desse forma, com a intenção de avaliar se o simulador trataria corretamente possíveis perdas de circulação e influxos foram introduzidos no cenário de perfuração três situações de instabilidade geomecânica: (i) duas onde o *BHP* supera a pressão de fratura e (ii) uma onde o *BHP* é inferior a pressão de poros dentro de um trecho que favoreça o influxo. O gráfico a seguir apresenta o perfil do *ECD* e da densidade do fluido utilizado ao longo da perfuração, onde se evidencia tais cenários críticos.

¹⁰ Estado de perfuração onde a pressão de poros é maior que pressão dentro do poço

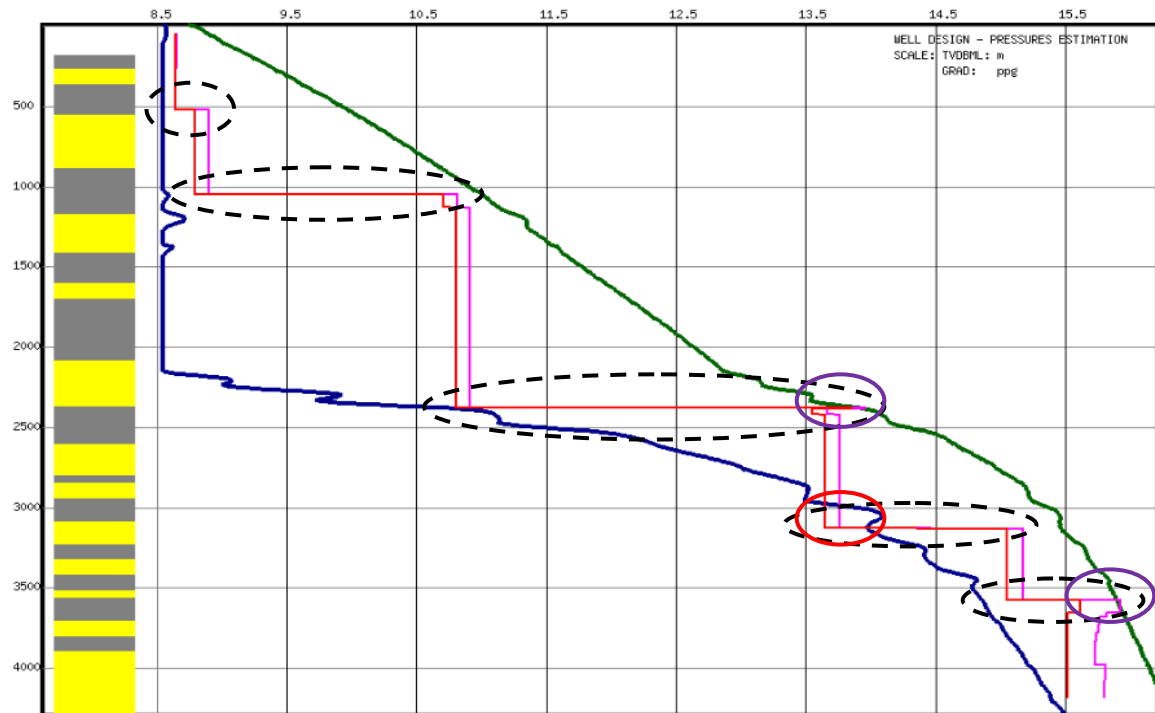


Figura 116 – Diagrama da densidade e da ECD frente a janela operacional do poço (resultado do simulador proposto)

Nesse gráfico, os pontos de descontinuidade marcados em preto pontilhado representam os pontos de mudança de fluido devido a perfuração de uma nova fase. Já o ponto marcado em vermelho representa o cenário de influxo e os marcados em roxo representam os cenários de perda de circulação.

A) Perda de Circulação

Como sabido a ocorrência de perda de circulação acarretaria na queda de volume do poço, bem como seu nível, que deveria ser completado pelo operador. Podemos observar na Figura 116 o volume de fluido perdido ao longo do poço. Observa-se que são dois trechos, assim como elucidado na Figura 118. Quando a densidade é corrigida interrompe-se as perdas.

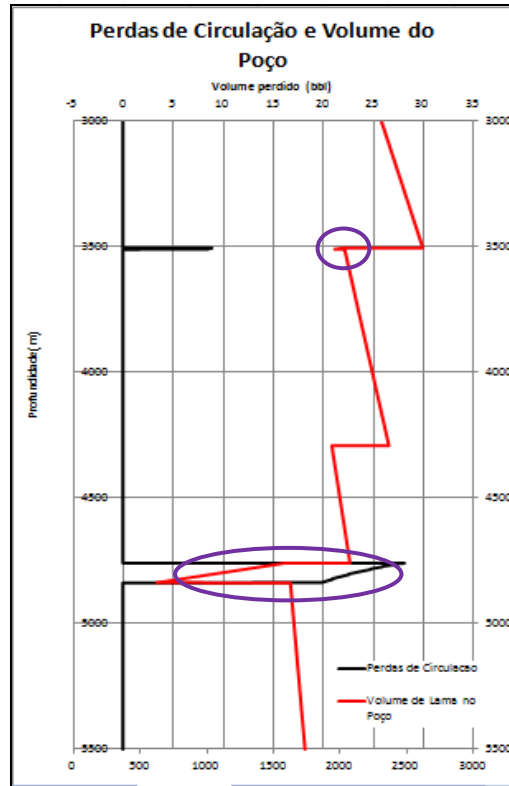


Figura 117 – Eventos de perdas de circulação (elaboração própria baseada nos resultados do simulador proposto)

Nos trechos marcados em roxo é notória a queda do volume dentro do poço, que, por sua vez, acontece devido a volume de fluido perdido para formação, desenhada em preto. A manobra de completar o volume do poço e corrigir o peso de lama foi tomada apenas após 20 horas. O exagero é proposital, visando evidenciar o efeito das perdas e os resultado do simulador frente a esse fato. O gráfico abaixo apresenta o volume de lama no poço e o volume das perdas frente ao tempo da perfuração, apenas para corroboração dos resultados anteriores.

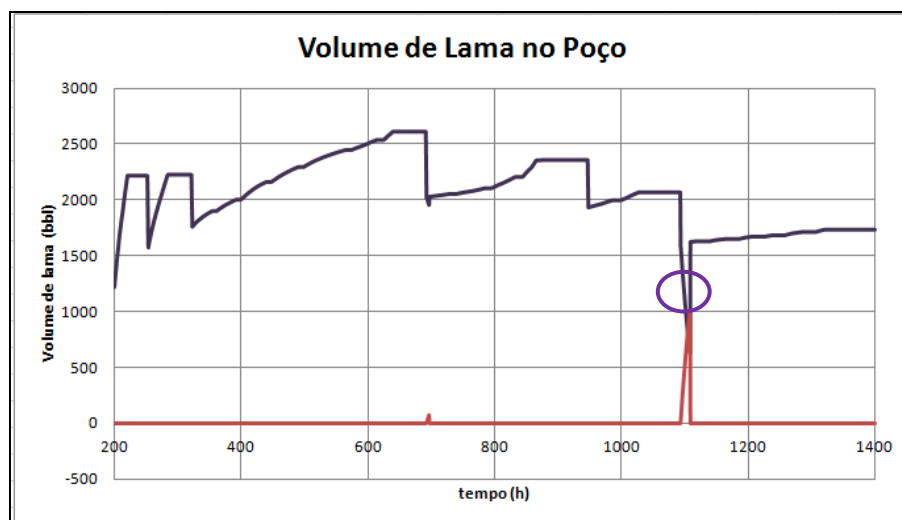


Figura 118 – Eventos de perdas de circulação, ao longo do tempo (elaboração própria baseada nos resultados do simulador proposto)

B) Influxos

É sabido que o kick impacta sobre o volume de lama nos tanques. Entretanto, a nível de simplificação, o respectiva influência do kick sobre os tanques será modelada de maneira instantânea, ou seja, imediatamente quando o influxo ocorrer o tanque sentirá o aumento no volume. Ao mesmo tempo, como já apresentado anteriormente, o gás é considerado incompressível, o que, embora não seja uma simplificação razoável, foi implementada, com objetivo que contornar a modelagem da migração desse gás.

Apenas um ponto ao longo da perfuração do poço apresenta as condições necessárias para haver influxos, que foi marcada na Figura 116. Nesse ponto é esperado um acúmulo crescente de gás no poço, o que acarretaria no aumento do volume de lama nos tanques. No gráfico abaixo podemos observar os resultados do simulador para esse trecho.

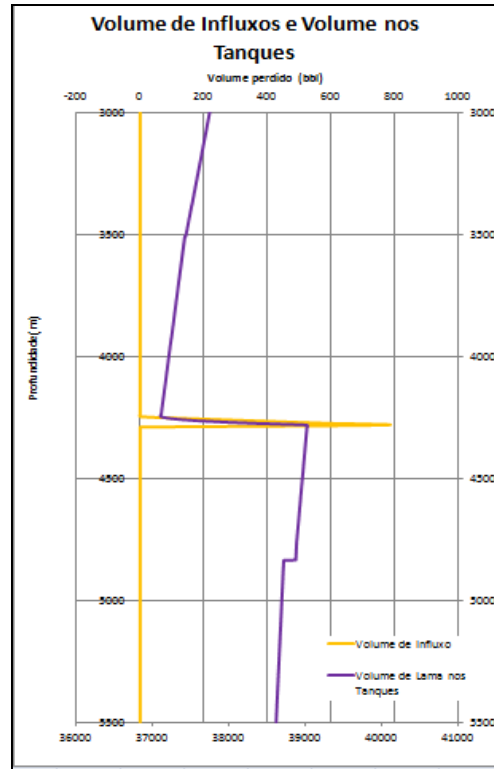


Figura 119 – Resultados do simulador em um evento de perdas de kick (elaboração própria baseada nos resultados do simulador proposto)

Podemos observar no gráfico acima que anteriormente ao influxo o perfil do volume de lama nos tanques era decrescente, como apresenta o modelo da Eq. 154. Entretanto, quando atingiu-se a condição para o kick o influxo entrou no poço regido pela Eq. 152, que fez aumentar rapidamente o volume de lama nos tanques. Quando iniciou-se o processo de circulação, aproximadamente 1,5 h após o início do influxo, o volume de kick reduziu-se a zero e perfil do volume de lama nos tanques retomou a normalidade. O exagero é para evidenciar a questão do influxo e avaliar qual o tratamento dado a este pelo simulador. O gráfico abaixo apresenta o volume de lama nos tanques e o volume de influxo ao longo do tempo.

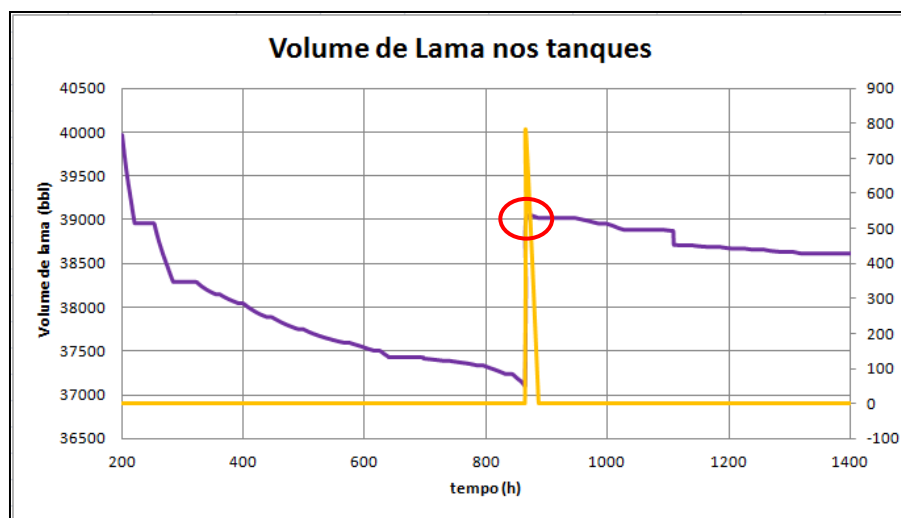


Figura 120 – Resultados do simulador em um evento de kick, em função do tempo (elaboração própria baseada nos resultados do simulador proposto)

6.3.2.4. Hidráulica do Poço

O simulador também avalia ao longo de todo o poço, nas suas diferentes seções, a pressão no ponto. Assim é estimada a pressão no fundo do poço, por exemplo, bem como a pressão de bombeio e queda de pressão na broca, ao longo de toda a perfuração.

A) Perfil Hidráulico

O gráfico abaixo fornece o perfil hidráulico para todo o poço – dentro da coluna e no anular – para a última profundidade medida, ou seja, 5500 m. Ao longo da perfuração o arquivo que fornece tais resultados se atualiza automaticamente, fornecendo ao usuário esse perfil ao longo de toda a perfuração.

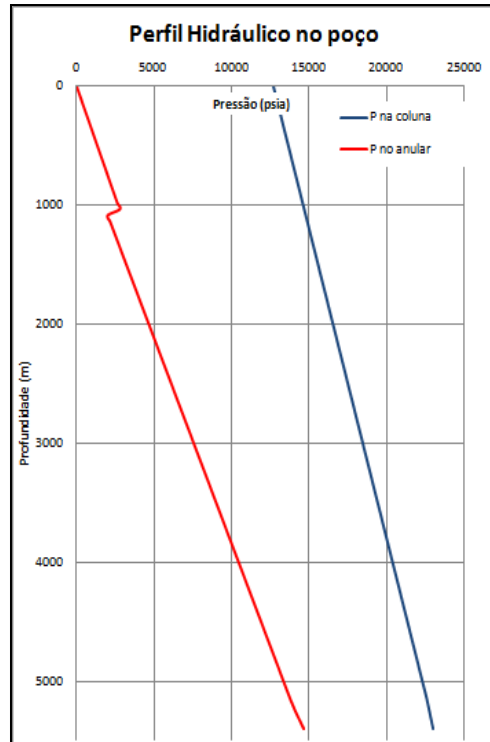


Figura 121 – Perfil hidráulico do poço estudado (elaboração própria baseada nos resultados do simulador proposto)

Nesse gráfico observa-se que a queda de pressão no fundo do poço que distância a pressão no anular da pressão na coluna, marcada em verde, corresponde a queda de pressão na broca, enquanto que a pressão inicial dentro da coluna corresponde a pressão de bombeio. Já a descontinuidade do perfil hidráulico do anular, marcado em roxo, é resultado da mudança de perfil quando atinge-se um diâmetro de parede muito maior que o diâmetro anterior. No caso, o diâmetro da parede em profundidades maiores que 1000 eram da ordem de 9,63 in até 4750 m e 8,5 in até o fundo do poço. Como o diâmetro do riser é 19 polegadas a queda de pressão tende a ser menor, o que acarreta nesse “degrau” na curva.

B) Pressão de Fundo de Poço

Podemos observar também outros gráficos gerados ao longo da perfuração, como, por exemplo, o perfil de *BHP* em função do tempo.

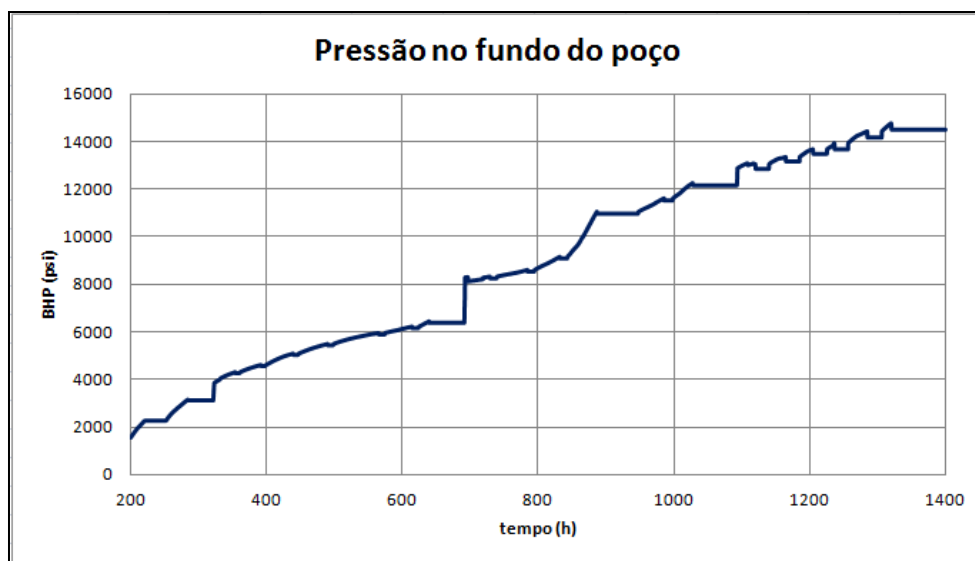


Figura 122 – Perfil de *BHP* ao longo do tempo de perfuração (elaboração própria baseada nos resultados do simulador proposto)

Nesse gráfico observamos que o perfil é continuamente crescente, o que é razoável dado que a poço segue avançado. Todavia existem diversos pontos de descontinuidade no gráfico, onde a pressão é alterada repentinamente. Quando a descontinuidade elucida uma queda no *BHP* significa potencialmente uma parada de circulação, que reduzir a pressão para a pressão hidrostática. Já os sucessivos aumentos se justificam por mudanças na densidade do fluido, na vazão do fluido ou mesmo mudanças na geometria do poço, como a interface de uma fase para outra.

C) Vazão de Bombeio

O gráfico abaixo apresenta os valores da vazão de fluido utilizados durante toda a perfuração.

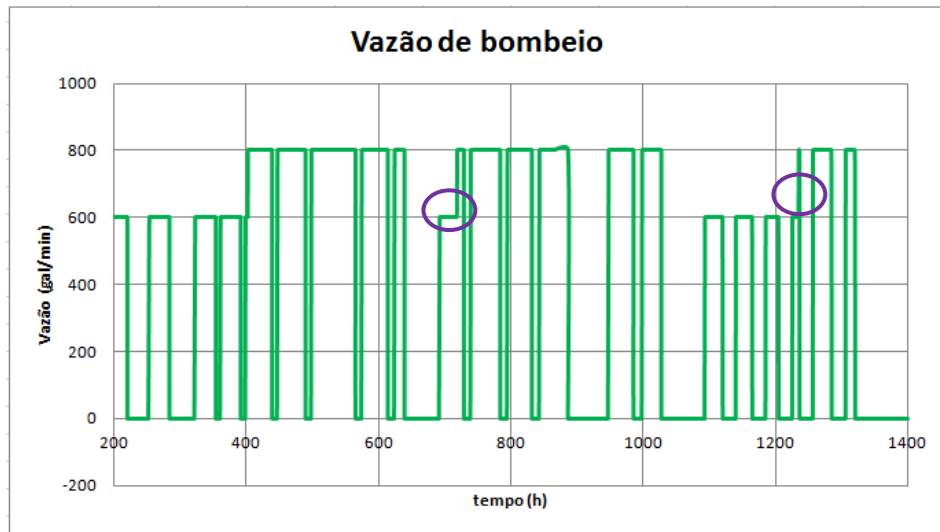


Figura 123 – Perfil de vazão de bombeio definida para o poço (elaboração própria baseada nos resultados do simulador proposto)

Podemos observar que ao longo de toda a perfuração a vazão oscilou entre 600 e 800 gal/min, sendo limitadas pela pressão de fundo e a janela operacional. Como esperado a vazão é reduzida a zero quando ocorrem paradas de circulação.

D) Pressão de Bombeio

Na figura abaixo podemos observar a pressão de bombeio como função do tempo.

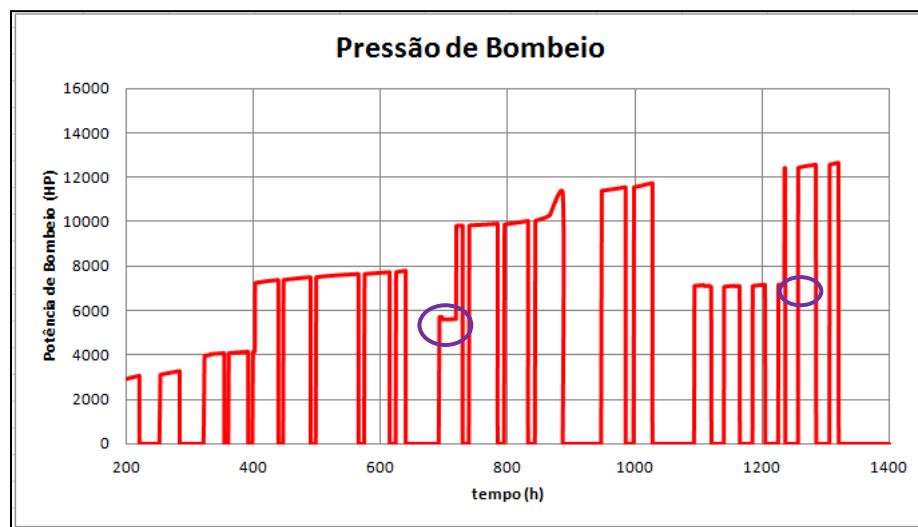


Figura 124 – Perfil de pressão de bombeio ao longo do tempo de perfuração (elaboração própria baseada nos resultados do simulador proposto)

Podemos notar que o perfil da pressão de bombeio é muito pouco sensível ao avanço do poço, já que está é definida apenas pelas componentes de fricção. Portanto as maiores alterações na pressão de bombeio estão intimamente relacionadas a densidade do fluido e a vazão de bombeio. Quando ocorrem paradas de circulação a pressão de bombeio é reduzida a zero, assim como a vazão. Ao mesmo tempo podemos observar a que existe uma correlação direta onde o aumento da vazão acarreta em um acréscimo substancial da pressão de bombeio. Note os pontos marcados em roxo nos dois gráficos.

E) Potência de Bombeio

A figura abaixo representa a potência exigida pela bomba ao longo da perfuração.

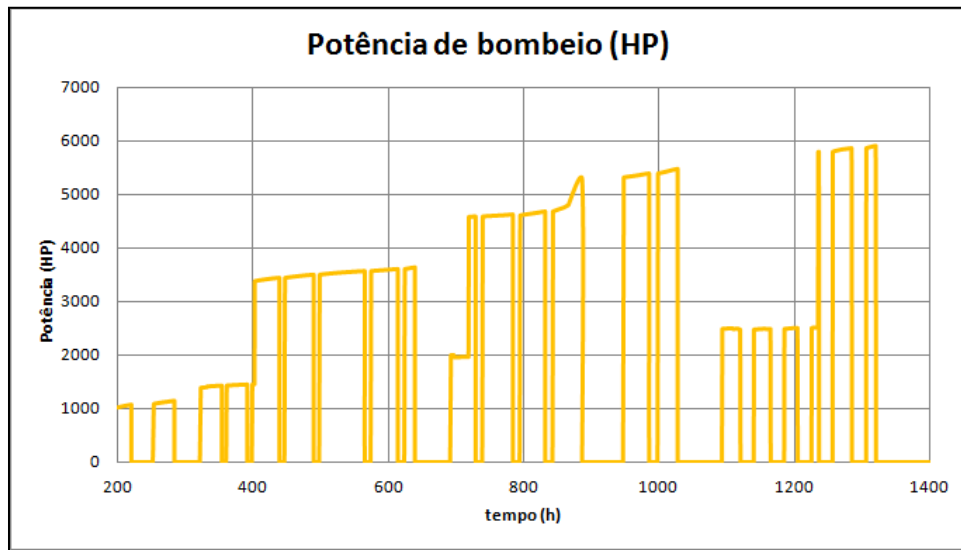


Figura 125 – Perfil de potência da bomba ao longo da perfuração (elaboração própria baseada nos resultados do simulador proposto)

Como a potência da bomba é função da vazão e da pressão de bombeio, e ambas apresentam um perfil bastante semelhante, como vimos acima, os resultados quanto a potência de bombeio deverão acompanhar tais perfis, como, de fato, se observa.

6.3.2.5. Resultados quanto ao peso sobre o gancho

Como é sabido o peso sobre o gancho é função do tamanho da coluna e da densidade do fluido de perfuração. Durante o avanço do poço existe uma tendência crescente do peso sobre

o gancho (*WOH*) devido ao fato de, conforme a profundidade aumenta também aumentará a coluna, bem como o seu peso, que será sustentado pelo gancho.

Todavia, durante as manobras, o perfil de *WOH* é alterado. No momento da parada de circulação e parada do poço a coluna perderá contato com o fundo do poço, e, conseqüentemente, esse excedente de peso deverá ser suportado pelo gancho. Assim, no instante inicial, haverá um pico no *WOH*. Ao longo da manobra a coluna vai sendo continuamente retirada, o que reduzirá o peso sobre o gancho até que não haja mais seções para serem sustentadas. Nesse ponto o *WOH* tenderá a zero. Quando a descida da coluna for reiniciada o peso sobre o gancho tenderá a aumentar até atingir o peso no instante inicial e, finalmente, a coluna toca o solo e o peso ser reajustado.

Nesse sentido o perfil do peso sobre o gancho como resultado do simulador é dado pelo gráfico a seguir.

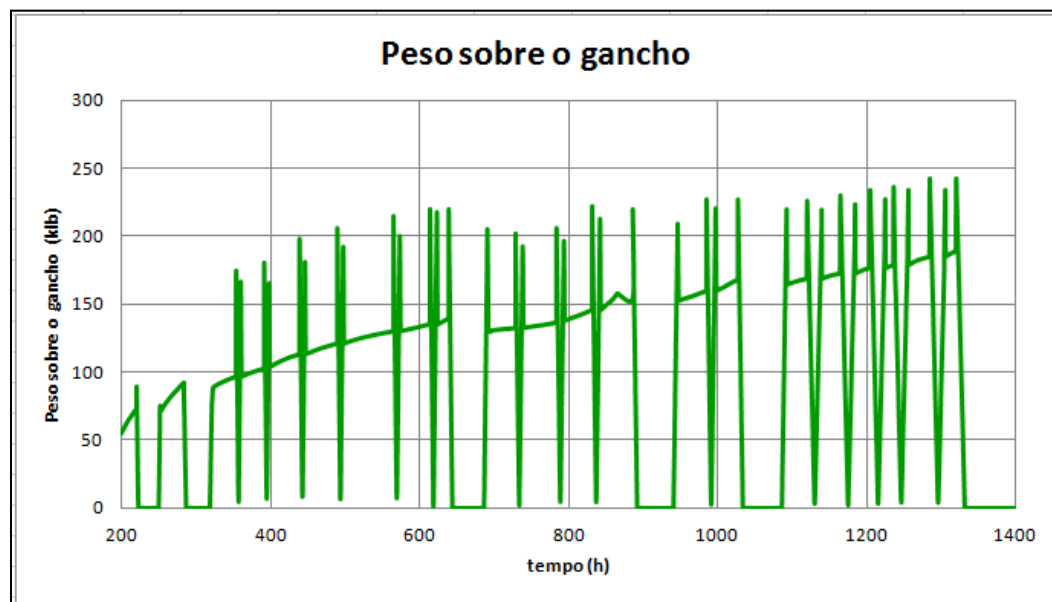


Figura 126 – Perfil do *WOH* ao longo do tempo de perfuração (elaboração própria baseada nos resultados do simulador proposto)

Podemos observar que nos pontos de manobra o modelo descrito acima é respeitado, havendo um pico, redução linear até zero, retomada até o pico e, finalmente, valor normal de operação, tudo isso dentro do intervalo de tempo de manobra. Já nos pontos de descida dos

revestimentos o *WOH* é mantido zero em todo o intervalo de tempo entre o içamento e a descida da coluna, já que estamos avaliando o *WOH* como suspensão da coluna apenas. O gráfico abaixo mostra mais nitidamente o perfil do peso durante as manobras, explicitando os eventos que foram tratados aqui.

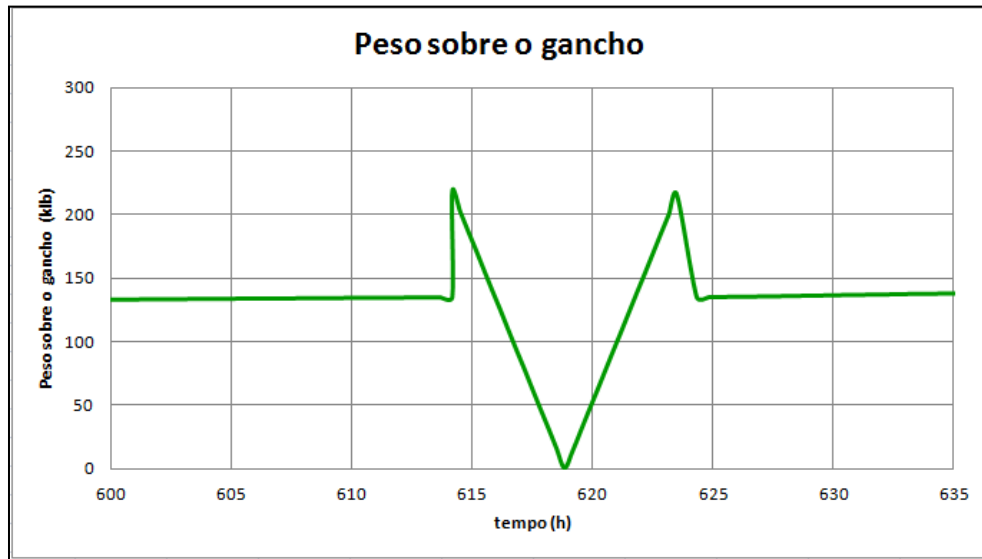


Figura 127 – Enfoque do comportamento do *WOH* durante uma manobra (elaboração própria baseada nos resultados do simulador proposto)

7. Conclusões e Recomendações Finais

Foi apresentado ao longo deste trabalho o simulador de perfuração intitulado UFRJDrillingSimulator, desenvolvido em C++, compreendendo o módulo de planejamento de poço e o perfuração em tempo real, possibilitando a atuação dinâmica do usuário frente ao comportamento apresentado pelo poço. Nesse sentido foram abordadas três vertentes principais do trabalho: (i) metodologia teórica e computacional que foi implementada, como forma a criar o simulador; (ii) manual de usuário contendo as informações necessárias para algum interessado em utilizar o programa e (iii) resultados fornecidos. Para este últimos desenvolveu-se um estudo de caso onde foi simulada a perfuração de um poço sintético, analisando os resultados fornecidos pelo programa nos seus diversos blocos.

Nesse sentido o simulador proposto apresentou, na visão do autor, resultados coerentes e razoáveis frente a realidade de uma atividade de perfuração. Foi apresentado ao longo do capítulo 6 que todos os resultados fornecidos foram condizentes com o modelo implementado, o que valida o código como um simulador de perfuração em funcionamento, apesar de apresentar uma gama diversa de simplificações embutidas.

Por fim é importante frisar que este é o início de um trabalho. Um simulador de perfuração, para ser considerado completo e comercial, demanda a implementação de diversos outros módulos e funcionalidades, de forma a aproximar os resultados fornecidos cada vez mais da realidade. E nesse sentido para as recomendações para trabalhos futuros. A seguir estão listados alguns tópicos que, na visão do autor, deveriam ser abordadas em um segundo momento, visando converter a ferramenta que se tem hoje em uma ferramenta comercial e mais completa:

- Implementar uma interface gráfica amigável (*GUI*) para o simulador, facilitando a interação entre o usuário e a ferramenta. Atualmente a interfície utilizada é DOS;
- Implementar um módulo de física de coluna, analisando por meio de métodos numéricos o comportamento estrutural da coluna de perfuração. Atualmente a coluna é vista como perfeitamente rígida;

- Reduzir as pequenas simplificações inerentes ao modelo e assim torná-lo mais dinâmico, como, por exemplo, limitar o intervalo de interpolação em um número inteiro por exemplo;
- Implementar a análise hidráulica do poço com um enfoque completo, aplicando a equação de Navier-Stokes para permitir que se observe o trecho transiente da pressão;
- Implementar um modelo de fraturamento hidráulico, de modo a estimar mais corretamente as perdas de circulação (vide Fjaer *et al.* 2008);
- Implementar um modelo de *kick*, avaliando a migração e circulação do influxo desde fundo do poço até o *choke* (vide Ohara *et al.*, 1998);
- Avaliar a perfuração com diferentes tipos de brocas, que não as tricônicas;
- Implementação de um módulo de otimização da atividade através de algoritmos genéticos.

8. Referências Bibliográficas

- Aadnoy, B. S. (1999). *Modern Well Design*. Stavenger, Noruega: Rogaland University Center.
- Borgoyne, A., Young, F. S., Chevernet, M. E., & Millhein, K. K. (1986). *Applied Drilling Engineering* (Vol. Volume 2). SPE Textbooks Series.
- Bowers, G. (1995). Pore Pressure Estimation From Velocity Data: Accounting for Overpressure Mechanisms Besides Undercompactions. *SPE Drilling and Completion* .
- Bowers, G. (1999). *State of the Art in Pore Pressure Estimation*. Houston: Knowledge System Inc.
- Couto, P., & Ohara, S. (2009). *Notas de Aula da Disciplina Perfuração de Poços (EEW411)*.
- Eaton, B. A. (1975). The Equation for Geopressure Prediction from Well Logs. *AIME*. Dallas: SPE 5544.
- Economides, M., Watters, L., & Dunn-Norman, S. (1997). *Petroleum Well Construction*. Duncan, OK: Halliburton Press.
- EconomyWatch.com*. (s.d.). Acesso em 24 de 09 de 2011, disponível em <http://www.economywatch.com/world-industries/oil/extraction.html>
- Fjaer, E., Holt, R. M., Horsrud, P., Raaen, A. M., & Risnes, R. (2008). *Petroleum Related Rock Mechanics*. Elsevier.
- Gardner, G. H., Gregory, A. R., & Willyie, M. R. (1958). An Experimental Investigation os Fators Affecting Elastic Wave Velocities in Porous Media. *Geophysics* , 459-493.
- Halliburton. (2008). *IADC Classification Chart*.
- Machado, J. C. (2002). *Reologia e Escoamento de Fluidos*. Rio de Janeiro: Editora Interciência.
- Oualine, S. (2003). *Practical C++ Programming*.
- Pasqualino, I. (2011). *Notas de Aula da Disciplina Completação de Poços (EEW412)*.
- Pastor, J. (2010). *Geomechanics Guide for a Complete ID MEM*. Schlumberger Internal Report.
- Rocha, L. S., & Azevedo, C. T. (2009). *Geopressões e Assentamento de Colunas de Revestimento*. Interciência.
- Rodrigues, J. (2010). *Notas de Aula da Disciplina Fluidos de Perfuração e Completação (EEW413)*.

Thomas, J. E., Triggia, A. A., Correa, C. A., Filho, C. V., Xavier, J. A., & Carvalho, R. S. (2001). *Fundamentos de Engenharia de Petróleo* (2a. Edição ed.). (J. E. Thomas, Ed.) Rio de Janeiro.

Tsukada, R., Yamamoto, M., & Mooroka, C. (2004). Análise de Operações de Instalações de Riser e Descida de BOP em Sondas Flutuantes de Perfuração. Campinas.

US Energy Information Administration. (2011, 09 02). Retrieved 09 24, 2011, from <http://www.eia.gov>

9. Anexos

```
//Well Project Code - Victor Silva

#include <iostream.h>
#include <fstream.h>
#include <math.h>
#include <conio.h>
#include <iomanip.h>
#include <GD_Paul.h>
#define pi 3.1416
#include <string.h>
#include <windows.h>
using namespace std;
#define clrscr system("cls")

#include "PIPES.h"
#include "FLUIDS.h"
#include "PUMP.h"
#include "BITE.h"
#include "REVEST.h"

//DECLARATION OF GENERIC FUNCTIONS

void LOGO_design()
{
    cout << "      *      * ***** ***** ***** ***** ***** * *      *      * **
* *****      " << endl;
    cout << "      *      * *      * *      *      *      * * * * * * * *      *      * * *
* *      " << endl;
    cout << "      *      * ***** *****      *      * * ***** * *      *      * * *
* ***      " << endl;
    cout << "      *      * *      * * * * *      * * * * * * * *      *      * * * *
* *      " << endl;
    cout << " ***** *      * * * * ***** ***** ***** * *      *
*****      " << endl;
}

double abs(double x)
{
    double y;
    if (x>=0) y=x;
    if (x<0) y=-x;
    return y;
}

double root(double a, double b, double c)
{
    double delta, x1, x2;

    delta = b*b-4*a*c;
    x1 = (-b + sqrt(delta))/(2*a);
    x2 = (-b - sqrt(delta))/(2*a);

    if (x1 >= 0) { if (x1 <= 1.0) return x1; }
    if (x2 >= 0) { if (x2 <= 1.0) return x2; }
    else if (x1 > 1.0) return 1.0;
    else if (x2 > 1.0) return 1.0;
}

//DECLARATION OF MODULO 1 FUNCTIONS
```

```

double CimV(double ODstage, double OD, double depth_cim)
{
    double cim_vol, A, ODstage_corr, OD_corr, depth_cim_corr;

    ODstage_corr = ODstage/12;
    OD_corr = OD/12;
    depth_cim_corr = depth_cim*3.28;

    A = pi*(pow(ODstage_corr,2)-pow(OD_corr,2))/4;
    cim_vol = (1/0.159)*A*depth_cim_corr/(pow(3.28,3)); //bbl
    return cim_vol;
}

double gradient_criteria (double depth, double density, double coef)
{
    double p_bh = coef*density*depth;
    return p_bh;
}

double riser_margin_criteria (double LDA, double depth, double grad_porous)
{
    double MSR, grad_water=8.5, p_bh, press;

    MSR = (grad_porous*(LDA+depth)-grad_water*depth)/depth;
    p_bh = 0.052*(depth+LDA)*MSR;
    return p_bh;
}

double baritina(double density_new, double density_old)
{
    double qnt_baritina = 1500*(density_new-density_old)/(35.8-density_new);
    return qnt_baritina;
}

double casing(double sigma, double density, double density_new, double depth_new,
double depth_old, double OD, double thick, double coef, double d_aco, double
d_gas)//function to return the type
{
    double spe_weigth, stress, int_press, lim_stress, lim_int_press, d, area,
ext_press, press_sup, ID;
    double p_eq;

    L1:

    if (thick > 2) return thick;

    spe_weigth = 0.25*3.1415*(pow(OD/12,2)-pow((OD-2*thick)/12,2))*65.5/0.13;
    stress = 1.15*spe_weigth*(depth_new*3.28)*(1-density/d_aco);

    ID = (OD-2*thick);

    ext_press = coef*d_gas*depth_new;
    int_press = coef*density_new*depth_new;

    p_eq = 1.15/OD * (int_press*ID - ext_press*OD);

    area = 0.25*pi*(pow(OD,2)-pow(ID,2));

    lim_stress = area*sigma;
    lim_int_press = 0.875*(2*thick*sigma)/OD;

    if (stress > lim_stress) { thick=thick+0.01; goto L1; }
}

```

```

    if (p_eq > lim_int_press) { thick=thick+0.01; goto L1; }
    return thick;
}

double cond_sup_stress(double sigma, double depth_new, double density, double d_aco,
double OD, double thick)
{
    double spe_weigth, stress, ID, area, lim_stress, thick1 = thick;

    L:
    spe_weigth = 0.25*3.1415*(pow(OD/12,2)-pow((OD-2*thick1)/12,2))*65.5/0.13;
    stress = 1.25*spe_weigth*(depth_new*3.28)*(1-density/d_aco);
    ID = (OD-2*thick);
    area = 0.25*pi*(pow(OD,2)-pow(ID,2));
    lim_stress = area*sigma;

    if (stress > lim_stress) { thick = thick+0.01; goto L; }
    return thick;
}

double colapse(double sigma, double density, double OD, double thick, double depth,
double coef, double LDA, double Gf)
{
    double razao, sigma_colapse[25], F1[25], F2[25], F3[25], F4[25], F5[25], F_1,
F_2, F_3, F_4, F_5;
    double sigma_1, razao_yp, razao_te, razao_pt, lim_ext_press, ext_press, ID,
p_eq, int_press;
    int m, j=0;
    ifstream entrada;

    entrada.open("C:/UFRJDrillingSimulator/Input/InputFCoefficient.txt");

    while (j<22)
    {
        entrada >> sigma_colapse[j] >> F1[j] >> F2[j] >> F3[j] >> F4[j] >> F5[j];
        j = j+1;
        m=0;

        T:{
        if (sigma == sigma_colapse[m])
        {F_1=F1[m]; F_2=F2[m]; F_3=F3[m]; F_4=F4[m]; F_5=F5[m];}
        else {m=m+1; goto T;}}

        ID = OD-2*thick;

        ext_press = coef*Gf*depth;
        int_press = coef*density*depth;

        p_eq = 1.15/OD * (ext_press*OD - int_press*ID);

        razao=OD/thick;

        V: {
        sigma_1=sigma;
        razao_yp = 0.5*(sqrt(pow((F_1-2),2)+8*(F_2+F_3/sigma_1)))+(F_1-
2))/(F_2+F_3/sigma_1);
        razao_te = (2+F_2/F_1)/(3*F_1/F_2);
        razao_pt = (sigma_1*(F_1 - F_4))/(F_3+sigma_1*(F_2-F_5));

        if (razao <= razao_yp) lim_ext_press = 2*sigma_1*(razao-1)/pow(razao,2);
        else if (razao <= razao_pt) lim_ext_press = sigma_1*(F_1/razao - F_2) - F_3;
        else if (razao <= razao_pt) lim_ext_press = sigma_1*(F_4/razao - F_5);
        else lim_ext_press = 46950000/(razao*pow(razao-1,2));
        }
}

```

```

    if (p_eq > lim_ext_press) {thick = thick+0.01; goto T;}
    return thick;

    entrada.close();
}

void drawing(double depth, double x, double y, double xpmax, double ypmax, double
x_dif, double cement, int final_depth, double thick, double depth_old, double
depth_old_old, double x_1, int cg, double corr, double x_corr)
{
    double y_pos=y, x_pos=x, y_pos1, thick_pixel, y_pos2;
    int x_pixel_0, x_pixel, x_pixel_1, y_pixel_0, y_pixel, y_pixel_1, y_pixel_2,
x_pixel_2;

    pensize(2);
    bgcolor("gray");

    x_pixel_0 = 130 + x_pos*(xpmax-150)/50;
    y_pixel_0 = 20 + y_pos*(ypmax-40)/(final_depth);
    x_pixel_1 = 130 + x_dif*(xpmax-150)/50;

    if (cg==0) y_pos1 = y_pos + depth;
    else y_pos1 = y_pos + (depth-depth_old_old);

    y_pixel_2 = 20 + (y_pos1)*(ypmax-40)/(final_depth);
    y_pixel_1 = 20 + (y_pos1-cement)*(ypmax-40)/(final_depth);
    x_pixel_2 = 130 + x_1*(xpmax-150)/50;

    if (cg==0) y_pos2 = y_pos + depth_old;
    else y_pos2 = y_pos + (depth_old-depth_old_old);

    y_pixel = 20 + y_pos2*(ypmax-40)/(final_depth);

    fgcolor("gray");
    rectangle(x_pixel_1,y_pixel_1,x_pixel_2,y_pixel_2);
    rectangle(x_pixel_2,y_pixel_1,x_pixel_1,y_pixel_2);
    rectangle(x_pixel_0,y_pixel_1,x_pixel_1,y_pixel);
    rectangle(x_pixel_2,y_pixel_1,x_pixel_0,y_pixel);
    thick_pixel = 0.9*thick*(xpmax-50)/50;

    fgcolor("black");
    pensize(thick_pixel);
    moveto(x_pixel_0,y_pixel_0);

    if (cg==0) y_pos = y_pos + depth_old;
    else y_pos = y_pos + (depth_old-depth_old_old);

    y_pixel = 20 + y_pos*(ypmax-40)/(final_depth);
    lineto(x_pixel_0,y_pixel);

    double y_pos_3 = y + corr/2, x_pos_3 = x_corr;
    int y_pixel_3 = 20 + y_pos_3*(ypmax-40)/(final_depth);
    int x_pixel_3 = 130 + x_pos_3*(xpmax-150)/50;

    if (cg==1) {pensize(12); moveto(x_pixel_0,y_pixel_3);
    lineto(x_pixel_3,y_pixel_3);}
}

void contorno_drawing_cond (double xpmax, double ypmax, double xpos1, double ypos1,
double xpos2, double ypos2, double final_depth)
{
    double x_pixel, y_pixel, x_pixel_2, y_pixel_2;

```



```

fgcolor("black");
x_pixel = 130 + xpos1*(xpmax-150)/50;
y_pixel = 20 + ypos1*(ypmax-40)/(final_depth);
x_pixel_2 = 130 + xpos2*(xpmax-150)/50;
y_pixel_2 = 20 + ypos2*(ypmax-40)/(final_depth);

fgcolor("black");
pensize(2.5);
moveto(x_pixel,y_pixel);
lineto(x_pixel,y_pixel_2);
lineto(x_pixel_2,y_pixel_2);
}

void contorno_drawing (double xpmax, double ypmax, double xpos1, double ypos1, double
xpos2, double ypos2, double xpos3, double ypos3, double final_depth)
{
    double x_pixel, y_pixel, x_pixel_2, y_pixel_2, x_pixel_3, y_pixel_3;

    fgcolor("black");
    x_pixel = 130 + xpos1*(xpmax-150)/50;
    y_pixel = 20 + ypos1*(ypmax-40)/(final_depth);
    x_pixel_2 = 130 + xpos2*(xpmax-150)/50;
    y_pixel_2 = 20 + ypos2*(ypmax-40)/(final_depth);
    x_pixel_3 = 130 + xpos3*(xpmax-150)/50;
    y_pixel_3 = 20 + ypos3*(ypmax-40)/(final_depth);

    pensize(2.5);
    moveto(x_pixel,y_pixel);
    lineto(x_pixel,y_pixel_2);
    lineto(x_pixel_2,y_pixel_2);
    lineto(x_pixel_2,y_pixel_3);
    lineto(x_pixel_3,y_pixel_3);
}

void gradmarg_drawing(double xpmax, double ypmax, double Gp, double Gf, double depth,
double final_depth, double Gpmin, double Gov)
{
    double Gp_pixel, Gf_pixel, Gov_pixel, depth_pixel,C;

    C = Gov-Gpmin;
    Gp_pixel = 130 + (Gp-Gpmin)*(xpmax-150)/C;
    Gf_pixel = 130 + (Gf-Gpmin)*(xpmax-150)/C;
    depth_pixel = 20 + (depth)*(ypmax-40)/(final_depth);

    fgcolor("blue");
    moveto(Gp_pixel,depth_pixel);
    ellipse(2.5,2.5);

    fgcolor("green");
    moveto(Gf_pixel,depth_pixel);
    ellipse(2.5,2.5);
}

void grad_drawing(double xpmax, double ypmax, double Gp, double Gp_ant, double Gf,
double Gf_ant, double Gov, double Gov_ant, double depth, double depth_ant, double
final_depth, double Gpmin, double Gov_max, int ind)
{
    double Gp_pixel, Gp_pixel_ant, Gf_pixel, Gf_pixel_ant, Gov_pixel_ant, Gov_pixel,
depth_pixel_ant, depth_pixel, RFTp1_pixel, RFTp2_pixel, C;

    C = Gov_max-Gpmin;
    Gp_pixel = 130 + (Gp-Gpmin)*(xpmax-150)/C;

```

```

Gp_pixel_ant = 130 + (Gp_ant-Gpmin)*(xpmax-150)/C;
Gf_pixel = 130 + (Gf-Gpmin)*(xpmax-150)/C;
Gf_pixel_ant = 130 + (Gf_ant-Gpmin)*(xpmax-150)/C;
Gov_pixel = 130 + (Gov-Gpmin)*(xpmax-150)/C;
Gov_pixel_ant = 130 + (Gov_ant-Gpmin)*(xpmax-150)/C;
depth_pixel = 20 + (depth)*(ypmax-40)/(final_depth);
depth_pixel_ant = 20 + (depth_ant)*(ypmax-40)/(final_depth);

if (ind==0)
{
pensize(4);
fgcolor("darkblue");
}
if (ind==1)
{
pensize(2);
fgcolor("lightblue");
}
if (ind==2)
{
pensize(2.5);
fgcolor("magenta");
}

moveto(Gp_pixel_ant,depth_pixel_ant);
lineto(Gp_pixel,depth_pixel);

if (ind==0)
{
pensize(4);
fgcolor("darkgreen");
}
if (ind==1)
{
pensize(2);
fgcolor("lightgreen");
}
if (ind==2)
{
pensize(2.5);
fgcolor("red");
}

moveto(Gf_pixel_ant,depth_pixel_ant);
lineto(Gf_pixel,depth_pixel);

fgcolor("brown");
moveto(Gov_pixel_ant,depth_pixel_ant);
lineto(Gov_pixel,depth_pixel);
}

void pontos_drawing(int xpmax, int ypmax, int RFTp1, double RFTd1, int RFTp2, double
RFTd2, double Df1, double G_LOT1, double Df2, double G_LOT2, double Df3, double
G_LOT3, double final_depth, double Gpmin, double Gov)
{
double RFTd1_pixel, RFTd2_pixel, Df1_pixel, Df2_pixel, Df3_pixel, G_LOT1_pixel,
G_LOT2_pixel, G_LOT3_pixel, RFTp1_pixel, RFTp2_pixel,C;

C = Gov-Gpmin;
RFTp1_pixel = 20 + (RFTp1)*(ypmax-40)/(final_depth);
RFTp2_pixel = 20 + (RFTp2)*(ypmax-40)/(final_depth);
RFTd1_pixel = 130 + (RFTd1-Gpmin)*(xpmax-150)/C;
RFTd2_pixel = 130 + (RFTd2-Gpmin)*(xpmax-150)/C;

```

```

Df1_pixel = 20 + (Df1)*(ypmax-40)/(final_depth);
Df2_pixel = 20 + (Df2)*(ypmax-40)/(final_depth);
Df3_pixel = 20 + (Df3)*(ypmax-40)/(final_depth);

G_LOT1_pixel = 130 + (G_LOT1-Gpmin)*(xpmax-150)/C;
G_LOT2_pixel = 130 + (G_LOT2-Gpmin)*(xpmax-150)/C;
G_LOT3_pixel = 130 + (G_LOT3-Gpmin)*(xpmax-150)/C;

fgcolor("darkblue");
bgcolor("darkblue");

moveto(RFTd1_pixel, RFTp1_pixel);
ellipse(10,10);
moveto(RFTd2_pixel, RFTp2_pixel);
ellipse(10,10);

fgcolor("darkgreen");
bgcolor("darkgreen");

moveto(G_LOT1_pixel,Df1_pixel);
ellipse(10,10);

moveto(G_LOT2_pixel,Df2_pixel);
ellipse(10,10);

moveto(G_LOT3_pixel,Df3_pixel);
ellipse(10,10);
}

void gradcrit_drawing(double xpmax, double ypmax, double density, double density_ant,
double depth, double depth_ant, double final_depth, double Gpmin, int i, double Gov)
{
    double density_pixel, density_pixel_ant, depth_pixel_ant, depth_pixel,C;

    C = Gov - Gpmin;
    density_pixel = 130 + (density-Gpmin)*(xpmax-150)/C;
    density_pixel_ant = 130 + (density_ant-Gpmin)*(xpmax-150)/C;
    depth_pixel = 20 + (depth)*(ypmax-40)/(final_depth);
    depth_pixel_ant = 20 + (depth_ant)*(ypmax-40)/(final_depth);

    pensize(3);

    fgcolor("magenta");
    moveto(density_pixel,depth_pixel);
    lineto(density_pixel,depth_pixel_ant);
    if (i=1){
    lineto(density_pixel_ant,depth_pixel_ant);}
}

void gradkick_drawing(double xpmax, double ypmax, double density, double density_ant,
double depth, double depth_ant, double final_depth, double Gpmin, double Gov_max, int
index)
{
    double d_pixel, d_pixel_ant, depth_pixel_ant, depth_pixel, C;

    C = Gov_max-Gpmin;
    d_pixel = 130 + (density-Gpmin)*(xpmax-150)/C;
    d_pixel_ant = 130 + (density_ant-Gpmin)*(xpmax-150)/C;
    depth_pixel = 20 + (depth)*(ypmax-40)/(final_depth);
    depth_pixel_ant = 20 + (depth_ant)*(ypmax-40)/(final_depth);

    if (index == 0) { fgcolor("yellow"); pensize(3); }
}

```

```

    if (index == 1) { fgcolor("magenta"); pensize(3); }

    moveto(d_pixel_ant,depth_pixel_ant);
    lineto(d_pixel,depth_pixel);
}

void litology_drawing(int xpmax, int ypmax, int lito, double depth, double depth_ant,
double final_depth)
{
    double x0_pixel = 40, x1_pixel = 110, depth_ant_pixel, depth_pixel;

    pensize(1);

    if (lito==1) {fgcolor("gray"); bgcolor("gray"); }
    if (lito==2) {fgcolor("yellow"); bgcolor("yellow"); }
    if (lito==3) {fgcolor("darkblue"); bgcolor("darkblue"); }

    depth_ant_pixel = 20 + (depth_ant)*(ypmax-40)/(final_depth);
    depth_pixel = 20 + (depth)*(ypmax-40)/(final_depth);

    rectangle(x0_pixel,depth_ant_pixel,x1_pixel,depth_pixel);
}

void circle(double x, double y, double xpmax, double ypmax, double d_ext, int N, int
final_depth, double thick, int color)
{
    double y_pos = y, x_pos = x, thick_pixel;
    int x_pixel, y_pixel, x_1, y_1;

    if (color==1) {fgcolor("gray"); bgcolor("gray");}
    else fgcolor("black");
    thick_pixel = 0.8*thick*(xpmax-40)/50;
    pensize(thick_pixel);
    if (N==1) bgcolor("gray");
    if (N==2) bgcolor("white");
    x_pixel = 20 + x_pos*(xpmax-40)/50;
    y_pixel = 20 + y_pos*(ypmax-40)/(final_depth);
    x_1 = 0.85*(20 + d_ext*(xpmax-40)/50);
    moveto(x_pixel,y_pixel);
    ellipse(x_1,x_1);
}

void drawing_final_casing(double depth, double x, double y, int xpmax, int ypmax,
double thick, int final_depth, double corr, int cg, double x_corr)
{
    double x_pixel_0, y_pixel_0, y_pos1, y_pixel_1, thick_pixel;

    x_pixel_0 = 130 + x*(xpmax-150)/50;
    y_pixel_0 = 20 + y*(ypmax-40)/(final_depth);
    y_pos1 = y + depth;
    y_pixel_1 = 20 + (y_pos1)*(ypmax-40)/(final_depth);

    thick_pixel = 0.8*thick*(xpmax-50)/50;
    fgcolor("black");
    pensize(thick_pixel);
    moveto(x_pixel_0,y_pixel_0);
    lineto(x_pixel_0,y_pixel_1);

    double y_pos_3 = y + corr/2, x_pos_3 = x_corr;
    int y_pixel_3 = 20 + y_pos_3*(ypmax-40)/(final_depth);
    int x_pixel_3 = 130 + x_pos_3*(xpmax-150)/50;

```

```

    if (cg==1) {pensize(12); moveto(x_pixel_0,y_pixel_3);
lineto(x_pixel_3,y_pixel_3);}
}

```

```

void scale_y(int f, int j)

```

```

{
    fgcolor("black");
    if (f==1) {
        if (j==1) write_string("250");
        if (j==2) write_string("500");
        if (j==3) write_string("750");
        if (j==4) write_string("1000");
        if (j==5) write_string("1250");
        if (j==6) write_string("1500");
        if (j==7) write_string("1750");
        if (j==8) write_string("2000");
        if (j==9) write_string("2250");
        if (j==10) write_string("2500"); }

    if (f==2) {
        if (j==1) write_string("500");
        if (j==2) write_string("1000");
        if (j==3) write_string("1500");
        if (j==4) write_string("2000");
        if (j==5) write_string("2500");
        if (j==6) write_string("3000");
        if (j==7) write_string("3500");
        if (j==8) write_string("4000");
        if (j==9) write_string("4500");
        if (j==10) write_string("5000"); }

    if (f==3) {
        if (j==1) write_string("750");
        if (j==2) write_string("1500");
        if (j==3) write_string("2250");
        if (j==4) write_string("3000");
        if (j==5) write_string("3750");
        if (j==6) write_string("4500");
        if (j==7) write_string("5250");
        if (j==8) write_string("6000");
        if (j==9) write_string("6750");
        if (j==10) write_string("7500"); }
}

```

```

void scale_x(int i)

```

```

{
    fgcolor("black");

    if (i==0) write_string("8.5");
    if (i==1) write_string("9.5");
    if (i==2) write_string("10.5");
    if (i==3) write_string("11.5");
    if (i==4) write_string("12.5");
    if (i==5) write_string("13.5");
    if (i==6) write_string("14.5");
    if (i==7) write_string("15.5");
    if (i==8) write_string("16.5");
    if (i==9) write_string("17.5");
    if (i==10) write_string("18.5");
    if (i==11) write_string("19.5");
    if (i==12) write_string("20.5");
}

```

```

}

void scale_time_x(int i)
{
    fgcolor("black");

    if (i==0) write_string("0");
    if (i==1) write_string("10");
    if (i==2) write_string("20");
    if (i==3) write_string("30");
    if (i==4) write_string("40");
    if (i==5) write_string("50");
    if (i==6) write_string("60");
    if (i==7) write_string("70");
    if (i==8) write_string("80");
    if (i==9) write_string("90");
    if (i==10) write_string("100");
    if (i==11) write_string("110");
    if (i==12) write_string("120");
    if (i==13) write_string("130");
    if (i==14) write_string("140");
    if (i==15) write_string("150");
}

void scale_cost_x(int i)
{
    fgcolor("black");

    if (i==0) write_string("0");
    if (i==1) write_string("5");
    if (i==2) write_string("10");
    if (i==3) write_string("15");
    if (i==4) write_string("20");
    if (i==5) write_string("25");
    if (i==6) write_string("30");
    if (i==7) write_string("35");
    if (i==8) write_string("40");
    if (i==9) write_string("45");
    if (i==10) write_string("50");
    if (i==11) write_string("55");
    if (i==12) write_string("60");
    if (i==13) write_string("65");
    if (i==14) write_string("70");
}

void scale_ROP_x(int i)
{
    if (i==0) write_string("0");
    if (i==1) write_string("5");
    if (i==2) write_string("10");
    if (i==3) write_string("15");
    if (i==4) write_string("20");
    if (i==5) write_string("25");
    if (i==6) write_string("30");
    if (i==7) write_string("35");
    if (i==8) write_string("40");
    if (i==9) write_string("45");
    if (i==10) write_string("50");
}

void scale_x_RHOB(int i)
{

```

```

        fgcolor("black");

        if (i==0) write_string("1.7");
        if (i==1) write_string("1.9");
        if (i==2) write_string("2.1");
        if (i==3) write_string("2.3");
        if (i==4) write_string("2.5");
        if (i==5) write_string("2.7");
    }

void scale_x_TREND(int i)
{
    fgcolor("black");

    if (i==0) write_string("10");
    if (i==1) write_string("100");
    if (i==2) write_string("1000");
}

void scale_DTCO (int i)
{
    fgcolor("black");

    if (i==0) write_string("280");
    if (i==1) write_string("240");
    if (i==2) write_string("200");
    if (i==3) write_string("160");
    if (i==4) write_string("120");
    if (i==5) write_string("80");
    if (i==6) write_string("40");
}

void gridding(double final_depth, int grid, double xpmx, double ypmx, double Gov,
double Gpmin)
{
    double x_pixel, y_pixel, x, y=10, x_dif, y_pixel_0, y_dif, C;
    int i=0, j=0, f;

    fgcolor("gray");
    pensize(1);

    if (final_depth < 2500) { y_dif = 250; f=1; }
    else if (final_depth < 5000) { y_dif = 500; f=2; }
    else { y_dif = 750; f=3; }

    x_dif = 1;
    x = 8.5 - Gpmin;
    y_pixel_0 = 20 + y*(ypmx-40)/(final_depth);

    while (y < final_depth)
    {
        fgcolor("gray");
        y_pixel = 20 + y*(ypmx-40)/(final_depth);
        moveto(30,y_pixel); lineto(xpmx-20,y_pixel);
        moveto(5,y_pixel+5); scale_y(f,j); j=j+1;
        y = y + y_dif;
    }

    fgcolor("gray");
    i=0;
    C = Gov - Gpmin;
    while (x < Gov-Gpmin)
    {

```

```

    x_pixel = 130 + x*(xpmax-150)/C;
    moveto(x_pixel,y_pixel_0); lineto(x_pixel,ypmax-20);
    moveto(x_pixel-5,y_pixel_0-5); scale_x(i); i=i+1;
    x = x + x_dif;
}

x_pixel = 130 + x*(xpmax-150)/C;
moveto(x_pixel,y_pixel_0); lineto(x_pixel,ypmax-20);
moveto(x_pixel-5,y_pixel_0-5); scale_x(i);

fgcolor("black");
moveto(xpmax-250,40); write_string("WELL DESIGN - PRESSURES ESTIMATION");
moveto(xpmax-250,55); write_string("SCALE: TVDBML: m");
moveto(xpmax-250,70); write_string("      GRAD:  ppg");
}

void gridding_TIME(double final_depth, double xpmax, double ypmax, double time_max,
double ROPmax)
{
    double x_pixel, y_pixel, x, y=10, x_dif, y_pixel_0, y_dif, C, C2;
    int i=0, j=0, f;

    fgcolor("gray");
    pensize(1);

    if (final_depth < 2500) { y_dif = 250; f=1; }
    else if (final_depth < 5000) { y_dif = 500; f=2; }
    else { y_dif = 750; f=3; }

    x_dif = 10;
    x = 0;
    y_pixel_0 = 20 + y*(ypmax-40)/(final_depth);

    while (y < final_depth)
    {
        fgcolor("gray");
        y_pixel = 20 + y*(ypmax-40)/(final_depth);
        moveto(30,y_pixel); lineto(xpmax-20,y_pixel);
        moveto(5,y_pixel+5); scale_y(f,j); j=j+1;
        y = y + y_dif;
    }

    fgcolor("gray");
    i=0;
    C = time_max/24;
    C2 = ROPmax;

    while (x < time_max)
    {
        x_pixel = 130 + x*(xpmax-150)/C;
        moveto(x_pixel,y_pixel_0); lineto(x_pixel,ypmax-20);
        moveto(x_pixel-5,y_pixel_0-5); scale_time_x(i); i=i+1;
        x = x + x_dif;
    }

    x=0;
    i=0;
    x_dif = 5;
    pensize(1);
    while (x < ROPmax)
    {
        x_pixel = 130 + x*(xpmax-150)/C2;
        moveto(x_pixel,y_pixel_0); fgcolor ("lightblue"); lineto(x_pixel,ypmax-20);
        moveto(x_pixel-5,763); fgcolor ("gray"); scale_ROP_x(i); i=i+1;
    }
}

```



```

    x = x + x_dif;
}

fgcolor("black");
moveto(xpmax-250,40); write_string("WELL PLANING - TIME ESTIMATION");
moveto(xpmax-250,55); write_string("SCALE: TVDBML:    m");
moveto(xpmax-250,70); write_string("    AC. TIME:  Days");
moveto(xpmax-250,85); write_string("    ROP:      m/h");
}

void gridding_COST(double final_depth, double xpmax, double ypmax, double Cmax)
{
    double x_pixel, y_pixel, x, y=10, x_dif, y_pixel_0, y_dif, C;
    int i=0, j=0, f;

    fgcolor("gray");
    pensize(1);

    if (final_depth < 2500) { y_dif = 250; f=1; }
    else if (final_depth < 5000) { y_dif = 500; f=2; }
    else { y_dif = 750; f=3; }

    x_dif = 5;
    x = 0;
    y_pixel_0 = 20 + y*(ypmax-40)/(final_depth);

    while (y < final_depth)
    {
        fgcolor("gray");
        y_pixel = 20 + y*(ypmax-40)/(final_depth);
        moveto(30,y_pixel); lineto(xpmax-20,y_pixel);
        moveto(5,y_pixel+5); scale_y(f,j); j=j+1;
        y = y + y_dif;
    }

    fgcolor("gray");
    i=0;
    C = Cmax/(pow(10,6));

    while (x < C)
    {
        x_pixel = 30 + x*(xpmax-50)/C;
        moveto(x_pixel,y_pixel_0); lineto(x_pixel,ypmax-20);
        moveto(x_pixel-5,y_pixel_0-5); scale_cost_x(i); i=i+1;
        x = x + x_dif;
    }

    fgcolor("black");
    moveto(xpmax-250,40); write_string("WELL PLANING - COST ESTIMATION");
    moveto(xpmax-250,55); write_string("SCALE: TVDBML:    m");
    moveto(xpmax-250,70); write_string("    AC. COST:  $ x 1e6");
}

void cost_drawing(double xpmax, double ypmax, double cost_sum, double cost_sum_ant,
double depth, double depth_ant, double final_depth, double cost_max)
{
    double cost_sum_pixel, cost_sum_pixel_ant, time_pixel_ant, depth_pixel_ant,
depth_pixel, C;

    C = cost_max/(pow(10,6));
    cost_sum_pixel = 30 + (cost_sum/(pow(10,6)))*(xpmax-50)/C;
    cost_sum_pixel_ant = 30 + (cost_sum_ant/(pow(10,6)))*(xpmax-50)/C;

```

```

depth_pixel = 20 + (depth)*(ypmax-40)/(final_depth);
depth_pixel_ant = 20 + (depth_ant)*(ypmax-40)/(final_depth);

pensize(3);
fgcolor("darkgreen");
moveto(cost_sum_pixel_ant,depth_pixel_ant);
lineto(cost_sum_pixel,depth_pixel);
}

void time_drawing(double xpmx, double ypmx, double ROP, double ROP_ant, double
time_sum, double time_sum_ant, double depth, double depth_ant, double final_depth,
double time_max, double ROPmax, int type)
{
    double time_sum_pixel, time_sum_pixel_ant, ROP_pixel, ROP_pixel_ant,
time_pixel_ant, depth_pixel_ant, depth_pixel, C, C2;

    C = time_max/24+2;
    C2 = ROPmax;
    time_sum_pixel = 130 + (time_sum/24)*(xpmx-150)/C;
    time_sum_pixel_ant = 130 + (time_sum_ant/24)*(xpmx-150)/C;
    ROP_pixel = 130 + ROP*(xpmx-150)/C2;
    ROP_pixel_ant = 130 + ROP_ant*(xpmx-150)/C2;

    depth_pixel = 20 + (depth)*(ypmx-40)/(final_depth);
    depth_pixel_ant = 20 + (depth_ant)*(ypmx-40)/(final_depth);

    if (type==1) {
    pensize(4);
    fgcolor("darkblue"); }
    if (type==2) {
    pensize(3);
    fgcolor("red"); }

    moveto(time_sum_pixel_ant,depth_pixel_ant);
    lineto(time_sum_pixel,depth_pixel);

    if (type==1) {
    pensize(2.5);
    fgcolor("lightblue"); }
    if (type==2) {
    pensize(2);
    fgcolor("orange"); }

    moveto(ROP_pixel_ant,depth_pixel_ant);
    lineto(ROP_pixel,depth_pixel);
}

void gridding_RHOB(double final_depth, double xpmx, double ypmx)
{
    double x_pixel, y_pixel, x, y=10, x_dif, y_pixel_0, y_dif;
    int i=0, j=0, f;

    fgcolor("gray");
    pensize(1);

    if (final_depth < 2500) { y_dif = 250; f=1; }
    else if (final_depth < 5000) { y_dif = 500; f=2; }
    else { y_dif = 750; f=3; }

    x_dif = 0.2;
    x = 0;
    y_pixel_0 = 20 + y*(ypmx-40)/(final_depth);

```

```

while (y < final_depth)
{
    fgcolor("gray");
    y_pixel = 20 + y*(ypmax-40)/(final_depth);
    moveto(30,y_pixel); lineto(xpmax-20,y_pixel);
    moveto(5,y_pixel+5); scale_y(f,j); j=j+1;
    y = y + y_dif;
}

fgcolor("gray");
i=0;
while (x < 1)
{
    x_pixel = 30 + x*(xpmax-50);
    moveto(x_pixel,y_pixel_0); lineto(x_pixel,ypmax-20);
    moveto(x_pixel-5,y_pixel_0-5); scale_x_RHOB(i); i=i+1;
    x = x + x_dif;
}

x_pixel = 30+(xpmax-50);
moveto(x_pixel-5,y_pixel_0-5); scale_x_RHOB(5);

fgcolor("black");
moveto(xpmax-275,40); write_string("WELL DESIGN - FORMATION DENSITY
ESTIMATION");
moveto(xpmax-275,55); write_string("SCALE: TVDBML: m");
moveto(xpmax-275,70); write_string("          DENSITY: g/cm3");
}

void gridding_TREND(double final_depth, double xpmax, double ypmax)
{
    double x_pixel, y_pixel, x, y=10, x_dif, y_pixel_0, y_dif;
    int i=0, j=0, f;

    fgcolor("gray");
    pensize(1);

    if (final_depth < 2500) { y_dif = 250; f=1; }
    else if (final_depth < 5000) { y_dif = 500; f=2; }
    else { y_dif = 750; f=3; }

    int x_dif1 = 1; int x_dif2 = 10;
    x = 2;
    y_pixel_0 = 20 + y*(ypmax-40)/(final_depth);

    while (y < final_depth)
    {
        fgcolor("gray");
        y_pixel = 20 + y*(ypmax-40)/(final_depth);
        moveto(30,y_pixel); lineto(xpmax-20,y_pixel);
        moveto(5,y_pixel+5); scale_y(f,j); j=j+1;
        y = y + y_dif;
    }

    fgcolor("gray");
    i=0;

    moveto(130,y_pixel_0-5); lineto(130,ypmax-20);
    moveto(130,y_pixel_0-5); scale_x_TREND(i); i=i+1;

    while (x < 10)
    {
        x_pixel = 130+log10(x)*(xpmax-150)/2;

```

```

        moveto(x_pixel,y_pixel_0); lineto(x_pixel,ypmax-20);
        x = x + x_dif1;
    }

x_pixel = 130+log10(x)*(xpmax-150)/2;
moveto(x_pixel-10,y_pixel_0-5); scale_x_TREND(i); i=i+1;

while (x <= 100)
{
    x_pixel = 130+log10(x)*(xpmax-150)/2;
    moveto(x_pixel,y_pixel_0); lineto(x_pixel,ypmax-20);
    x = x + x_dif2;
}

moveto(x_pixel-10,y_pixel_0-5); scale_x_TREND(i);

    fgcolor("black");
    moveto(xpmax-275,40); write_string("WELL DESIGN - EATON TREND CURVE");
    moveto(xpmax-275,55); write_string("SCALE: TVDBML: m");
    moveto(xpmax-275,70); write_string("          DTCO:  us/ft");
}

void gridding_FINAL(double final_depth, double xpmax, double ypmax)
{
    double x_pixel, y_pixel, x, y=10, x_dif, y_pixel_0, y_dif;
    int i=0, j=0, f;

    fgcolor("gray");
    pensize(1);

    if (final_depth < 2500) { y_dif = 250; f=1; }
    else if (final_depth < 5000) { y_dif = 500; f=2; }
    else { y_dif = 750; f=3; }

    x_dif = 0.2;
    x = 0;
    y_pixel_0 = 20 + y*(ypmax-40)/(final_depth);

    while (y < final_depth)
    {
        fgcolor("gray");
        y_pixel = 20 + y*(ypmax-40)/(final_depth);
        moveto(30,y_pixel); lineto(xpmax-20,y_pixel);
        moveto(5,y_pixel+5); scale_y(f,j); j=j+1;
        y = y + y_dif;
    }
}

void gridding_DTCOxRHOB(double final_depth, double xpmax, double ypmax)
{
    double x_pixel, y_pixel, x, y=0, x_dif, y_pixel_0, y_dif = 40;
    int i=0, j=0, f;

    fgcolor("gray");
    pensize(1);

    x_dif = 0.2;
    x = 0;
    y_pixel_0 = 20 + y*(ypmax-40)/200;

    while (y < 240)
    {
        fgcolor("gray");

```

```

        y_pixel = 20 + y*(ypmax-40)/240;
        moveto(30,y_pixel); lineto(xpmax-20,y_pixel);
        moveto(5,y_pixel+5); scale_DTCO(j); j=j+1;
        y = y + y_dif;
    }

fgcolor("gray");
i=0;
while (x < 1)
{
    x_pixel = 30 + x*(xpmax-50);
    moveto(x_pixel,y_pixel_0); lineto(x_pixel,ypmax-20);
    moveto(x_pixel-5,y_pixel_0-5); scale_x_RHOB(i); i=i+1;
    x = x + x_dif;
}

x_pixel = 30+(xpmax-50);
moveto(x_pixel-5,y_pixel_0-5); scale_x_RHOB(5);

fgcolor("black");
moveto(xpmax-275,40); write_string("WELL DESIGN - DTCO EXTRAPOLATION");
moveto(xpmax-275,55); write_string("SCALE: DTCO: us/ft");
moveto(xpmax-275,70); write_string("          DENSITY: g/cm3");
}

void trend_curve_drawing(double xpmax, double ypmax, double depth_new, double
delta_t0_new, double depth_old, double delta_t0_old, double delta_tn_new, double
delta_tn_old, int final_depth, int data, int index, int ind,int cor, int par)
{

    double depth_pixel_new = 20 + depth_new*(ypmax-40)/(final_depth);
    double delta_t0_pixel_new = 130+log10(delta_t0_new/10)*(xpmax-150)/2;
    double depth_pixel_old = 20 + depth_old*(ypmax-40)/(final_depth);
    double delta_t0_pixel_old = 130+log10(delta_t0_old/10)*(xpmax-150)/2;

    double delta_tn_pixel_new = 130+log10(delta_tn_new/10)*(xpmax-150)/2;
    double delta_tn_pixel_old = 130+log10(delta_tn_old/10)*(xpmax-150)/2;

    pensize(2.5);
    if (ind==0)
    fgcolor("blue");

    if (cor == 1)
    fgcolor("red");
    if (cor == 2)
    fgcolor("magenta");
    if (cor == 3)
    fgcolor("orange");

    moveto(delta_t0_pixel_old,depth_pixel_old);
    fgcolor("blue");
    lineto(delta_t0_pixel_new,depth_pixel_new);

    if (par==0)
    {
        moveto(delta_t0_pixel_new,depth_pixel_new);
        fgcolor("black"); bgcolor("black"); ellipse(4,4);
    }

    if (index==1)
    {
        fgcolor ("red"); pensize(4);
        moveto(delta_tn_pixel_old,depth_pixel_old);

```

```

        lineto(delta_tn_pixel_new,depth_pixel_new);
    }
}

void RHOB_drawing(double xpmx, double ypmx, double depth_new, double RHOB_new,
double depth_old, double RHOB_old, int final_depth, int data, int ind)
{
    double depth_pixel_new = 20 + depth_new*(ypmx-40)/(final_depth);
    double RHOB_pixel_new = 30+(RHOB_new-1.7)*(xpmx-50);
    double depth_pixel_old = 20 + depth_old*(ypmx-40)/(final_depth);
    double RHOB_pixel_old = 30+(RHOB_old-1.7)*(xpmx-50);

    if (ind==0) {
        pensize(4);
        fgcolor("darkblue");
        bgcolor("darkblue"); }

    if (ind==1) {
        pensize(4);
        fgcolor("red");
        bgcolor("red"); }

    moveto(RHOB_pixel_old,depth_pixel_old);
    lineto(RHOB_pixel_new,depth_pixel_new);
}

void RHOB_ext_drawing(double xpmx, double ypmx, double depth_new, double RHOB_new,
double depth_old, double RHOB_old, int final_depth, int data, int ind)
{
    double depth_pixel_new = 20 + depth_new*(ypmx-40)/(final_depth);
    double RHOB_pixel_new = 30+(RHOB_new-1.7)*(xpmx-50);
    double depth_pixel_old = 20 + depth_old*(ypmx-40)/(final_depth);
    double RHOB_pixel_old = 30+(RHOB_old-1.7)*(xpmx-50);

    if (ind==0) {
        pensize(4);
        fgcolor("green"); }

    if (ind==1) {
        pensize(4);
        fgcolor("black"); }

    moveto(RHOB_pixel_old,depth_pixel_old);
    lineto(RHOB_pixel_new,depth_pixel_new);
}

void DTCO_ext_drawing(double xpmx, double ypmx, double RHOB, double DTCO, double
RHOB_old, double DTCO_old, int final_depth, int data, int ind, int ind2)
{
    double DTCO_pixel = 20 + (280-DTCO)*(ypmx-40)/240;
    double RHOB_pixel = 30 + (RHOB-1.7)*(xpmx-50);
    double DTCO_pixel_old = 20 + (280-DTCO_old)*(ypmx-40)/240;
    double RHOB_pixel_old = 30 + (RHOB_old-1.7)*(xpmx-50);

    pensize(4);

    if (ind==0)
    {
        fgcolor ("magenta");
        bgcolor("magenta");

```

```

moveto(RHOB_pixel,DTCO_pixel);
ellipse(4,4);
}

else
{
fgcolor("black");

moveto(RHOB_pixel_old,DTCO_pixel_old);
lineto(RHOB_pixel,DTCO_pixel);
}
}

double reg_beta(double K1, double K2, double K3, double Ln1, double Ln2, double Ln3)
{
double K_med, Ln_med, beta_num, beta_den, beta;

K_med = (K1+K2+K3)/3;
Ln_med = (Ln1 + Ln2 + Ln3)/3;

beta_num = (3*(K1*Ln1+K2*Ln2+K3*Ln3)-(K1+K2+K3)*(Ln1+Ln2+Ln3));
beta_den = (3*(pow(Ln1,2)+pow(Ln2,2)+pow(Ln3,2))-pow((Ln1+Ln2+Ln3),2));
beta = beta_num/beta_den;
return beta;
}

double reg_alfa (double beta, double K1, double K2, double K3, double Ln1, double Ln2,
double Ln3)
{
double K_med, Ln_med, alfa;

K_med = (K1+K2+K3)/3;
Ln_med = (Ln1 + Ln2 + Ln3)/3;

alfa = K_med - beta*Ln_med;
return alfa;
}

double cemento (double WOH, double OD, double thick, double depth, double spe_weight,
double density, double ODF, double depth_old, double OD_old, double d_cim, double
d_aco, int conf)
{
double cim_depth, A, P_cim, P_gancho, A_an1, A_an2, d, ID, V, OD1, d1, d2,
d_ret;

OD1 = OD*0.0254;
ID = (OD-2*thick)*0.0254;
A = 0.25*pi*pow(ID,2);
V = A*depth;
P_cim = (d_cim*A*depth)*265;
A_an1 = 0.25*pi*(pow(ODF*0.0254,2)-pow(OD1,2));
A_an2 = 0.25*pi*(pow(OD_old*0.0254,2)-pow(OD1,2));
P_gancho = spe_weight*0.4545*(depth*3.28)*(1-density/d_aco)+P_cim;
d1 = V/A_an1;

if (conf == 1) {

if (d1 > depth-depth_old) {d1 = (depth-depth_old); d2 = (V-A_an1*d1)/A_an2;
d_ret = d1+d2;}
else d_ret=d1; }

else d_ret = d1;
}

```

```

        if (P_gancho >= WOH) return cemento(WOH, OD, thick, depth-10, spe_weight,
density, ODF, depth_old, OD_old, d_cim, d_aco,conf);
        else if (d_ret > depth) return depth;
        else return d_ret;
    }

double cemento_2 (double depth_tot, double depth_cim, double d_cim, double Gf, double
LDA)
{
    double Pf, Pcim;

    Pf = 0.1705*Gf*(depth_tot+LDA);
    Pcim = 0.1705*d_cim*depth_cim + 0.1705*8.5*(depth_tot-d_cim);

    if (Pcim > Pf) return cemento_2(depth_tot,depth_cim-10,d_cim,Gf,LDA);
    else return depth_cim;
}

double BOP_PP(double Gpmax, double Dh, double Dbop, double coef, double d_gas)
{
    double P_BOP, P_gas;

    P_gas = coef*(Dh-Dbop)*d_gas;
    P_BOP = coef*Dh*Gpmax - P_gas;
    return P_BOP;
}

double friction_factor(double Re, double f_guess, double prec ) //function to return
the converged value of friction_fator given by
{
    // newton-raphson
method application
    double f, error, abs_error, func, func_der;

    func = 4*log(Re*sqrt(f_guess)) - 1/sqrt(f_guess) - 0.395;
    func_der = -2/(Re*pow(f_guess,2)) + 1/(2*f_guess*sqrt(f_guess));
    f = f_guess - func/func_der;
    error = f-f_guess;
    abs_error = abs(error);
    if (abs_error <= prec) return f;
    else return friction_factor(Re,f,prec);
}

double rho_circ(double q, double density, double d_stage, double OD, double visc,
double depth) //function to calculate the bottom hole
{
// pressure (dynamic pressure)
    double vel, Re, charge_loss, p_bh, f, rho_dinamico, Deq;

    Deq = 0.816*(d_stage-OD);
    vel = q/(2.448*(pow(d_stage,2)-pow(OD,2))); // mean velocity calculation to be
used in the other parameters
    Re = 928*density*vel*Deq/visc; //Reynolds Number calculation
    if (Re<2100) f = 16/Re;
    else f = friction_factor(Re,0.001,0.00001); //if turbulent flow - friction
factor needed

    charge_loss = (f*density*pow(vel,2)/(25.8*Deq))*(depth);

    p_bh = 0.17*density*depth + charge_loss; //final bottom hole pressure
    rho_dinamico = p_bh/(0.17*depth);
    return rho_dinamico;
}

```



```

double BOP_LDA(double Gpmax, double Dh, double Dbop, double coef, double d_gas, double
Dw, double grad_water)
{
    double P_BOP, P_gas, P_LDA;

    P_LDA = coef*Dw*grad_water;
    P_gas = coef*(Dh - Dbop)*d_gas;
    P_BOP = coef*Dh*Gpmax - P_LDA - P_gas;
    return P_BOP;
}

double BOP_PF(double Gf, double Dcg, double Dbop, double coef, double d_gas)
{
    double P_BOP, P_gas;

    P_gas = coef*(Dcg-Dbop)*d_gas;
    P_BOP = coef*Dcg*Gf - P_gas;
    return P_BOP;
}

double BOP_CR(double sigma, double OD, double thick)
{
    double P_BOP, P_cg, A, ID;

    ID = OD - 2*thick;
    P_cg = 0.875*(2*thick*sigma)/ID;
    P_BOP = P_cg;
    return P_BOP;
}

double BOP(double PBOP_PP, double PBOP_LDA, double PBOP_PF, double PBOP_CR)
{
    double PBOP_norm, PBOP=0;

    if (PBOP_PP > PBOP) PBOP = PBOP_PP;
    if (PBOP_LDA > PBOP) PBOP = PBOP_LDA;
    if (PBOP_PF > PBOP) PBOP = PBOP_PF;
    if (PBOP_CR > PBOP) PBOP = PBOP_CR;

    if (PBOP < 5000) PBOP_norm = 5000;
    else if (PBOP < 10000) PBOP_norm = 10000;
    else if (PBOP < 15000) PBOP_norm = 15000;
    else if (PBOP < 20000) PBOP_norm = 20000;
    else PBOP_norm = 0;

    return PBOP_norm;
}

void desenho_linha()
{
    int i=0;
    while (i<70)
    { cout << "*"; i=i+1; }
    cout << endl;
}

int tela_inicial()
{
    int opcao;

    desenho_linha();
    cout << "*"          MODULO DE PLANEJAMENTO DE POCO - UFRJ DRILLING SIMULATOR" <<
endl;
}

```

```

cout << "*" << endl;
cout << "*"      MENU INICIAL:"<< endl;
cout << "*" << endl;
cout << "*" << endl;
cout << "*"      PRESS 1) DEFINICAO DE PARAMETROS" << endl;
cout << "*"      PRESS 2) INTERPOLACAO DOS DADOS" << endl;
cout << "*"      PRESS 3) DEFINICAO DAS DENSIDADES" << endl;
cout << "*"      PRESS 4) EXTRAPOLACAO DAS DENSIDADES" << endl;
cout << "*"      PRESS 5) DEFINICAO DO GRADIENTE DE POROS E SOBRECARGA" << endl;
cout << "*"      PRESS 6) DEFINICAO DO GRADIENTE DE FRATURA" << endl;
cout << "*"      PRESS 7) PROJETO DE POCO - CRITERIO DE GRADIENTE" << endl;
cout << "*"      PRESS 8) PROJETO DE POCO - CRITERIO DE MARGEM DE RISER" << endl;
cout << "*" << endl;
cout << "*"      OPCA0: "; cin >> opcao;
desenho_linha;

return opcao;
}

int tela_OD()
{
int opcao1;

desenho_linha();

cout << "*"      DETERMINE O PADRAO DE DIAMETROS" << endl;
cout << "*" << endl;
cout << "*"      PRESS 1) DEFAULT DO PROGRAMA" << endl;
cout << "*"      PRESS 2) INSERIR MANUALMENTE" << endl;
cout << "*" << endl;
cout << "*"      OPCA0: "; cin >> opcao1;
desenho_linha;

return opcao1;
}

int tela_GP()
{
int opcao2;

desenho_linha();

cout << "*"      DETERMINE A ENTRADA DOS GRADIENTES DE POROS" << endl;
cout << "*" << endl;
cout << "*"      PRESS 1) MODELOS GEOMECANICOS" << endl;
cout << "*"      PRESS 2) INSERIR POR ARQUIVO .TXT" << endl;
cout << "*" << endl;
cout << "*"      OPCA0: "; cin >> opcao2;
desenho_linha;

return opcao2;
}

int tela_PP()
{
int opcao3;

desenho_linha();

cout << "*"      DETERMINE O METODO DE DETERMINACAO DOS PONTOS DA TRENDLINE" <<
endl;
cout << "*" << endl;

```

```

    cout << "*"          PRESS 1) CALIBRACAO TRECHO SUPERFICIAL - MINIMIZACAO DA
VARIANCIA" << endl;
    cout << "*"          PRESS 2) CALIBRACAO COM RFT - MINIMIZACAO DA VARIANCIA" << endl;
    cout << "*"          PRESS 3) INSERIR MANUALMENTE" << endl;
    cout << "*" << endl;
    cout << "*"          OPCAO: "; cin >> opcao3;
desenho_linha;

    return opcao3;
}

int tela_MG()
{
    int opcao4;

    cout << "*"          DETERMINE O MODELO GEOMECANICO DESEJADO" << endl;
    cout << "*" << endl;
    cout << "*"          PRESS 1) MODELO DE EATON" << endl;
    cout << "*"          PRESS 2) MODELO DE BOWERS" << endl;
    cout << "*" << endl;
    cout << "*"          OPCAO: "; cin >> opcao4;
desenho_linha;

    return opcao4;
}

int tela_RHOB()
{
    int opcao5;

    cout << "*"          CALIBRACAO DE DADOS DO RHOB - MODELO DE GARDNER" << endl;
    cout << "*" << endl;
    cout << "*"          PRESS 1) MODELO JA ESTA CALIBRADO" << endl;
    cout << "*"          PRESS 2) MODELO NECESSITA DE CALIBRACAO" << endl;
    cout << "*" << endl;
    cout << "*"          OPCAO: "; cin >> opcao5;
desenho_linha;

    return opcao5;
}

int tela_MediaGp()
{
    int opcao6;

    cout << "*"          DEFINICAO DO METODO DE SUAUIZACAO DO Gp" << endl;
    cout << "*" << endl;
    cout << "*"          PRESS 1) MEDIA ARITIMETICA" << endl;
    cout << "*"          PRESS 2) MEDIA PONDERADA" << endl;
    cout << "*" << endl;
    cout << "*"          OPCAO: "; cin >> opcao6;
desenho_linha;

    return opcao6;
}

int tela_RHOB_EXT()
{
    int opcao7;

    cout << "*"          CALIBRACAO DA CURVA EXTRAPOLADA" << endl;
    cout << "*" << endl;
    cout << "*"          PRESS 1) CALIBRACAO - MINIMIZACAO DA VARIANCIA" << endl;

```

```

    cout << "*"          PRESS 2) INSERIR MANUALMENTE" << endl;
    cout << "*" << endl;
    cout << "*"          OPCAO: "; cin >> opcao7;
    desenho_linha;

    return opcao7;
}

int tela_RHOB_MAIN()
{
    int opcao8;

    desenho_linha();

    cout << "*"          DETERMINE A ENTRADA DAS DENSIDADES" << endl;
    cout << "*" << endl;
    cout << "*"          PRESS 1) METODO DE GARDNER" << endl;
    cout << "*"          PRESS 2) INSERIR POR ARQUIVO .TXT" << endl;
    cout << "*" << endl;
    cout << "*"          OPCAO: "; cin >> opcao8;
    desenho_linha;

    return opcao8;
}

int tela_BOWERS_OPTION()
{
    int opcao9;

    desenho_linha();

    cout << "*"          COMO DESEJA DETERMINAR OS PARAMETROS DE CALIBRAÇÃO?" << endl;
    cout << "*" << endl;
    cout << "*"          PRESS 1) ENTRADA DIRETA" << endl;
    cout << "*"          PRESS 2) MINIMIZACAO DA VARIANCIA" << endl;
    cout << "*" << endl;
    cout << "*"          OPCAO: "; cin >> opcao9;
    desenho_linha;

    return opcao9;
}

int tela_MAIN()
{
    int opcao_main;

    LOGO_design();
    cout << endl;

    desenho_linha();

    cout << "*" << endl;

    cout << "*"          DETERMINE O MODULO DA SIMULACAO" << endl;
    cout << "*" << endl;
    cout << "*"          PRESS 1) PLANEJAMENTO DE POCO" << endl;
    cout << "*"          PRESS 2) SIMULADOR DINAMICO REAL-TIME" << endl;
    cout << "*" << endl;
    cout << "*"          OPCAO: "; cin >> opcao_main;

    return opcao_main;
}

```

```

//DECLARATION OF MODULO 2 FUNCTIONS

double friction_factor_newt(double Re, double f_guess, double prec )
{
    double f, error, abs_error, func, func_der;

    func = 4*log10(Re*sqrt(f_guess)) - 1/sqrt(f_guess) - 0.395;
    func_der = -2/(Re*pow(f_guess,2)) + 1/(2*f_guess*sqrt(f_guess));
    f = f_guess - func/func_der;
    error = f-f_guess;
    abs_error = abs(error);
    if (abs_error <= prec) return f;
    else return friction_factor_newt(Re,f,prec);
}

double newtonian_pipe(double q, double density, double d_int, double visc) //conferido
{
    double vel, Re, charge_loss, dP, f, A;

    A = 2.448*(pow(d_int,2));
    vel = q/A;
    Re = 968*density*vel*(d_int)/visc;
    if (Re<2100) charge_loss = visc*vel/(1500*pow(d_int,2));
    else { f = friction_factor_newt(Re,0.001,0.00001);
          charge_loss = f*density*pow(vel,2)/(25.8*d_int);}
    dP = charge_loss;

    return dP;
}

double newtonian_anulus(double q, double density, double d_ext, double d_int, double
visc) //conferido
{
    double vel, Re, charge_loss, Pf, f, d_eq, d_eq_tur, A, dP;

    d_eq = 0.816*(d_ext-d_int);
    d_eq_tur = sqrt(pow(d_ext,2)+pow(d_int,2)-(pow(d_ext,2)-
pow(d_int,2))/log(d_ext/d_int));
    A = 2.448*(pow(d_ext,2)-pow(d_int,2));
    vel = q/A;
    Re = 928*density*vel*d_eq/visc;
    if (Re<2100) charge_loss = visc*vel/(1500*pow(d_eq,2));
    else { f = friction_factor_newt(Re,0.001,0.00001);
          charge_loss = f*density*pow(vel,2)/(25.8*d_eq_tur);}
    dP = charge_loss;
    return dP;
}

double Rec_power(double n)
{
    double Rec;

    Rec = -9188.5*n + 5947.4;

    if (Rec > 2100) return 2100;
    else return Rec;
}

double friction_factor_power(double Re, double f_guess, double prec, double n)
{
    double f, error, abs_error, func, func_der;
    ofstream output;

```

```

    func = 4.0/pow(n,0.75)*log10(Re*pow(f_guess,1-0.5*n)) - 1/sqrt(f_guess) -
0.395/pow(n,1.2);
    func_der = 0.5*pow(1/f_guess,1.5)+4.0*(1-0.5*n)/(f_guess*pow(n,0.75)*log(10));
    f = f_guess - func/func_der;
    error = f-f_guess;
    abs_error = abs(error);
    if (abs_error <= prec) return f;
    else return friction_factor_power(Re,f,prec,n);
}

double power_anulus(double q, double density, double d_ext, double d_int, double K,
double n)
{
    double vel, Re, He, razao, f, charge_loss, Re_c, dP, d_eq_tur, d_eq, visc, A;

    d_eq = 0.816*(d_ext-d_int);
    A = 2.448*(pow(d_ext,2)-pow(d_int,2));
    vel = q/A;

    Re = (1/K)*109000*density*pow(vel,2-n)*pow(0.02549*d_eq/(2+1/n),n);

    Re_c = Rec_power(n);

    if (Re <= Re_c)
    {
        charge_loss = K*pow(vel,n)*pow(((2+1/n)/0.0208),n)/(144000*pow(d_ext-
d_int,1+n),n);
    }
    else if (Re > Re_c)
    {
        f = friction_factor_power(Re,0.001,0.0001,n);
        charge_loss = f*density*pow(vel,2)/(25.8*d_eq);
    }

    dP = charge_loss;
    return dP;
}

double power_pipe(double q, double density, double d_int, double K, double n)
{
    double vel, Re, He, razao, f, charge_loss, Re_c, A, dP, visc;

    A = 2.448*(pow(d_int,2));
    vel = q/A;
    visc = K*pow(d_int,1-n)*pow((3+1/n)/0.0416,n)/(96*pow(vel,1-n));

    Re = (1/K)*89100*density*pow(vel,2-n)*pow(0.0416*d_int/(3+1/n),n);

    Re_c = Rec_power(n);

    if (Re <= Re_c)
    {
        charge_loss =
K*pow(vel,n)*pow(((3+1/n)/0.0416),n)/(144000*pow(d_int,1+n));
    }
    else if (Re > Re_c)
    {
        f = friction_factor_power(Re,0.001,0.0001,n);
        charge_loss = f*density*pow(vel,2)/(25.8*d_int);
    }

    dP = charge_loss;
    return dP;
}

```

```

    }

double razao_bingham (double He, double razao_guess, double prec)
{
    double func, func_der, abs_error, razao, error, C;

    C = He/16800;
    func = C*pow(1-razao_guess,3)-razao_guess;
    func_der = -3*C*pow(1-razao_guess,2)-1;
    razao = razao_guess - func/func_der;
    error = razao-razao_guess;
    abs_error = abs(error);
    if (abs_error <= prec) return razao;
    else return razao_bingham(He,razao,prec);
}

double bingham_anulus(double q, double density, double d_ext, double d_int, double
T_0, double p_visc)
{
    double vel, Re, He, razao, f, charge_loss, Re_c, dP, d_eq, d_eq_tur, A;

    d_eq = 0.816*(d_ext-d_int);
    A = 2.448*(pow(d_ext,2)-pow(d_int,2));
    vel = q/A;
    Re = 928*density*vel*d_eq/p_visc;
    He = 37100*density*T_0*pow(d_eq,2)/pow(p_visc,2);
    razao = razao_bingham(He,1,0.01);
    Re_c = (1-(4/3)*razao+(1/3)*pow(razao,4))*He/(8*razao);

    if (Re <= Re_c)
    {
        charge_loss = p_visc*vel/(1000*pow((d_ext-d_int),2)) + T_0/(200*(d_ext-
d_int));
    }
    else if (Re > Re_c)
    {
        f = 0.0791/pow(Re,0.25);
        charge_loss = f*density*pow(vel,2)/(25.8*d_eq);
        //charge_loss =
(pow(density,0.75)*pow(p_visc,0.25)*pow(vel,1.75))/(1.396*(pow((d_ext-d_int),1.25)));
    }

    dP = charge_loss;
    return dP;
}

double bingham_pipe(double q, double density, double d_int, double T_0, double p_visc)
{
    double vel, Re, dP, He, razao, f, charge_loss, Re_c, A;

    A = 2.448*(pow(d_int,2));
    vel = q/A;
    Re = 928*density*vel*(d_int)/p_visc;
    He = 37100*density*T_0*pow(d_int,2)/pow(p_visc,2);
    razao = razao_bingham(He,1,0.01);
    Re_c = (1-(4/3)*razao+(1/3)*pow(razao,4))*He/(8*razao);

    if (Re <= Re_c)
    {
        charge_loss = p_visc*vel/(1500*pow(d_int,2)) + T_0/(225*d_int);
    }
    else if (Re > Re_c)

```

```

    {
        f = 0.0791/pow(Re,0.25);
        charge_loss = f*density*pow(vel,2)/(25.8*d_int);
        //charge_loss =
(pow(density,0.75)*pow(p_visc,0.25)*pow(vel,1.75))/(1.396*(pow(d_int,1.25)));
    }

    dP = charge_loss;
    return dP;
}

//funcoes para plotagem no simulador

//plot 1 - tempo atual p/ tempo previsto (tempo x depth)
void gridding_TIME_simul(double final_depth, double xpmax, double ypmax, double
time_max)
{
    double x_pixel, y_pixel, x, y=10, x_dif, y_pixel_0, y_dif, C, C2;
    int i=0, j=0, f;

    fgcolor("gray");
    pensize(1);

    if (final_depth < 2500) { y_dif = 250; f=1; }
    else if (final_depth < 5000) { y_dif = 500; f=2; }
    else { y_dif = 750; f=3; }

    x_dif = 10;
    x = 0;
    y_pixel_0 = 20 + y*(ypmax-40)/(final_depth);

    while (y < final_depth)
    {
        fgcolor("gray");
        y_pixel = 20 + y*(ypmax-40)/(final_depth);
        moveto(30,y_pixel); lineto(xpmax-20,y_pixel);
        moveto(5,y_pixel+5); scale_y(f,j); j=j+1;
        y = y + y_dif;
    }

    fgcolor("gray");
    i=0;
    C = time_max/24;

    while (x < time_max)
    {
        x_pixel = 130 + x*(xpmax-150)/C;
        moveto(x_pixel,y_pixel_0); lineto(x_pixel,ypmax-20);
        moveto(x_pixel-5,y_pixel_0-5); scale_time_x(i); i=i+1;
        x = x + x_dif;
    }

    fgcolor("black");
    moveto(xpmax-250,40); write_string("WELL PLANING - TIME ESTIMATION");
    moveto(xpmax-250,55); write_string("SCALE: TVDBML:    m");
    moveto(xpmax-250,70); write_string("          AC. TIME:  Days");
    moveto(xpmax-250,85); write_string("          ROP:      m/h");
}

void time_drawing_simul(double xpmax, double ypmax, double time_simul, double
time_simul_ant, double depth, double depth_ant, double final_depth, double time_max,
int type)
{

```



```

double time_simul_pixel, time_simul_pixel_ant, depth_pixel_ant, depth_pixel, C;

C = time_max/24+2;
time_simul_pixel = 130 + (time_simul/24)*(xpmax-150)/C;
time_simul_pixel_ant = 130 + (time_simul_ant/24)*(xpmax-150)/C;

depth_pixel = 20 + (depth)*(ypmax-40)/(final_depth);
depth_pixel_ant = 20 + (depth_ant)*(ypmax-40)/(final_depth);

pensize(3);
if (type == 1) fgcolor("darkblue");
if (type == 2) fgcolor("red");
moveto(time_simul_pixel_ant,depth_pixel_ant);
lineto(time_simul_pixel,depth_pixel);
}

int tela_simulacao()
{
int opcao;

desenho_linha();
cout << "*"          REAL-TIME DRILLING SIMULATOR PROJECT - UFRJ" << endl;
cout << "*" << endl;
cout << "*"          MENU INICIAL:"<< endl;
cout << "*" << endl;
cout << "*" << endl;
cout << "*"          PRESS 1) DEFINIR PARAMETROS DE FLUIDO" << endl;
cout << "*"          PRESS 2) DEFINIR PARAMETROS DOS TUBOS" << endl;
cout << "*"          PRESS 3) DEFINIR PARAMETROS DA BROCA" << endl;
cout << "*"          PRESS 4) INICIAR PROCESSO DE ASSENTAMENTO DE SAPATA" << endl;
cout << "*"          PRESS 5) COMPLETAR O POÇO COM LAMA DE PERFURACAO" << endl;
cout << "*"          PRESS 6) INICIAR PROCESSO DE CONTROLE DE POÇO" << endl;
cout << "*"          PRESS 7) RETORNAR A SIMULACAO" << endl << endl;
cout << "*"          PRESS 8) INICIAR SIMULACAO" << endl;
cout << "*" << endl;
cout << "*"          OPCAO: "; cin >> opcao;

return opcao;
}

int main ()
{
//DECLARAÇÃO DE VARIÁVEIS PARA MÓDULO PROJETO DE POÇO
double *depth, *Gp, *Gf, *Gfmarg, d_ext[10], d_stage[10], thick[10], column,
kick_tol, dif_RFT_min=10, Dh, pond_coef[5];
double density[10], *Gpmarg, depth_old[20], depth_new[20], q, d_ext1, thick1,
d_stage1, ODrev[10][30], ODest[10][30];
double grad_water=8.5, xpmax=1024, ypmax=768, xpmax2=400, ypmax2=500, error,
cond_depth, qnt_baritina[15], coef, s1, WOH, ac_rho=0, *Gpm, abras[2];
int i=0, n, N=0, y=0, b=0, lim_tension[15], sigma[15], c=0, count, criteria, j=0,
final_depth, data, input, opcao, opcao1, cg[20],LDA,interp,R;
int Df1, Df2, Df3, j1, j2, j3, l, cond_sec, f1, f2, RFTp1, RFTp2, g_norm1,
g_norm2, srt, norm_lim1, norm_lim2, s, v, x1, x2, u=0, opcao5,interp0=0;
int
data1,opcao6,opcao7,comp_sup,data2,opcao2,k=0,opcao3,opcao4,cg_final,opcao8,RHOB_index
,extrap,m,cor,kick_index,*lito,a, BRL, bit[50], Time_Model;
string opt, teste, c_tipo[20],opt2, nome;
double *Gov, *sigma_ov, *delta_t0, *delta_tn, *rho_ov, K1, K2, K3, p_fracture,
p_bh, p_porous, dif_RFT_2;
double m_inc, coef_Eaton, G_LOT1, G_LOT2, G_LOT3, D0, D1, delta_t0p, delta_t1p,
coef_seg, Pp1, Pp2, Pp3, dif_RFT_1;

```

```

    double alfa, beta, *K, cim_prof[10], cim_prof_final, Ln1, Ln2, Ln3, Gpmarg_max,
    WOH_m, RFTd1, RFTd2, dif_RFT, d_gas, PBOP_PF_max=0;
    double well_obj, sem_reg, air_gap, a_ind, b_ind, delta_D, WPF[20], d_aco, d_cim,
    Gp_max, PBOP_PP, PBOP_LDA, PBOP_PF[5], PBOP_CR[5], PBOP_CR_max=0;
    double depth_ant[20], z_inicial[20], z_inicial_final_1, z_inicial_final_2,
    depth_ant1[20], *RHOB, *depth_RHO, well_obj1, dif_RFT_teste, pond_coef_sum=0;
    double RHOB_ext, a_ext, b_ext, c_ext, a_ext_min, b_ext_min, c_ext_min, variancia,
    dev_pad, dev_pad_min=1000, RHOB_erro, t_trig_sum[20];
    double
    DTCO_erro, depth_final_i, depth_final_2, DTCO_ext, DTCO_error, cim_prof_2[20], cim_prof_fina
    l_2, *delta_tn_2, t_t;
    double Hdc, ODDc, Cdc, Hhwdp, ODhwdp, Chwdp, Hcom, ODcom, Ccom, Hdp, ODdp, Cdp,
    kick_dif_min, kick_v, depth_casing[15], kick_h, kick_tolerance[15];
    double Vo, A, B, U,
    delta_t0_min=300, value[60], ODprod, *MR, a_borg[2][10], F1, F2, F3, F4, F5, F6, F7, F8,
    *t_sum_plot, Kbite, tb[10], tb_Dlim[10];
    double TVD_ft, visc, Cd, WOB_t, *ROP, t_trig[15], ts, ls, *t, t_stage[20], *t_sum,
    depth_trig[100], t_casing[20], t_cas_plac;
    double API[15], thick_teste, *Di, *Dfim, *T, *Tac, *H1, *H2, *WOB_d_m, *WOB,
    *Freq, *jato1, *jato2, *jato3;
    double Dil[100], Dfim1[100], T1[100], Tac1[100], H11[100], H21[100],
    WOB_d_m1[100], WOB1[100], Freq1[100], jato11[100], jato21[100], jato31[100];
    double rho_lama, d_rev, d_bite, At_bites, P_bites, Fj, rho_dinam, ROPmax=0, J2,
    hear, t_tooth[100], bit_depth[100], t_trig_hear[100];
    double t_drilling=0, t_manobra=0, *mud_weight, *Gf_kick_corr, Gf_kick, p_kick_tol,
    RHOB_ext_ant, DTCO_ext_ant, WOBd_t[2];
    int M, P, hear_cont, r, well_obj_data, *B_par, opcao_main, opcao9;
    double depth_cas, beta2_num, beta2_den, beta2, alfa2, t_desl, abrasm, ODriser;

    cout << "Defina o nome do poço: "; cin >> nome;
    clrscr;

    MAIN:
    clrscr;

    opcao_main = tela_MAIN();

    if (opcao_main==1)
    {
    M:
    clrscr;
    opcao = tela_inicial();

    ifstream param_input, input_0, Gov_input, Gp_input, Gf_input, Gf_fin,
    G_total_input, Gp1_input, Gpm_input, ODCasing, Gp_externo,
    RHOB_input, DTCO_interp, DTCO_calib, Gov_input2, Borgoyne_input, cfixo, cvar;
    ofstream Gov_output, Gp_output, Gf_output, G_total, Gpm_output, Code_output,
    RFT_output, DTCO_interp_out, Gov_output2, RHOB_teste, DTCO_conf;

    param_input.open("C:/UFRJDrillingSimulator/Inputs/InputMain.txt");
    input_0.open("C:/UFRJDrillingSimulator/Inputs/InputDT.txt");
    Gov_output.open("C:/UFRJDrillingSimulator/Debug/DebugGOV.txt");

    if (opcao == 1)
    {
    clrscr;

    i=0;
    param_input >> R;

    while (i<R)
    {
        param_input >> value[i];

```

```

        i=i+1;
    }

    data1 = value[0];          //numero de dados
    well_obj1 = value[1];     //objetivo do poço
    LDA = value[2];          //lamina água
    sem_reg = value[3];      //zona sem registro de sismica
    cond_sec = value[4];     //numero de secoes do condutor
    comp_sup = value[5];     //comprimento do rev de superficie
    WOH_m = value[6];        //peso sobre o gancho (maximo)
    d_aço = value[7];        //densidade do aco
    d_cim = value[8];        //densidade do cimento
    d_gas = value[9];        //densidade do gas
    error = value[10];       //gap de seguranca no gradiente
    coef_Eaton = value[11];  //coef de Eaton
    Vo = value[12];          //Vo de Bowers
    A = value[13];           //coef A de Bowers
    B = value[14];           //coef B de Bowers
    U = value[15];           //coef U de Bowers
    RFTp1 = value[16];       //Repeted Formation test Prof1
    RFTd1 = value[17];       //Repeted Formation test Grad1
    RFTp2 = value[18];       //Repeted Formation test Prof2
    RFTd2 = value[19];       //Repeted Formation test Grad2
    Df1 = value[20];         //Prof1 LOT
    G_LOT1 = value[21];      //Grad1 LOT
    Df2 = value[22];         //Prof2 LOT
    G_LOT2 = value[23];      //Grad2 LOT
    Df3 = value[24];         //Prof3 LOT
    G_LOT3 = value[25];      //Grad3 LOT
    ODprod = value[26];      //OD rev de producao
    q = value[27];           //vazao para calculo de Kick Tolerance
    visc = value[28];        //visc para calculo de kick tolerance
    Cd = value[29];          //Cd para calculo de pressao hidraulica na broca
    WOB_t = value[30];       //peso sobre a broca
    ts = value[31];          //tempo para substituir cada pipe
    ls = value[32];          //comprimento de cada pipe
    abrasm = value[33];      //valor medio da abrasividade
    ODriser = value[34];     //OD riser

    WOH = 1000000*WOH_m;    //redimsnionando o peso max sobre gancho
    cond_depth = 12*cond_sec; //prof do condutor
    srt = LDA+sem_reg;      //coordenada da zona sem registro sismico
    well_obj=well_obj1-LDA;  //distancia absoluta em TVDBML
    well_obj_data = 6000;    //total de dados discretizados (1 em 1 m)
    final_depth = well_obj-sem_reg;
    data = data1;
    coef = 0.17;            //coef para conv de unidade (prof em m)
    param_input.close();

    cout << "CARREGAMENTO DOS DADOS INICIAIS DO POCO " << nome << endl << endl << endl
    << endl;
    cout << "CARREGANDO "; Sleep(300); cout << ">"; Sleep(300); cout << ">";
    Sleep(300); cout << ">"; Sleep(300); cout << ">";
    cout << " CARREGADO!" << endl << endl;

    cout << endl << endl;
    cout << "REPETIR A OPERACAO?      "; cin >> opt;
    if (opt == "NAO") goto M;
}

if (opcao==2)
{

```

```

DTCO_interp.open("C:/UFRJDrillingSimulator/Inputs/InputDT.txt");

clrscr;
interpol = 1;
i=sem_reg;

//Alocate memory for this variables - considering 1m step
B_par = new int[well_obj_data];

depth = new double[well_obj_data];
delta_t0 = new double[well_obj_data];
lito = new int[well_obj_data];
while (i<=well_obj)
{
DTCO_interp >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];
i=i+final_depth/data;
}

DTCO_interp_out.open("C:/UFRJDrillingSimulator/Debug/DebugDTInterpol.txt");

cout << "DEFINICAO DA INTERPOLACAO DOS DADOS SISMICOS DO POCO " << nome << endl <<
endl << endl;
cout << "INTERVALO DE INTERPOLACAO (m): "; cin >> interpol;

i=sem_reg+interpol; j=i-interpol;

DTCO_interp_out << depth[j] << " " << delta_t0[j] << " " << lito[j] << " "
<< B_par[i] << endl;

while (i<=well_obj)
{
delta_t0[i] = delta_t0[j] + (((i+LDA)-
depth[j])*(delta_t0[j+(final_depth/data)]-delta_t0[j])) /
(depth[j+(final_depth/data)]-depth[j]); depth[i] = i+LDA;
lito[i] = lito[j];
B_par[i] = B_par[j];
DTCO_interp_out << depth[i] << " " << delta_t0[i] << " " << lito[i] <<
" " << B_par[i] << endl;

i=i+interpol; if (i==j+final_depth/data) {j=j+final_depth/data;}
}

DTCO_interp.close();
DTCO_interp_out.close();

data = final_depth/interpol;
//cout << endl;
cout << "INTERPOLANDO "; Sleep(300); cout << ">"; Sleep(300); cout << ">";
Sleep(300); cout << ">"; Sleep(300); cout << ">";
cout << " INTERPOLADO!" << endl << endl;
Gf_fin.close(); G_total.close();

DTCO_calib.open("C:/UFRJDrillingSimulator/Debug/DebugDTInterpol.txt");

cout << endl << "REPETIR A OPERACAO? "; cin >> opt;
if (opt == "NAO") goto M;
}

if (opcao==3)
{
clrscr;
RHO_MED:

```

```

opcao8 = tela_RHOB_MAIN();

//Allocate memory for rho_overburden
rho_ov = new double[well_obj_data];

if (opcao8==1)
{
RHOB_direto:
clrscr;
opcao5 = tela_RHOB();

if (opcao5==1) {clrscr;
cout << "METODO DE GARDNER - coef. a: "; cin >> a_ind;
cout << " - coef. b: "; cin >> b_ind;

RHOB_direto_2:
cout << endl;
cout << "REPETIR A OPERACAO? "; cin >> opt;
if (opt == "NAO") goto M;
else if (opt == "SIM") goto RHOB_direto;
else {cout << "Houve um erro. Favor tentar novamente!" << endl <<
endl; goto RHOB_direto_2; }
}

if (opcao5==2) {

//Allocate memory for calibration RHO
depth_RHO = new double[well_obj_data];
RHOB = new double[well_obj_data];

RHOB_10:
png_start(xpmax2,ypmax);
pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmax2-20,ypmax-20);
gridding_RHOB(well_obj,xpmax2,ypmax);

clrscr;
cout << "METODO DE GARDNER - coef. a: "; cin >> a_ind;
cout << " - coef. b: "; cin >> b_ind;
cout << endl;

i=sem_reg;

if (interpol==0) input_0 >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];
if (interpol==1) DTCO_calib >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];

rho_ov[i] = a_ind*pow((pow(10,6)/delta_t0[i]),b_ind);

RHOB_input.open("C:/UFRJDrillingSimulator/Inputs/InputRHO.txt");

i=i+final_depth/data;
while (i<=well_obj)
{
if (interpol==0) input_0 >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];
if (interpol==1) DTCO_calib >> depth[i] >> delta_t0[i] >> lito[i] >>
B_par[i];

rho_ov[i] = a_ind*pow((pow(10,6)/delta_t0[i]),b_ind);
RHOB_drawing(xpmax2,ypmax,depth[i]-LDA,rho_ov[i],depth[i-final_depth/data]-
LDA,rho_ov[i-final_depth/data],well_obj,data,0);
i=i+final_depth/data;
}
}

```

```

    }

    RHOB_input >> depth_RHO[0] >> RHOB[0];

    int transf = depth_RHO[0];
    depth_RHO[transf]=depth_RHO[0]; RHOB[transf]=RHOB[0];

    i = depth_RHO[transf]+final_depth/data;

    RHOB_medido:
        RHOB_input >> depth_RHO[i] >> RHOB[i];

        if (depth_RHO[i] > 0) {
            RHOB_drawing(xpmax2,ypmax,depth_RHO[i]-LDA,RHOB[i],depth_RHO[i-
final_depth/data]-LDA,RHOB[i-final_depth/data],well_obj,data,1);
            i=i+final_depth/data; goto RHOB_medido; }

    png_finish("C:/UFRJDrillingSimulator/Debug/DebugRHOB.png");
    RHOB_input.close();

    RHOB_11:
    cout << endl << endl << "DESEJA REPETIR A OPERACAO? "; cin >> opt;
    if (opt == "NAO") goto M;
        else if (opt == "SIM") { cout << endl << endl; goto RHOB_10; }
        else {cout << "Houve um erro. Favor tentar novamente!" << endl << endl;
goto RHOB_11; }

    //Releasing memory space from these variables delete
    delete depth_RHO;
    delete RHOB;

} }

else if (opcao8==2)
{
    RHOB_index=1;

    RHO_MED_2:

    png_start(xpmax2,ypmax);
    pensize(4);
    bgcolor("white");
    fgcolor("black");
    rectangle(30,20,xpmax2-20,ypmax-20);
    gridding_RHOB(well_obj,xpmax2,ypmax);

    RHOB_input.open("C:/UFRJDrillingSimulator/Inputs/RHO.txt");

    i=sem_reg;

    RHOB_input >> depth[i] >> rho_ov[i];
    i=i+final_depth/data;

    while (i<=well_obj)
    {
        RHOB_input >> depth[i] >> rho_ov[i];
        if (i!=well_obj)
            RHOB_drawing(xpmax2,ypmax,depth[i]-LDA,rho_ov[i],depth[i-final_depth/data]-
LDA,rho_ov[i-final_depth/data],well_obj,data,0);
        i=i+final_depth/data;
    }
}

```

```

RHOB_input.close();
png_finish("C:/UFRJDrillingSimulator/Debug/DebugRHOB.png");
}

cout << endl;

RHOB_MED_3:
cout << "REPETIR A OPERACAO? "; cin >> opt;
    if (opt == "NAO") goto M;
    else if (opt == "SIM") { cout << endl << endl; goto RHO_MED_2; }
    else {cout << "Houve um erro. Favor tentar novamente!" << endl << endl;
goto RHOB_MED_3; }
}

if (opcao==4)
{
extrap = 1;

png_start(xpmax2,ypmax);
pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmax2-20,ypmax-20);
gridding_RHOB(well_obj,xpmax2,ypmax);

//RHOB_teste.open("C:/VICTOR/WELL PROJECT PROGRAM/RHOB_conf.txt");

i=sem_reg;

if (interpol==0) input_0 >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];
if (interpol==1) DTCO_calib >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];

if (RHOB_index==0) rho_ov[i] = a_ind*pow((pow(10,6)/delta_t0[i]),b_ind);

i=i+final_depth/data;

while (i<=well_obj)
{
    if (interpol==0) input_0 >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];
    if (interpol==1) DTCO_calib >> depth[i] >> delta_t0[i] >> lito[i] >>
B_par[i];

    if (RHOB_index==0) rho_ov[i] = a_ind*pow((pow(10,6)/delta_t0[i]),b_ind);

    if (i!=well_obj)
        RHOB_drawing(xpmax2,ypmax,depth[i]-LDA,rho_ov[i],depth[i-final_depth/data]-
LDA,rho_ov[i-final_depth/data],well_obj,data,0);

    //RHOB_teste << depth[i] << " " << delta_t0[i] << " " << rho_ov[i] <<
endl;

    i=i+final_depth/data;
}

png_finish("C:/UFRJDrillingSimulator/Debug/DebugRHOBextrap.png");

clrscr;
opcao7 = tela_RHOB_EXT();
clrscr;

RHOB_extrap:
png_start(xpmax2,ypmax);

```

```

pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmax2-20,ypmax-20);
gridding_RHOB(well_obj,xpmax2,ypmax);
int sem_reg_int;

i=0; sem_reg_int = sem_reg;
while (i<sem_reg)
{
    depth[i] = i+LDA;
    lito[i] = lito[sem_reg_int];
    i = i+final_depth/data;
}

if (opcao7==2){

clrscr;
cout << "EXTRAPOLACAO DA DENSIDADE A PARTIR DE " << LDA << " m ATE " <<
LDA+sem_reg << " m" << endl << endl;

cout << "DETERMINACAO DOS COEFICIENTES DA CURVA EXTRAPOLADA: " << endl << endl;
cout << "Parametro A: "; cin >> a_ext;
cout << "Parametro B: "; cin >> b_ext;
cout << "Parametro C: "; cin >> c_ext;

cout << endl; }

else if (opcao7==1) {

//RHOB_teste.open("C:/VICTOR/WELL PROJECT PROGRAM/RHOB_teste.txt");

cout << "EXTRAPOLACAO DA DENSIDADE A PARTIR DE " << LDA << " m ATE " <<
LDA+sem_reg << " m" << endl << endl;
cout << "COTA LIMITE PARA CALIBRACAO - TVD (m): "; cin >> norm_lim1; cout << endl;

a_ext = -1;
b_ext = -1;
c_ext = 1.8;

while (a_ext<1)
{
while (b_ext<1)
{
while (c_ext<1.9)
{
i=sem_reg;

while (i<=norm_lim1-LDA)
{
RHOB_ext = a_ext*pow((depth[i]-LDA),b_ext)+c_ext;
RHOB_erro = pow((rho_ov[i] - RHOB_ext),2);
variancia = variancia + RHOB_erro;
j=j+1; i=i+final_depth/data;}
dev_pad = pow(variancia/j,0.5);

variancia=0;

if (dev_pad < dev_pad_min) {dev_pad_min=dev_pad; a_ext_min=a_ext;
b_ext_min=b_ext; c_ext_min=c_ext;}

c_ext=c_ext+0.1; } c_ext = 1.8; b_ext=b_ext+0.01; } c_ext = 1.5; b_ext=-1;
a_ext=a_ext+0.01;}

```



```

a_ext = a_ext_min; b_ext = b_ext_min; c_ext = c_ext_min;

cout << "DETERMINACAO DOS COEFICIENTES DA CURVA EXTRAPOLADA: " << endl << endl;
cout << "Parametro A: " << a_ext << endl;
cout << "Parametro B: " << b_ext << endl;
cout << "Parametro C: " << c_ext << endl;
}

//RHOB_teste.close();

i=sem_reg+final_depth/data;

while (i < well_obj)
{
    RHOB_drawing(xpmax2,ypmax,depth[i]-LDA,rho_ov[i],depth[i-final_depth/data]-
LDA,rho_ov[i-final_depth/data],well_obj,data,0);
    i=i+final_depth/data;
}

i=final_depth/data;

while (i <= well_obj)
{
    RHOB_ext_ant = a_ext*pow((depth[i-final_depth/data]-LDA),b_ext)+c_ext;
    RHOB_ext = a_ext*pow((depth[i]-LDA),b_ext)+c_ext;

    if (i!=well_obj)
        RHOB_drawing(xpmax2,ypmax,depth[i]-LDA,RHOB_ext,depth[i-final_depth/data]-
LDA,RHOB_ext_ant,well_obj,data,1);
    i=i+final_depth/data;
}

png_finish("C:/UFRJDrillingSimulator/Debug/DebugRHOBextrap.png");

RHOB_extrap_2:
cout << endl;

cout << "REPETIR A OPERACAO? "; cin >> opt;
    if (opt == "SIM") { cout << endl << endl; goto RHOB_extrap; }
    else if (opt != "NAO") {cout << "Houve um erro. Favör tentar novamente!"
<< endl; goto RHOB_extrap_2; }

DTCO_extrap:
png_start(xpmax,ypmax);
pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmax-20,ypmax-20);
gridding_DTCOxRHOB(well_obj,xpmax,ypmax);

i=sem_reg+final_depth/data;

while (i<well_obj)
{
    DTCO_ext_drawing(xpmax,ypmax,rho_ov[i],delta_t0[i],rho_ov[i-
final_depth/data],delta_t0[i-final_depth/data], well_obj,data,0,1);
    i=i+final_depth/data;
}

i=0;
while (i<sem_reg)
{

```

```

        rho_ov[i] = a_ext*pow((depth[i]-LDA),b_ext)+c_ext;
        i=i+final_depth/data;
    }

png_finish("C:/UFRJDrillingSimulator/Debug/DebugDTextrap.png");
clrscr;

opcao7 = tela_RHOB_EXT();

png_start(xpmax,ypmax);
pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmax-20,ypmax-20);
gridding_DTCOxRHOB(well_obj,xpmax,ypmax);

if (opcao7==2) {

    clrscr;
    cout << "PARAMETROS DA EQUACAO DE GARDNER INVERTIDA NO POCO " << nome;
    cout << endl << endl;
    cout << "EXTRAPOLACAO DA DENSIDADE A PARTIR DE " << LDA << " m ATE " <<
LDA+sem_reg << " m" << endl << endl;
    cout << "Parametro A: "; cin >> a_ext;
    cout << "Parametro B: "; cin >> b_ext;
    cout << endl;
}

else if (opcao7==1) {

    clrscr;
    cout << "PARAMETROS DA EQUACAO DE GARDNER INVERTIDA NO POCO " << nome;
    cout << endl << endl;

    cout << "EXTRAPOLACAO DA DENSIDADE A PARTIR DE " << LDA << " m ATE " <<
LDA+sem_reg << " m" << endl << endl;
    cout << "COTA LIMITE PARA CALIBRACAO - TVD (m): "; cin >> norm_lim1; cout << endl;

    a_ext = 0.2;
    b_ext = 0.2;
    dev_pad_min=1000;

    while (a_ext<0.3)
    {
        while (b_ext<0.3)
        {
            i=sem_reg;
            while (i<norm_lim1-LDA)
            {
                DTCO_ext = pow(10,6)*pow(a_ext/rho_ov[i],1/b_ext);

                DTCO_erro = pow((delta_t0[i] - DTCO_ext),2);
                variancia = variancia + DTCO_erro;
                j=j+1; i=i+final_depth/data;}
                dev_pad = pow(variancia/j,0.5);
                DTCO_conf << a_ext << " " << b_ext << " " << dev_pad << " " <<
a_ext_min << " " << b_ext_min << " " << dev_pad_min << endl;
                variancia=0;
                if (dev_pad < dev_pad_min) {dev_pad_min=dev_pad; a_ext_min=a_ext;
b_ext_min=b_ext; }
                b_ext=b_ext+0.001; } b_ext=0.2; a_ext=a_ext+0.001;}

            a_ext = a_ext_min; b_ext = b_ext_min;

```

```

cout << "DETERMINACAO DOS COEFICIENTES DA CURVA EXTRAPOLADA: " << endl << endl;
cout << "Parametro A: " << a_ext << endl;
cout << "Parametro B: " << b_ext << endl;
cout << endl;

DTCO_conf.close();
}

i=sem_reg+final_depth/data;

while (i < well_obj)
{
    DTCO_ext_drawing(xpmax,ypmax,rho_ov[i],delta_t0[i],rho_ov[i-
final_depth/data],delta_t0[i-final_depth/data],well_obj,data,0,1);
    i=i+final_depth/data;
}

i=final_depth/data;

while (i <= well_obj)
{
    DTCO_ext = pow(10,6)*pow(a_ext/rho_ov[i],1/b_ext);
    DTCO_ext_ant = pow(10,6)*pow(a_ext/rho_ov[i-final_depth/data],1/b_ext);
    if (i!=well_obj)
        DTCO_ext_drawing(xpmax,ypmax,rho_ov[i],DTCO_ext,rho_ov[i-
final_depth/data],DTCO_ext_ant,well_obj,data,1,1);
    i=i+final_depth/data;
}

png_finish("C:/UFRJDrillingSimulator/Debug/DebugDTCOextrap.png");

i=0;

while (i<sem_reg)
{
    delta_t0[i] = pow(10,6)*pow(a_ext/rho_ov[i],1/b_ext);
    i=i+final_depth/data;
}

RHOB_extrap_3:

cout << "REPETIR A OPERACAO?      "; cin >> opt;
if (opt == "NAO") goto M;
else if (opt == "SIM") { cout << endl << endl; goto DTCO_extrap; }
else {cout << "Houve um erro. Favor tentar novamente!" << endl << endl; goto
RHOB_extrap_3; }
}

if (opcao==5)
{
//Allocate memory for Overburden Grad
sigma_ov = new double[well_obj_data];
Gov = new double[well_obj_data];
Gp = new double[well_obj_data];
Gpm = new double[well_obj_data];
Gpmarg = new double[well_obj_data];

clrscr;

k=0;

```

```

Gov_output2.open("C:/UFRJDrillingSimulator/Debug/DebugGOVtrend.txt");
Gov_output.open("C:/UFRJDrillingSimulator/Debug/DebugGOV.txt");
input_0 >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];

rho_ov[i] = a_ind*pow((pow(10,6)/delta_t0[i]),b_ind);
sigma_ov[i] = 1.422*rho_ov[i]*(depth[i]-LDA)+coef*8.5*LDA;
Gov[i] = sigma_ov[i]/(coef*depth[i]);
Gov_output << setiosflags (ios::fixed) << setprecision(2) << depth[i] << " " <<
delta_t0[i] << " " << rho_ov[i] << " " << sigma_ov[i] << " " << Gov[i] <<
endl;

i=i+final_depth/data1;

while (i<=well_obj)
{
    input_0 >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];

    rho_ov[i] = a_ind*pow((pow(10,6)/delta_t0[i]),b_ind);
    delta_D = depth[i]-depth[i-final_depth/data1];
    sigma_ov[i] = 1.422*rho_ov[i]*delta_D+sigma_ov[i-final_depth/data1];
    Gov[i] = sigma_ov[i]/(coef*depth[i]);
    Gov_output2 << setiosflags (ios::fixed) << setprecision(2) << i << " " <<
depth[i] << " " << delta_t0[i] << " " << rho_ov[i] << " " << sigma_ov[i] <<
" " << Gov[i] << endl;
    i=i+(final_depth/data1);
}

i=sem_reg;
Gov_output2 << endl << "#PONTO " << "PROFUNDIDADE " << "DT (us/ft) " << "RHO
(g/cm3) " << "Sov (psi) " << "Gov (ppg)" << endl;
Gov_output2.close();

k=0;

if (extrap==1) {
    i=0;
    while (i<sem_reg)
    {
        sigma_ov[i] = 1.422*rho_ov[i]*(depth[i]-LDA)+coef*8.5*LDA;
        Gov[i] = sigma_ov[i]/(coef*depth[i]);
        Gov_output << setiosflags (ios::fixed) << setprecision(2) <<
depth[i] << " " << delta_t0[i] << " " << rho_ov[i] << " " << sigma_ov[i] <<
" " << Gov[i] << endl;
        i=i+final_depth/data;
    }
}

i=sem_reg;

if (interpol==0) input_0 >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];
if (interpol==1) DT_CO_calib >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];

rho_ov[i] = a_ind*pow((pow(10,6)/delta_t0[i]),b_ind);
sigma_ov[i] = 1.422*rho_ov[i]*(depth[i]-LDA)+coef*8.5*LDA;
Gov[i] = sigma_ov[i]/(coef*depth[i]);
Gov_output << setiosflags (ios::fixed) << setprecision(2) << depth[i] << " " <<
delta_t0[i] << " " << rho_ov[i] << " " << sigma_ov[i] << " " << Gov[i] <<
endl;

i=i+final_depth/data;

while (i<=well_obj)
{
    if (interpol==0) input_0 >> depth[i] >> delta_t0[i] >> lito[i] >> B_par[i];

```

```

        if (interpol==1) DTCO_calib >> depth[i] >> delta_t0[i] >> lito[i] >>
B_par[i];

        rho_ov[i] = a_ind*pow((pow(10,6)/delta_t0[i]),b_ind);
        delta_D = depth[i]-depth[i-final_depth/data];
        sigma_ov[i] = 1.422*rho_ov[i]*delta_D+sigma_ov[i-final_depth/data];
        Gov[i] = sigma_ov[i]/(coef*depth[i]);
        Gov_output << setiosflags (ios::fixed) << setprecision(2) << depth[i] << "
" << delta_t0[i] << "      " << rho_ov[i] << "      " << sigma_ov[i] << "      " << Gov[i]
<< endl;
        i=i+(final_depth/data);
    }

    if (extrap==0) i=sem_reg;
    if (extrap==1) i=0;

    Gov_output << "#PONTO " << "PROFUNDIDADE " << "DT (us/ft) " << "RHO (g/cm3) " <<
"Sov (psi) " << "Gov (ppg)" << endl;

    Gov_output.close();
    opcao2 = tela_GP();
    clrscr;

    if (opcao2==2)
    {
        Gov_output.close();

Gp_externo.open("C:/UFRJDrillingSimulator/Debug/DebugGpexterno.txt");
        Gov_input.open("C:/UFRJDrillingSimulator/Debug/DebugGOV.txt");
        Gpm_output.open("C:/UFRJDrillingSimulator/Debug/DebugGpm.txt");
        clrscr;

        if (extrap==0) i=sem_reg;
        if (extrap==1) i=0;

        while (i<=well_obj)
        {
            Gov_input >> depth[i] >> delta_t0[i] >> rho_ov[i] >>
sigma_ov[i] >> Gov[i];
            Gp_externo >> depth[i] >> Gp[i];
            i = i+(final_depth/data);
        }

        Gov_input.close();
        opcao6 = tela_MediaGp();
        clrscr;

        j=0;
        if (opcao6==1) { while (j<5)
        {
            pond_coef[j]=1;
            pond_coef_sum = pond_coef_sum+pond_coef[j];
            j=j+1; }
        }

        else if (opcao6==2) { cout << "DETERMINACAO DOS COEFICIENTES DE PONDERACAO" <<
endl << endl;

        j=0;
        while (j<5)
        {
            if (j==0) {cout << "Gp_i:      "; cin >> pond_coef[0];}
            else {cout << "Gp_(i-" << j << "): "; cin >> pond_coef[j];}
        }
    }
}

```

```

        pond_coef_sum = pond_coef_sum+pond_coef[j];
        j=j+1; }
cout << endl; }

if (extrap==0) i=sem_reg;
if (extrap==1) i=0;

l=final_depth/data;

Gpm[i] = Gp[i]; Gpmarg[i] = Gpm[i]+error;
Gpm_output << setiosflags (ios::fixed) << setprecision(2) << depth[i] << " " <<
Gov[i] << " " << Gp[i] << " " << Gpm[i] << " " << Gpmarg[i] << endl;

Gpm[i+1] = Gp[i+1]; Gpmarg[i+1] = Gpm[i+1]+error;
Gpm_output << setiosflags (ios::fixed) << setprecision(2) << depth[i+1] << " "
<< Gov[i+1] << " " << Gp[i+1] << " " << Gpm[i+1] << " " << Gpmarg[i+1] << endl;

Gpm[i+2*1] = Gp[i+2*1]; Gpmarg[i+2*1] = Gpm[i+2*1]+error;
Gpm_output << setiosflags (ios::fixed) << setprecision(2) << depth[i+2*1] << " "
<< Gov[i+2*1] << " " << Gp[i+2*1] << " " << Gpm[i+2*1] << " " << Gpmarg[i+2*1]
<< endl;

Gpm[i+3*1] = Gp[i+3*1]; Gpmarg[i+3*1] = Gpm[i+3*1]+error;
Gpm_output << setiosflags (ios::fixed) << setprecision(2) << depth[i+3*1] << " "
<< Gov[i+3*1] << " " << Gp[i+3*1] << " " << Gpm[i+3*1] << " " << Gpmarg[i+3*1] <<
endl;

i = i+4*1;

while (i<=well_obj)
{
    Gpm[i] = (pond_coef[0]*Gp[i]+pond_coef[1]*Gp[i-1]+pond_coef[2]*Gp[i-
2*1]+pond_coef[3]*Gp[i-3*1]+pond_coef[4]*Gp[i-4*1])/pond_coef_sum;;
    Gpmarg[i] = Gpm[i]+error;
    Gpm_output << setiosflags (ios::fixed) << setprecision(2) << depth[i] << "
" << Gov[i] << " " << Gp[i] << " " << Gpm[i] << " " << Gpmarg[i] << endl;
    i=i+final_depth/data;
}

Gpm_output.close();
Gp_externo.close();
}

else if (opcao2==1)
{

opcao4 = tela_MG();
if (opcao4==1)
{

//Allocate memory for delta_tnormalized
delta_tn = new double[well_obj_data];

xpmx=600;
png_start(xpmx, ypmx);
pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmx-20,ypmx-20);
gridding_TREND(well_obj, xpmx, ypmx);

a=sem_reg;

```

```

while(a <= well_obj)
{
if (a==sem_reg) litology_drawing(xpmax,ypmax,lito[a],sem_reg,sem_reg,well_obj);
else litology_drawing(xpmax,ypmax,lito[a],depth[a]-LDA,depth[a-final_depth/data]-
LDA,well_obj);
a=a+final_depth/data;
}

if (extrap==0) i=sem_reg+final_depth/data;
if (extrap==1) i=final_depth/data;

while (i<well_obj)
{
if (lito[i]==1)
trend_curve_drawing(xpmax,ypmax,depth[i]-LDA,delta_t0[i],depth[i-
final_depth/data]-LDA,delta_t0[i-final_depth/data],0,0,well_obj,data,0,0,0,0);
else
trend_curve_drawing(xpmax,ypmax,depth[i]-LDA,delta_t0[i],depth[i-
final_depth/data]-LDA,delta_t0[i-final_depth/data],0,0,well_obj,data,0,0,0,1);
i=i+final_depth/data;
}

png_finish("C:/UFRJDrillingSimulator/Debug/DebugTrendLine.png");
input_0.close();
DTCO_calib.close();
//Gov_output.close();
clrscr;

PR1:
opcao3 = tela_PP();
clrscr;

if (opcao3==2)
{
clrscr;
cor=3;
cout << endl << "*****ZONA DE COMPACTACAO
NORMAL*****" << endl << endl;
cout << "COTA SUPERIOR PARA CALIBRACAO - TVD (m): "; cin >> norm_lim1;
cout << "COTA INFERIOR PARA CALIBRACAO - TVD (m): "; cin >> norm_lim2;

xpmax=600;
png_start(xpmax,ypmax);
pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmax-20,ypmax-20);
gridding_TREND(well_obj, xpmax, ypmax);

a=sem_reg;
while(a <= well_obj)
{
if (a==sem_reg) litology_drawing(xpmax,ypmax,lito[a],sem_reg,sem_reg,well_obj);
else litology_drawing(xpmax,ypmax,lito[a],depth[a]-LDA,depth[a-final_depth/data]-
LDA,well_obj);
a=a+final_depth/data;
}

Gov_input.open("C:/UFRJDrillingSimulator/Debug/DebugGOV.txt");
Gp_output.open("C:/UFRJDrillingSimulator/Debug/DebugGP.txt");

i = sem_reg;
dif_RFT_min=10;

```

```

RFT_output.open("C:/UFRJDrillingSimulator/Debug/DebugRFT.txt" );
srt=LDA;

data2 = data;
data = data1;

while (i<=well_obj)
{
    Gov_input >> depth[i] >> delta_t0[i] >> rho_ov[i] >> sigma_ov[i] >> Gov[i];
    i = i+(final_depth/data);
}

int L=final_depth/data;

while (s<10)
{
    f1 = norm_lim1-LDA;

    X0:
    if (lito[f1] == 1) f1=f1;
    else { f1 = f1+final_depth/data; goto X0; }

    while (f1 <= norm_lim2-LDA)
    {
        D0 = depth[f1]; delta_t0p = delta_t0[f1];
        f2 = f1+final_depth/data;

        X1:
        if (lito[f2] == 1) f2=f2;
        else { f2 = f2+final_depth/data; goto X1; }

        while (f2 <= norm_lim2-LDA)
        {
            D1 = depth[f2]; delta_t1p = delta_t0[f2];
            m_inc = (log10(delta_t1p/delta_t0p))/(D1-D0);

            delta_tn[RFTp1-LDA] = delta_t0p*pow(10,(m_inc*(RFTp1-D0)));
            Gp[RFTp1-LDA] = Gov[RFTp1-LDA]-(Gov[RFTp1-LDA]-
8.5)*pow((delta_tn[RFTp1-LDA]/delta_t0[RFTp1-LDA]),coef_Eaton);

            delta_tn[RFTp2-srt] = delta_t0p*pow(10,m_inc*(depth[RFTp2-
srt]-D0));
            Gp[RFTp2-srt] = Gov[RFTp2-srt]-(Gov[RFTp2-srt]-
8.5)*pow((delta_tn[RFTp2-srt]/delta_t0[RFTp2-srt]),coef_Eaton);

            dif_RFT_1 = pow((RFTd1-Gp[RFTp1-srt]),2);
            dif_RFT_2 = pow((RFTd2-Gp[RFTp2-srt]),2);

            dif_RFT = pow((dif_RFT_1+dif_RFT_2)/2,0.5);
            dif_RFT_teste = pow(dif_RFT_1,0.5)+pow(dif_RFT_2,0.5);;

            RFT_output << f1+LDA << " " << f2+LDA << " " <<
delta_tn[RFTp1-LDA] << " " << delta_t0[RFTp1-LDA] << " " << RFTd1 << " " <<
Gp[RFTp1-srt] << " " << RFTd2 << " " << Gp[RFTp2-srt] << " " << dif_RFT_teste
<< " " << dif_RFT << endl;

            if (dif_RFT < dif_RFT_min) {dif_RFT_min = dif_RFT; g_norm1 =
f1; g_norm2 = f2;}

            f2 = f2+final_depth/data;

```



```

    }
    fl = fl+final_depth/data;
    }
    s=s+1;}

    srt=LDA+sem_reg;
    RFT_output.close();
    Gov_input.close();
    data=data2;

    cout << endl << "*****GRAD. COMPACTACAO
NORMAL*****" << endl << endl;

    D0 = depth[g_norm1]; D1 = depth[g_norm2];
    delta_t0p = delta_t0[g_norm1]; delta_t1p = delta_t0[g_norm2];

    cout << "PONTOS DA TREND CURVE:" << endl << endl;
    cout << "TVD 1:      " << D0 << " m" << endl;
    cout << "DTCO  1:      " << delta_t0p << " us/ft" << endl << endl;

    cout << "TVD 2:      " << D1 << " m" << endl;
    cout << "DTCO  2:      " << delta_t1p << " us/ft" << endl << endl;
    }

    else if (opcao3==1)
    {

    xpmx=600;
    png_start(xpmx,ypmx);
    pensize(4);
    bgcolor("white");
    fgcolor("black");
    rectangle(30,20,xpmx-20,ypmx-20);
    gridding_TREND(well_obj, xpmx, ypmx);

    a=sem_reg;
    while(a <= well_obj)
    {
    if (a==sem_reg) litology_drawing(xpmx,ypmx,lito[a],sem_reg,sem_reg,well_obj);
    else litology_drawing(xpmx,ypmx,lito[a],depth[a]-LDA,depth[a-final_depth/data]-
LDA,well_obj);
    a=a+final_depth/data;
    }

    cout << endl << "*****ZONA DE COMPACTACAO
NORMAL*****" << endl << endl;
    cout << "COTA SUPERIOR PARA CALIBRACAO - TVD (m): "; cin >> norm_lim1;
    cout << "COTA SUPERIOR PARA CALIBRACAO - TVD (m): "; cin >> norm_lim2;

    Gov_input.open("C:/UFRJDrillingSimulator/Debug/DebugGOV.txt");
    Gp_output.open("C:/UFRJDrillingSimulator/Debug/DebugGP.txt");

    RFT_output.open("C:/UFRJDrillingSimulator/Debug/DebugRFT.txt" );
    srt=LDA;

    data2 = data;
    data = data1;

    while (s<10)
    {
    dev_pad_min=1000;
    fl = norm_lim1-LDA;

```

```

X2:
    if (lito[f1] == 1) f1=f1;
    else { f1 = f1+final_depth/data; goto X2; }

i = f1;

while (f1 <= norm_lim2-LDA)
{
    D0 = depth[f1]; delta_t0p = delta_t0[f1];
    f2 = f1+final_depth/data;

    X3:
    if (lito[f2] == 1) f2=f2;
    else { f2 = f2+final_depth/data; goto X3; }

    while (f2 <= norm_lim2-LDA)
    {
        D1 = depth[f2]; delta_t1p = delta_t0[f2];
        m_inc = (log10(delta_t1p/delta_t0p))/(D1-D0);
        i=norm_lim1-LDA;
        while (i<norm_lim2-LDA)
        {
            X4:
            if (lito[i] != 1) { i=i+final_depth/data; goto X4; }

            delta_tn[i] = delta_t0p*pow(10, (m_inc*(depth[i]-D0)));

            DTCO_erro = pow((delta_tn[i]-delta_t0[i]),2);
            variancia = variancia + DTCO_erro;
            i=i+final_depth/data2; j=j+1;
        }

        dev_pad = pow(variancia/j,0.5);
        if (dev_pad < dev_pad_min) {dev_pad_min = dev_pad; g_norm1 =
f1; g_norm2 = f2;}

        variancia=0;

        RFT_output << f1+LDA << "          " << f2+LDA << "          " <<
dev_pad << "          " << dev_pad_min << endl;

        f2 = f2+final_depth/data;

        X6:
        if (lito[f2] == 1) f2=f2;
        else { f2 = f2+final_depth/data; goto X6; }
        }
        f1 = f1+final_depth/data;

        X7:
        if (lito[f1] == 1) f1=f1;
        else { f1 = f1+final_depth/data; goto X7; }
        }

        s=s+1;
        }
        srt=LDA+sem_reg;

        Gov_input.close();
        data=data2;

RFT_output.close();

```

```

    cout << endl << "*****GRAD. COMPACTACAO
NORMAL*****" << endl << endl;

    D0 = depth[g_norm1]; D1 = depth[g_norm2];
    delta_t0p = delta_t0[g_norm1]; delta_t1p = delta_t0[g_norm2];

    cout << "PONTOS DA TREND CURVE:" << endl << endl;
    cout << "TVD 1:      " << D0 << " m" << endl;
    cout << "DTCO  1:      " << delta_t0p << " us/ft" << endl << endl;

    cout << "TVD 2:      " << D1 << " m" << endl;
    cout << "DTCO  2:      " << delta_t1p << " us/ft" << endl << endl;
}

else if (opcao3==3)
{
    cor=1;
    cout << "PONTOS DA TREND CURVE:" << endl << endl;
    cout << "TVD 1 (m) :      "; cin >> D0;
    cout << "DTCO (us/ft):      "; cin >> delta_t0p;
    cout << endl;
    cout << "TVD 2 (m) :      "; cin >> D1;
    cout << "DTCO (us/ft):      "; cin >> delta_t1p;
    cout << endl;
}

m_inc = (log10(delta_t1p/delta_t0p))/(D1-D0);

if (extrap==0) i=sem_reg;
if (extrap==1) i=0;

delta_tn[i] = delta_t0p*pow(10,m_inc*(depth[i]-D0));

Gp[i] = Gov[i]-(Gov[i]-8.5)*pow((delta_tn[i]/delta_t0[i]),coef_Eaton);
if (Gp[i] < 8.5) Gp[i]=8.5;
Gp_output << setiosflags (ios::fixed) << setprecision(2) << i << " " << depth[i]
<< " " << rho_ov[i] << " " << " " << " " << sigma_ov[i] << " " << delta_tn[i] << "
" << Gov[i] << " " << Gp[i] << endl;

i=i+final_depth/data;

while (i<=well_obj)
{
    delta_tn[i] = delta_t0p*pow(10,m_inc*(depth[i]-D0));

    if (i!=well_obj)

        if (lito[i] == 1)
            trend_curve_drawing(xpmax,ypmax,depth[i]-LDA,delta_t0[i],depth[i-
final_depth/data]-LDA,delta_t0[i-final_depth/data],delta_tn[i],delta_tn[i-
final_depth/data],well_obj,data,1,0,1,0);
        else
            trend_curve_drawing(xpmax,ypmax,depth[i]-LDA,delta_t0[i],depth[i-
final_depth/data]-LDA,delta_t0[i-final_depth/data],delta_tn[i],delta_tn[i-
final_depth/data],well_obj,data,1,0,1,1);

        Gp[i] = Gov[i]-(Gov[i]-8.5)*pow((delta_tn[i]/delta_t0[i]),coef_Eaton);
        if (Gp[i] < 8.5) Gp[i]=8.5;
        Gp_output << setiosflags (ios::fixed) << setprecision(2) <<depth[i] << "
" << rho_ov[i] << " " << " " << sigma_ov[i] << " " << delta_tn[i] << " " <<
Gov[i] << " " << Gp[i] << endl;
        i = i+(final_depth/data);
}

```

```

        if (extrap==0) i=sem_reg;
        if (extrap==1) i=0;

png_finish("C:/UFRJDrillingSimulator/Debug/DebugGTrendLine2.png");

//Gov_input.close();
Gp_output << endl << "#PONTO " << "PROFUNDIDADE " << "TENSAO DE SOBRACARGA [psi] "
<< "GRADIENTE DE POROS [ppg] " << "GRADIENTE DE SOBRECARGA " << endl;
Gp_output.close();

PR20:

cout << "REPETIR A OPERACAO?      "; cin >> opt;
    if (opt == "NAO") goto PR_cont;
    else if (opt == "SIM") { cout << endl << endl; goto PR1; }
    else {cout << "Houve um erro. Favor tentar novamente!" << endl << endl;
goto PR20; }

Gp1_input.open("C:/UFRJDrillingSimulator/Debug/DebugGP.txt");

PR_cont:
while (i<=well_obj)
{
    Gp1_input >> depth[i] >> rho_ov[i] >> sigma_ov[i] >> delta_tn[i] >> Gov[i]
>> Gp[i];
    i = i+(final_depth/data);
}
}

Gp1_input.close();
Gpm_output.open("C:/UFRJDrillingSimulator/Debug/DebugGpm.txt");

if (opcao4==2)
{

input_0.close();
DTCO_calib.close();
Gov_output.close();
clrscr;
BOW2:

RFT_output.open("C:/VICTOR/WELL PROJECT PROGRAM/RFT_teste.txt");

clrscr;

if (extrap==0) i=sem_reg;
if (extrap==1) i=0;

while (i<well_obj)
{
    if (delta_t0[i] < delta_t0_min) delta_t0_min = delta_t0[i];
    i=i+final_depth/data;
}

double Vmax = (pow(10,6)/delta_t0_min);
double Smax = pow((Vmax-Vo)/A,1/B);
double *V, *S, Amin, Bmin, Vomin, Umin, dif_RFT_min=1000;

opcao9 = tela_BOWERS_OPTION();
if (opcao9==1)
{

```

```

    cout << "PARAMETROS DE BOWERS " << endl << endl;
    cout << "Parametro A:          "; cin >> A;
    cout << "Parametro B:          "; cin >> B;
    cout << "Parametro U:          "; cin >> U;
    cout << "Parametro Vo [ft/s]: "; cin >> Vo;
}

if (opcao9==2)
{
i=RFTp1-LDA;
j=RFTp2-LDA;

A=6; B=0.1; U=3;

//Allocate memory for Bowers method
V = new double[well_obj_data];
S = new double[well_obj_data];

while (A < 30)
{
while (B < 1)
{
while (U < 5)
{
V[i] = (pow(10,6)/delta_t0[i]);
V[j] = (pow(10,6)/delta_t0[j]);

if (B_par[i] == 0) S[i] = sigma_ov[i]-8.5*coef*depth[i];
if (B_par[i] == 1) S[i] = pow((V[i]-Vo)/A,1/B);
if (B_par[i] == 2) S[i] = Smax*pow((1/Smax)*pow((V[i]-Vo)/A,1/B),U);

if (B_par[j] == 0) S[j] = sigma_ov[j]-8.5*coef*depth[j];
if (B_par[j] == 1) S[j] = pow((V[j]-Vo)/A,1/B);
if (B_par[j] == 2) S[j] = Smax*pow((1/Smax)*pow((V[j]-Vo)/A,1/B),U);

Gp[i] = (sigma_ov[i] - S[i])/(coef*depth[i]);
Gp[j] = (sigma_ov[j] - S[j])/(coef*depth[j]);

dif_RFT_1 = pow((RFTd1-Gp[i]),2);
dif_RFT_2 = pow((RFTd2-Gp[j]),2);

dif_RFT = pow((dif_RFT_1+dif_RFT_2)/2,0.5);

if (dif_RFT < dif_RFT_min) {Amin = A; Bmin = B; Vomn = Vo; Umin = U;
dif_RFT_min=dif_RFT;}
U=U+0.01; } U=3; B=B+0.01;} B=0.1; A=A+0.1;}

U = Umin; A = Amin; B = Bmin; Vo = Vomn;

cout << "PARAMETROS DE BOWERS " << endl << endl;
cout << "Parametro A:          " << A << endl;
cout << "Parametro B:          " << B << endl;
cout << "Parametro U:          " << U << endl;
cout << "Parametro Vo:          " << Vo << " ft/s" << endl;
}

if (extrap==0) i=sem_reg;
if (extrap==1) i=0;

for (i=i; i<=well_obj; i=i+final_depth/data)
{
V[i] = (pow(10,6)/delta_t0[i]);

```

```

if (B_par[i] == 0) S[i] = sigma_ov[i]-8.5*coef*depth[i];
if (B_par[i] == 1) S[i] = pow((V[i]-Vo)/A,1/B);
if (B_par[i] == 2) S[i] = Smax*pow((1/Smax)*pow((V[i]-Vo)/A,1/B),U);

Gp[i] = (sigma_ov[i] - S[i])/(coef*depth[i]);
if (Gp[i] < 8.5) Gp[i]=8.5;
}

BOW1:
cout << endl << endl;
cout << "REPETIR A OPERACAO?      "; cin >> opt;
if (opt == "NAO") goto BOW3;
else if (opt == "SIM") { cout << endl << endl; goto BOW2; }
else {cout << "Houve um erro. Favor tentar novamente!" << endl; goto BOW1; }

BOW3:
RFT_output.close();

delete V;
delete S;
}

clrscr;
opcao6 = tela_MediaGp();
clrscr;

j=0;
if (opcao6==1)
{
while (j<5)

{
    pond_coef[j]=1;
    pond_coef_sum = pond_coef_sum+pond_coef[j];
    j=j+1; }

else if (opcao6==2)
{
    cout << "SUAVIZACAO DO PERFIL DE GRADIENTE DE POROS NO POCO " << nome << endl
<< endl;
    cout << "DETERMINACAO DOS COEFICIENTES DE PONDERACAO" << endl << endl;

j=0;
while (j<5)
{
    if (j==0) {cout << "Gp_i:      "; cin >> pond_coef[j];}
    else {cout << "Gp_(i-" << j << "):  "; cin >> pond_coef[j];}
    pond_coef_sum = pond_coef_sum + pond_coef[j];
    j=j+1; }

cout << endl;
}

if (extrap==0) i=sem_reg;
if (extrap==1) i=0;

l = final_depth/data;

Gpm[i] = Gp[i]; Gpmarg[i] = Gpm[i]+error;
Gpm_output << depth[i] << "      " << Gov[i] << "      " << Gp[i] << "      " << Gpm[i] <<
"      " << Gpmarg[i] << endl;

```

```

    Gpm[i+1] = Gp[i+1]; Gpmarg[i+1] = Gpm[i+1]+error;
    Gpm_output << depth[i+1] << " " << Gov[i+1] << " " << Gp[i+1] << " " <<
Gpm[i+1] << " " << Gpmarg[i+1] << endl;

    Gpm[i+2*1] = Gp[i+2*1]; Gpmarg[i+2*1] = Gpm[i+2*1]+error;
    Gpm_output << depth[i+2*1] << " " << Gov[i+2*1] << " " << Gp[i+2*1] << " " <<
Gpm[i+2*1] << " " << Gpmarg[i+2*1] << endl;

    Gpm[i+3*1] = Gp[i+3*1]; Gpmarg[i+3*1] = Gpm[i+3*1]+error;
    Gpm_output << depth[i+3*1] << " " << Gov[i+3*1] << " " << Gp[i+3*1] << " " <<
Gpm[i+3*1] << " " << Gpmarg[i+3*1] << endl;

    i = i+4*1;

    while (i<=well_obj)
    {
        Gpm[i] = (pond_coef[0]*Gp[i]+pond_coef[1]*Gp[i-1]+pond_coef[2]*Gp[i-
2*1]+pond_coef[3]*Gp[i-3*1]+pond_coef[4]*Gp[i-4*1])/pond_coef_sum;;
        Gpmarg[i] = Gpm[i]+error;
        Gpm_output << setiosflags (ios::fixed) << setprecision(2) << depth[i] << "
" << Gov[i] << " " << Gp[i] << " " << Gpm[i] << " " << Gpmarg[i] << endl;
        i=i+final_depth/data;
    }

    Gpm_output.close();
}

PR2:
cout << "REPETIR A OPERACAO? "; cin >> opt;
if (opt == "NAO") goto M;
else if (opt == "SIM") { cout << endl << endl; goto PR1; }
else {cout << endl << "Houve um erro. Favor tentar novamente!" << endl << endl;
goto PR2; }
}

if (opcao==6)
{
delete sigma_ov;
delete rho_ov;
delete delta_t0;
delete B_par;
delete delta_tn;

LOT1:
Gpm_input.open("C:/UFRJDrillingSimulator/Debug/DebugGpm.txt");

clrscr;
cout << "PROCESSANDO "; Sleep(300); cout << ">"; Sleep(300); cout << ">";
Sleep(300); cout << ">"; Sleep(300); cout << ">";
cout << " PROCESSANDO!" << endl << endl;

int z=0;

if (extrap==0) i=sem_reg;
if (extrap==1) i=0;

while (i<=well_obj)
{
    Gpm_input >> depth[i] >> Gov[i] >> Gp[i] >> Gpm[i] >> Gpmarg[i];
    i = i+(final_depth/data);
}

```

```

if (extrap==0) i=sem_reg;
if (extrap==1) i=0;

    K1 = (G_LOT1 - Gpm[Df1-LDA]) / (Gov[Df1-LDA] - Gpm[Df1-LDA]); Ln1 = log(Df1);
    K2 = (G_LOT2 - Gpm[Df2-LDA]) / (Gov[Df2-LDA] - Gpm[Df2-LDA]); Ln2 = log(Df2);
    K3 = (G_LOT3 - Gpm[Df3-LDA]) / (Gov[Df3-LDA] - Gpm[Df3-LDA]); Ln3 = log(Df3);

beta = reg_beta(K1, K2, K3, Ln1, Ln2, Ln3);
alfa = reg_alfa(beta, K1, K2, K3, Ln1, Ln2, Ln3);

Gpm_input.close();
Gf_output.open("C:/UFRJDrillingSimulator/Debug/DebugGF.txt");

//Allocate memory for Fract. Grad. parameters
Gf = new double[well_obj_data];
Gfmarg = new double[well_obj_data];
K = new double[well_obj_data];

while (i<=well_obj)
{

    K[i] = beta*log(depth[i])+alfa;
    Gf[i] = Gpm[i] + K[i]*(Gov[i] - Gpm[i]);
    Gfmarg[i] = Gf[i]-error;
    Gf_output << setiosflags (ios::fixed) << setprecision(2) << depth[i] << " " <<
Gpm[i] << " " << Gpmarg[i] << " " << Gf[i] << " " << Gfmarg[i] << " " <<
Gov[i] <<endl;
    i = i+(final_depth/data);
}

Gf_output.close();

LOT2:
cout << endl << endl;
cout << "REPETIR A OPERACAO? "; cin >> opt;
if (opt == "NAO") goto M;
else if (opt == "SIM") { cout << endl << endl; goto LOT1; }
else {cout << "Houve um erro. Favor tentar novamente!" << endl; goto LOT2; }

}

Gf_fin.open("C:/UFRJDrillingSimulator/Debug/DebugGF.txt");

if (extrap==0) i=sem_reg;
if (extrap==1) i=0;

while (i<=well_obj)
{
    Gf_fin >> depth[i] >> Gpm[i] >> Gpmarg[i] >> Gf[i] >> Gfmarg[i] >> Gov[i];
    i = i+(final_depth/data);

delete K;
}

if (opcao==7)
{
    clrscr;
    G_total_input.open("C:/UFRJDrillingSimulator/Debug/DebugGF.txt");

    n=well_obj;

    m = well_obj-final_depth/data;
    xpmx = 1024;

```



```

png_start(xpmax,ypmax);
pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmax-20,ypmax-20);

gridding(well_obj,10,xpmax,ypmax,Gov[m],8.35);

if (extrap==0) k=sem_reg;
if (extrap==1) k=0;
i=k;
a=sem_reg;

while (k < well_obj)
{
    G_total_input >> depth[k] >> Gpm[k] >> Gpmarg[k] >> Gf[k] >> Gfmarg[k]
>> Gov[k];

    if (Gpm[k] > Gp_max) {Gp_max = Gpm[k]; Dh = depth[k];}
    if (k==i)
{grad_drawing(xpmax,ypmax,Gpm[k],Gpm[k],Gf[k],Gf[k],Gov[k],Gov[k],depth[k]-
LDA,depth[k]-LDA,well_obj,8.35,Gov[m],0);

grad_drawing(xpmax,ypmax,Gpmarg[k],Gpmarg[k],Gfmarg[k],Gfmarg[k],Gov[k],Gov[k],depth[k]
]-LDA,depth[k]-LDA,well_obj,8.35,Gov[m],1); }

    else if (k==well_obj)
grad_drawing(xpmax,ypmax,0,0,0,0,0,0,0,well_obj,8.35,Gov[m],0);

    else { grad_drawing(xpmax,ypmax,Gpm[k],Gpm[k-
final_depth/data],Gf[k],Gf[k-final_depth/data],Gov[k],Gov[k-
final_depth/data],depth[k]-LDA,depth[k-final_depth/data]-LDA,well_obj,8.35,Gov[m],0);
        grad_drawing(xpmax,ypmax,Gpmarg[k],Gpmarg[k-
final_depth/data],Gfmarg[k],Gfmarg[k-final_depth/data],Gov[k],Gov[k-
final_depth/data],depth[k]-LDA,depth[k-final_depth/data]-LDA,well_obj,8.35,Gov[m],1);
    }

    k = k+final_depth/data;
}

Gpmarg_max = Gpmarg[n];
i=well_obj;

while ((well_obj-i) < 1000)
{
    if (Gpmarg[i-final_depth/data] > Gpmarg_max) Gpmarg_max = Gpmarg[i-
final_depth/data];
    else Gpmarg_max = Gpmarg_max;
    i=i-final_depth/data;
}

if (Gpmarg_max > Gfmarg[n]) density[N] = Gfmarg[n];

else density[N] = Gpmarg_max;
depth_new[N] = depth[n];

int g=0;

pontos_drawing(xpmax,ypmax,RFTp1-LDA,RFTd1,RFTp2-LDA,RFTd2,Df1-
LDA,G_LOT1,Df2-LDA,G_LOT2,Df3-LDA,G_LOT3,well_obj,8.35,Gov[m]);

while(a <= well_obj)
{

```

```

        if (a==sem_reg)
litology_drawing(xpmax,ypmax,lito[a],sem_reg,sem_reg,well_obj);
        else litology_drawing(xpmax,ypmax,lito[a],depth[a]-LDA,depth[a-
final_depth/data]-LDA,well_obj);
        a=a+final_depth/data;
    }

    while (n > comp_sup)
    {
        p_porous = coef*(depth[n])*Gpmarg[n];
        p_fracture = coef*(depth[n])*Gfmarg[n];
        p_bh = gradient_criteria(depth[n],density[N],coef);

        if (p_bh >= p_fracture)
        {
            Gpmarg_max = Gpmarg[n];

            while ((n-g)>0)
            {
                if (Gpmarg[n-g] >= Gpmarg_max) Gpmarg_max=Gpmarg[n-g];
                else Gpmarg_max=Gpmarg_max;
                g=g+final_depth/data;
            }

            depth_new[N+1] = depth[n];
            density[N+1] = Gpmarg_max;
            depth_old[y] = depth[n];
            gradcrit_drawing(xpmax,ypmax,density[N],density[N+1],depth_new[N]-
LDA,depth_new[N+1]-LDA,well_obj,8.35,1,Gov[m]);
            g=0; Gpmarg_max = 8.5; N=N+1; y=y+1;
            if (depth[n]-srt <= 500) goto WD1;
        }

        else n=n-final_depth/data;
    }
    gradcrit_drawing(xpmax,ypmax,density[N],density[N],depth_new[N]-
LDA,0,well_obj,8.35,2,Gov[m]);
    png_finish("C:/UFRJDrillingSimulator/Outputs/OutputCritGradientses.png");
}

if (opcao==8)
{
    MR = new double[well_obj_data];

    clrscr;
    G_total_input.open("C:/UFRJDrillingSimulator/Debug/DebugGF.txt");

    n=well_obj;

    m = well_obj-final_depth/data;
    xpmax = 1024;
    png_start(xpmax,ypmax);
    pensize(4);
    bgcolor("white");
    fgcolor("black");
    rectangle(30,20,xpmax-20,ypmax-20);

    gridding(well_obj,10,xpmax,ypmax,Gov[m],8.35);

    if (extrap==0) k=sem_reg;
    if (extrap==1) k=0;
    i=k; a=k;
}

```

```

while (k <= well_obj)
{
    G_total_input >> depth[k] >> Gpm[k] >> Gpmarg[k] >> Gf[k] >> Gfmarg[k]
>> Gov[k];

    MR[k] = (10.5*depth[k] - 8.5*LDA)/(depth[k]-LDA);

    Gpmarg[k] = MR[k];
    Gfmarg[k] = Gf[k];

    if (Gpm[k] > Gp_max) {Gp_max = Gpm[k]; Dh = depth[k];}
    if (k==i)
{grad_drawing(xpmax, ypmax, Gpm[k], Gpm[k], Gf[k], Gf[k], Gov[k], Gov[k], depth[k]-
LDA, depth[k]-LDA, well_obj, 8.35, Gov[m], 0);

grad_drawing(xpmax, ypmax, Gpmarg[k], Gpmarg[k], Gfmarg[k], Gfmarg[k], Gov[k], Gov[k], depth[k]
]-LDA, depth[k]-LDA, well_obj, 8.35, Gov[m], 1); }

    else if (k==well_obj)
grad_drawing(xpmax, ypmax, 0, 0, 0, 0, 0, 0, 0, well_obj, 8.35, Gov[m], 0);

    else { grad_drawing(xpmax, ypmax, Gpm[k], Gpm[k-
final_depth/data], Gf[k], Gf[k-final_depth/data], Gov[k], Gov[k-
final_depth/data], depth[k]-LDA, depth[k-final_depth/data]-LDA, well_obj, 8.35, Gov[m], 0);
        grad_drawing(xpmax, ypmax, Gpmarg[k], Gpmarg[k-
final_depth/data], 0, 0, Gov[k], Gov[k-final_depth/data], depth[k]-LDA, depth[k-
final_depth/data]-LDA, well_obj, 8.35, Gov[m], 1); }

    k = k+final_depth/data;
}

density[N] = Gpmarg[n];
depth_new[N] = depth[n];
int g=0;

pontos_drawing(xpmax, ypmax, RFTp1-LDA, RFTd1, RFTp2-LDA, RFTd2, Df1-
LDA, G_LOT1, Df2-LDA, G_LOT2, Df3-LDA, G_LOT3, well_obj, 8.35, Gov[m]);

while(a <= well_obj)
{
    if (a==sem_reg)
litology_drawing(xpmax, ypmax, lito[a], sem_reg, sem_reg, well_obj);
    else litology_drawing(xpmax, ypmax, lito[a], depth[a]-LDA, depth[a-
final_depth/data]-LDA, well_obj);
    a=a+final_depth/data;
}

while (n>comp_sup)
{
    p_porous = coef*(depth[n])*Gpmarg[n];
    p_fracture = coef*(depth[n])*Gfmarg[n];
    p_bh = gradient_criteria(depth[n], density[N], coef);

    if (p_bh >= p_fracture)
    {
        Gpmarg_max=Gpmarg[n];

        while (n-g>0)
        {
            if (Gpmarg[n-g] >= Gpmarg_max) Gpmarg_max=Gpmarg[n-g];
            else Gpmarg_max=Gpmarg_max;
            g=g+final_depth/data;
        }
    }
}

```



```

ODcasing >> ODrev[x1][x2] >> ODest[x1][x2];
x1=x1+1;
}
x1=0; x2=x2+1;}

x1=0; x2=0;

int line;

D:
if (ODrev[0][x2]==ODprod) {if (ODn[x2]==(p+1)) {line=x2; goto D1;}
else x2=x2+1; goto D;}
else {x2=x2+1; goto D;}

D1:
ODcasing.close();

x1=ODn[line]-1;
int x3=1;

cout << "OD Rev: " << setw(6) << d_ext[0] << " in OD Fase: " << setw(6)
<< d_stage[0] << " in GRAU Aco - API " << setw(6); cin >> API[0];

while (x1>=0)
{
d_ext[x3] = ODrev[x1][line]; d_stage[x3] = ODest[x1][line];

if (x3==1) {cout << "OD Rev: " << setw(6) << d_ext[x3] << " in OD Fase: "
<< setw(6) << d_stage[x3] << " in GRAU Aco - API " << setw(6); cin >> API[x3]; }

else if (x1==0) {cout << "OD Rev: " << setw(6) << d_ext[x3] << " in OD
Fase: " << setw(6) << d_stage[x3] << " in GRAU Aco - API " << setw(6); cin >>
API[x3] >> c_tipo[x3];}

else {cout << "OD Rev: " << setw(6) << d_ext[x3] << " in OD Fase: " <<
setw(6) << d_stage[x3] << " in GRAU Aco - API " << setw(6); cin >> API[x3] >>
c_tipo[x3];}
x1=x1-1; x3=x3+1;
}}

if (opcao1==2)
{
clrscr;
cout << "Determine: OD DA BROCA, OD DA REVESTIMENTO, GRAU E TIPO DO ACO: "
<< endl << endl;

cout << "ESTAGIO 1: "; cin >> d_stage[0]; cout << setw(6); cin >>
d_ext[0]; cout << setw(6); cin >> API[0];
cout << "ESTAGIO 2: "; cin >> d_stage[1]; cout << setw(6); cin >>
d_ext[1]; cout << setw(6); cin >> API[1];

j=2;
while (j<=p+1)
{
cout << "ESTAGIO " << j+1 << ": "; cin >> d_stage[j]; cout <<
setw(6); cin >> d_ext[j]; cout << setw(6); cin >> API[j]; cout << setw(6); cin >>
c_tipo[j];

j=j+1;
} }

ofstream ROP_conf, time_conf, Bit_Rec_conf;
ifstream life_bite_input, Bit_Record, Casing_Plac;

```

```

Borgoyne_input.open("C:/UFRJDrillingSimulator/Inputs/InputBorgoyneParameters.txt");
//life_bite_input.open("C:/VICTOR/WELL PROJECT
PROGRAM/life_bite_input.txt");
ROP_conf.open("C:/UFRJDrillingSimulator/Debug/DebugROP.txt");
time_conf.open("C:/UFRJDrillingSimulator/Debug/DebugTime.txt");
Bit_Record.open("C:/UFRJDrillingSimulator/Inputs/InputBitRecord.txt");

Casing_Plac.open("C:/UFRJDrillingSimulator/Inputs/InputCasingPlacement.txt");
Bit_Rec_conf.open("C:/UFRJDrillingSimulator/Debug/DebugBitRecord.txt");

i=0;
int CPL, Mcas=0;
double t_cas[20];

Casing_Plac >> CPL; //total de dados
Casing_Plac >> t_desl; //tempo de deslocamento da plataforma

//Allocate memory for Drilling Model
ROP = new double[well_obj_data]; //Rate of Penetration
t_sum = new double[well_obj_data]; //auxiliar t parameter
t = new double[well_obj_data]; //time
t_sum_plot = new double[well_obj_data]; //time accumulator
Di = new double[well_obj_data]; //Initial section from bit record
Dfim = new double[well_obj_data]; //Final sextion from bit record
T = new double[well_obj_data]; //time for bit record
Tac = new double[well_obj_data];
H1 = new double[well_obj_data]; //
H2 = new double[well_obj_data];
WOB_d_m = new double[well_obj_data];
WOB = new double[well_obj_data]; //weight on bit from bit record
Freq = new double[well_obj_data]; //bit rotation from bit record
jato1 = new double[well_obj_data]; //tamanho jato 1 from bit record
jato2 = new double[well_obj_data]; //tamanho jato 2 from bit record
jato3 = new double[well_obj_data]; //tamanho jato 3 from bit record
double sum_XY=0;
double sum_X=0, sum_X2=0;
double sum_Y=0, sum_Y2=0;
double t_aux;
int lit;

//leitura de record de tempo de assentamento de sapata com profundidade
while (i<CPL)
{
    Casing_Plac >> depth_cas >> t_cas[i];
    sum_XY = sum_XY + depth_cas*t_cas[i];
    sum_X = sum_X + depth_cas;
    sum_Y = sum_Y + t_cas[i];
    sum_X2 = sum_X2 + pow(depth_cas,2);
    sum_Y2 = sum_Y2 + pow(t_cas[i],2);
    //cout << depth_cas << " " << t_cas[i] << " " << sum_X << " " << sum_Y
<< " " << sum_XY << " " << sum_X2 << " " << sum_Y2 << endl;
    i=i+1;
}

//estimar curva de tempo de assentamento por profundidade
beta2_num = i*sum_XY - sum_X*sum_Y;
beta2_den = i*sum_X2 - pow(sum_X,2);
beta2 = beta2_num/beta2_den;
alfa2 = sum_Y/i - beta2*sum_X/i;

//cout << alfa2 << " " << beta2;
//getch();

```

```

        i=0;
        Bit_Record >> BRL;
        while (i<BRL)
        {
            Bit_Record >> bit[i] >> Dil[i] >> Dfim1[i] >> T1[i] >> Tac1[i] >>
H11[i] >> H21[i] >> WOB_d_m1[i] >> WOB1[i] >> Freq1[i] >> jato11[i] >> jato21[i] >>
jato31[i];
                i=i+1;
            }

        i=0; j=0;
        while (j < BRL) //atributing the bit record sections values from the
whole well
        {
            while (i <= Dfim1[j]-LDA)
            {
                Di[i] = Dil[j];
                Dfim[i] = Dfim1[j];
                T[i] = T1[j];
                Tac[i] = Tac1[j];
                H1[i] = H11[j];
                H2[i] = H21[j];
                WOB_d_m[i] = WOB_d_m1[j];
                WOB[i] = WOB1[j];
                Freq[i] = Freq1[j];
                jato1[i] = jato11[j];
                jato2[i] = jato21[j];
                jato3[i] = jato31[j];

                Bit_Rec_conf << i << " " << Di[i] << " " << Dfim[i] << " " << T[i]
<< " " << Tac[i] << " " << H1[i] << " " << H2[i] << " " << WOB_d_m[i] << endl;

                i=i+final_depth/data;
            }
            j=j+1; }

        j=j-1;
        m=well_obj;

        if (Dfim[j1] < well_obj1) //extrapolating the last bit record section to
higher depths
        {
            while (i<=well_obj)
            {
                Di[i] = Dil[j];
                Dfim[i] = Dfim1[j];
                T[i] = T1[j];
                Tac[i] = Tac1[j];
                H1[i] = H11[j];
                H2[i] = H21[j];
                WOB_d_m[i] = WOB_d_m1[j];
                WOB[i] = WOB1[j];
                Freq[i] = Freq1[j];
                jato1[i] = jato11[j];
                jato2[i] = jato21[j];
                jato3[i] = jato31[j];

                Bit_Rec_conf << i << " " << Di[i] << " " << Dfim[i] << " "
<< T[i] << " " << Tac[i] << " " << H1[i] << " " << H2[i] << " " << WOB_d_m[i]
<< endl;

                i=i+final_depth/data;
            }
        }

```

```

        } }

// Leitura dos parâmetros de Borgoyne, da abrasividade
j=1;
while (j<=2)
{
i=0;
Borgoyne_input >> abras[j] >> WOBd_t[j];

while (i<8)
{
    Borgoyne_input >> a_borg[j][i];
    i=i+1;
}
j++;
}

M=1, P=N, hear_cont=0, r=1;
i = cond_depth;
rho_lama = 8.5;
d_bite = d_stage[1];
d_rev = ODdp;
k=0; ROPmax=0;
m = cond_depth-final_depth/data;
t_sum_plot[m] = t_desl;

//IMPLEMENTAÇÃO DO MODELO DE PERFURAÇÃO

while (i < comp_sup)
{
    lit = lito[i];
    TVD_ft = (depth[i]-LDA)*3.28;
    rho_dinam = rho_circ(q, rho_lama, d_rev, 5, visc, depth[i]-LDA);

    J2 = ((WOB_d_m[i]-WOB[i]/1000)/d_bite)/(WOB_d_m[i]-
4)*pow((60/Freq[i],H1[i])*(1/(1+0.5*H2[i])));
    hear = root(0.5*H2[i],1,(-1*t_tooth[hear_cont]/(J2*abras[lit])));

    At_bites =
0.25*pi*(pow(jato1[i]/32,2)+pow(jato2[i]/32,2)+pow(jato3[i]/32,2));
    P_bites = 8.311*pow(10,-5)*rho_lama*pow(q,2)/(Cd*At_bites);
    Fj = 0.01823*Cd*q*pow(rho_lama*P_bites,0.5);

    F1 = exp(2.303*a_borg[lit][0]);
    F2 = exp(2.303*a_borg[lit][1]*(10000-TVD_ft));
    F3 = exp(2.303*a_borg[lit][2]*pow(TVD_ft,0.69)*(Gpm[i]-9));
    F4 = exp(2.303*a_borg[lit][3]*TVD_ft*(Gpm[i]-rho_dinam));
    F5 = pow(((WOB[i]/1000)/d_bite)-WOBd_t[lit]/(4-
WOBd_t[lit]),a_borg[lit][4]);
    F6 = pow(Freq[i]/60,a_borg[lit][5]);
    F7 = exp(-1*a_borg[lit][6]*hear);
    //F7 = pow(1/(0.968125*pow(hear,2)+6*hear+1),a_borg[lit][6]);
    F8 = pow((Fj/1000),a_borg[lit][7]);

    ROP[i] = (F1*F2*F3*F4*F5*F6*F7*F8)/3.28;
    if (ROP[i] > ROPmax) ROPmax = ROP[i];

    t[i] = (final_depth/data)/ROP[i];

    if (depth[i]-LDA >= r*30) { t[i] = t[i]+(ts/60); r=r+1; }

    t_tooth[hear_cont] = t_tooth[j]+t[i];
    t_sum[i] = t_sum[i-final_depth/data] + t[i];
}

```



```

t_sum_plot[i] = t_sum_plot[i-final_depth/data] + t[i];

if (t_sum_plot[i] < 0) t_sum_plot[i] = t_sum_plot[i-final_depth/data];

if (hear >= 1.0) { t_trig_sum[k] = t_trig_sum[k-
1]+2*((ts/60)/ls)*depth[i]*3.28; t_trig_hear[hear_cont] =
2*((ts/60)/ls)*depth[i]*3.28;
t_sum_plot[i] = t_sum_plot[i] + 2*((ts/60)/ls)*depth[i]*3.28;
hear_cont=hear_cont+1; k=k+1;}

time_conf << depth[i] << " " << t_sum_plot[i] << " " << t[i] << "
" << ROP[i] << " " << F1 << " " << F2 << " " << F3 << " " << F4 << " " << F5 <<
" " << F6 << " " << F7 << " " << F8 << endl;

i=i+final_depth/data;
}

t_trig[P] = 2*((ts/60)/ls)*depth[i-final_depth/data]*3.28;
t_stage[P] = t_sum[i-final_depth/data];
t_casing[P] = t_cas[0];

hear_cont=hear_cont+1;
bit_depth[hear_cont] = depth[i];
t_trig_hear[hear_cont] = 2*((ts/60)/ls)*depth[i]*3.28;

t_casing[P] = (alfa2 + beta2*depth[i]);
t_drilling = t_drilling + t_stage[P];
t_manobra = t_manobra + t_trig[P];
t_cas_plac = t_cas_plac + t_casing[P];

t_aux = t_sum_plot[i-final_depth/data]+t_casing[P];

if (t_aux > 0) t_sum_plot[i-final_depth/data] = t_aux;
else t_sum_plot[i-final_depth/data] = t_sum_plot[i-final_depth/data];

t_sum[i-final_depth/data] = 0;

i=comp_sup; j=0, k=1;
M=2, P=N-1; depth_trig[0]=comp_sup;

depth_new[0] = well_obj1;

while (i <= well_obj)
{
rho_lama = density[P];
d_bite = d_stage[M];
d_rev = ODdp;

while (i <= depth_new[P]-LDA)
{
TVD_ft = (depth[i]-LDA)*3.28;
rho_dinam = rho_circ(q, rho_lama, d_bite, d_rev, visc, depth[i]-LDA);
lit = lito[i];

J2 = ((WOB_d_m[i]-WOB[i]/1000)/d_bite)/(WOB_d_m[i]-
4)*pow((60/Freq[i]),H1[i])*(1/(1+0.5*H2[i]));
hear = root(0.5*H2[i],1,(-1*t_tooth[hear_cont]/(J2*abras[lit])));

At_bites =
0.25*pi*(pow(jato1[i]/32,2)+pow(jato2[i]/32,2)+pow(jato3[i]/32,2));
P_bites = 8.311*pow(10,-5)*rho_lama*pow(q,2)/(Cd*At_bites);
Fj = 0.01823*Cd*q*pow(rho_lama*P_bites,0.5);

```

```

F1 = exp(2.303*a_borg[lit][0]);
F2 = exp(2.303*a_borg[lit][1]*(10000-TVD_ft));
F3 = exp(2.303*a_borg[lit][2]*pow(TVD_ft,0.69)*(Gpm[i]-9));
F4 = exp(2.303*a_borg[lit][3]*TVD_ft*(Gpm[i]-rho_dinam));
F5 = pow(((WOB[i]/1000)/d_bite)-WOBd_t[lit])/(4-
WOBd_t[lit]),a_borg[lit][4]);
F6 = pow(Freq[i]/60,a_borg[lit][5]);
F7 = exp(-1*a_borg[lit][6]*hear);
//F7 = pow(1/(0.968125*pow(hear,2)+6*hear+1),a_borg[lit][6]);
F8 = pow((Fj/1000),a_borg[lit][7]);

ROP[i] = (F1*F2*F3*F4*F5*F6*F7*F8)/3.28;

if (ROP[i] > ROPmax) ROPmax = ROP[i];

t[i] = (final_depth/data)/ROP[i];

if (depth[i]-LDA >= r*30)
{ t[i] = t[i]+(ts/60); r=r+1; }

t_tooth[hear_cont] = t_tooth[hear_cont]+t[i];
t_sum[i] = t_sum[i-final_depth/data] + t[i];
t_sum_plot[i] = t_sum_plot[i-final_depth/data] + t[i];

if (t_sum_plot[i] < t_sum_plot[i-final_depth/data]) t_sum_plot[i] =
t_sum_plot[i-final_depth/data];
if (t_sum_plot[i] < 0) t_sum_plot[i] = t_sum_plot[i-final_depth/data];

if (hear >= 1.0) { hear_cont=hear_cont+1; t_trig_sum[k] =
t_trig_sum[k-1]+2*((ts/60)/ls)*(depth[i]*3.28); bit_depth[hear_cont] = depth[i];
t_trig_hear[hear_cont] = 2*((ts/60)/ls)*(depth[i]*3.28); t_sum_plot[i]
= t_sum_plot[i] + 2*((ts/60)/ls)*(depth[i]*3.28); k=k+1;}

time_conf << depth[i] << " " << t_sum_plot[i] << " " << t[i] << "
" << ROP[i] << " " << F1 << " " << F2 << " " << F3 << " " << F4 << " " << F5 << "
" << F6 << " " << F7 << " " << F8 << endl;

i=i+final_depth/data;
}
t_trig[P] = 2*((ts/60)/ls)*(depth[i-
final_depth/data]*3.28)+t_trig_sum[k-1];
t_stage[P] = t_sum[i-final_depth/data];

if (Mcas > N) Mcas = CPL-1;
t_casing[P] = t_cas[Mcas];

hear_cont=hear_cont+1;
bit_depth[hear_cont] = depth[i];
t_trig_hear[hear_cont] = 2*((ts/60)/ls)*depth[i]*3.28;

t_casing[P] = alfa2 + beta2*depth[i];
t_drilling = t_drilling + t_stage[P];
t_manobra = t_manobra + t_trig[P];
t_cas_plac = t_cas_plac + t_casing[P];

t_sum_plot[i-final_depth/data] = t_sum_plot[i-
final_depth/data]+t_trig[P]-t_trig_sum[k-1]+t_casing[P];
t_sum[i-final_depth/data] = 0;
P=P-1; M=M+1; k=1; Mcas = Mcas+1;}

ROP_conf.close();
time_conf.close();
Bit_Record.close();

```

```

m = well_obj;

png_start(xpmax,ypmax);
pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmax-30,ypmax-20);
gridding_TIME(well_obj, xpmax, ypmax, t_sum_plot[m], ROPmax);

i=cond_depth-final_depth/data;
ROP[i-final_depth/data] = ROP[i+2*final_depth/data];
ROP[i] = ROP[i+final_depth/data];

while (i<=well_obj)
{
    time_drawing(xpmax,ypmax,ROP[i],ROP[i-
final_depth/data],t_sum_plot[i],t_sum_plot[i-final_depth/data],depth[i]-LDA,depth[i-
final_depth/data]-LDA,well_obj,t_sum_plot[m],ROPmax,1);
    i=i+final_depth/data;
}

a=sem_reg;
while(a <= well_obj)
{
    if (a==sem_reg)
litology_drawing(xpmax,ypmax,lito[a],sem_reg,sem_reg,well_obj);
    else litology_drawing(xpmax,ypmax,lito[a],depth[a]-LDA,depth[a-
final_depth/data]-LDA,well_obj);
    a=a+final_depth/data;
}

png_finish("C:/UFRJDrillingSimulator/Outputs/OutputTimeDepthDiagram.png");

delete t_sum;
delete Di;
delete Dfim;
delete T;
delete Tac;
delete H1;
delete H2;
delete WOB_d_m;
delete WOB;
delete Freq;
delete jato1;
delete jato2;
delete jato3;

//IMPLEMENTAÇÃO DE CUSTOS

cfixo.open("C:/UFRJDrillingSimulator/Inputs/InputFixedCosts.txt");

i=0;
int w=N;

cfixo >> R;

while (i<R)
{
cfixo >> value[i];
i=i+1;
}

double Cqch = value[0];          //Custo de controle de qualidade

```

```

double Cpso = value[1];           //Custo de posicionamento da sonda
double Csaf = value[2];           //Custo de eq. da cabeça de poço
double Cper = value[3];           //Custo de perfis eletricos
double Ctfo = value[4];           //Custo de teste de formação
double Chse = value[5];           //Custo fixo serviços HSE
double Cins = value[6];           //Custo de seguro
double Csle = value[7];           //Custo de serviço legal
double Cwhe = value[8];           //Custo da cabeça de poço
double Ctub = value[9];           //Custo do revestimento
double CBOP_5000 = value[10];     //Custo do BOP 5000
double CBOP_10000 = value[11];    //Custo do BOP 10000
double CBOP_15000 = value[12];    //Custo do BOP 15000
double Ccim = value[13];           //Custo do cimento por bbl
double Cdfl = value[14];           //Custo de materiais para fluido de
perfuração
double Cbit = value[15];           //Custo fixo por broca

cfixo.close();
i=0;

cvar.open("C:/UFRJDrillingSimulator/Inputs/InputVariableCosts.txt");
cvar >> R;

while (i<R)
{
cvar >> value[i];
i=i+1;
}

double C_fixo, C_var;
double Chel = value[0];           //Custo com helicóptero
double Cbap = value[1];           //Custo com a base de apoio
double Cbar = value[2];           //Custo com barcos de apoio
double Cgre = value[3];           //Custo com gerencia de residuos
double Cpte = value[4];           //Custo com previsão climatológicas
double Cdrev = value[5];          //Custo com descida do revestimento
double Ccpl = value[6];           //Custo cimentação
double Crov = value[7];           //Custo com ROV
double Cequ = value[8];           //Custo com equipamentos de perfuração
double Cepe = value[9];           //Custo com equipamentos de pescaria
double Cmlo = value[10];          //Custo com mud logging
double Cewh = value[11];          //Custo com MWD, LWD, etc
double Ctel = value[12];          //Custo com telecomunicações
double Cpes = value[13];          //Custo com pessoal
double Cdie = value[14];          //Custo com diesel
double Cson = value[15];          //Custo aluguel da sonda

cvar.close();

C_fixo = 1.1*(Cqch+Cpso+Csaf+Cper+Ctfo+Chse+Csle+Cins+Cdfl);
C_var =
1.1*(Chel+Cbap+Cbar+Cgre+Cpte+Crov+Cequ+Cepe+Cmlo+Cewh+Ctel+Cpes+Cdie+Cson);

xpmx = 600;
png_start(xpmx,ypmx);
pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(30,20,xpmx-30,ypmx-20);
gridding_FINAL(well_obj,xpmx,ypmx);

fgcolor("black");
moveto(40,ypmx-60); write_string("WELL PROFILE");

```

```

moveto(40,ypmax-45); write_string("SCALE: TVDBML (m)");

double y_pos = 0, x_pos=25-0.5*d_stage[0];
int y_pixel = 20 + y_pos*(ypmax-40)/(final_depth);
int x_pixel = 30 + x_pos*(xpmax-40)/50;
fgcolor("brown");
moveto(20,y_pixel); lineto(x_pixel,y_pixel);
y_pos = 0, x_pos=25+0.5*d_stage[0];
y_pixel = 20 + y_pos*(ypmax-40)/(final_depth);
x_pixel = 20 + x_pos*(xpmax-40)/50;

a=sem_reg;
while(a <= well_obj)
{
if (a==sem_reg)
litology_drawing(xpmax,ypmax,lito[a],sem_reg,sem_reg,well_obj);
else litology_drawing(xpmax,ypmax,lito[a],depth[a]-LDA,depth[a]-
final_depth/data]-LDA,well_obj);
a=a+final_depth/data;
}

Code_output.open("C:/UFRJDrillingSimulator/Outputs/OutputWellPlanning_Results.txt");

z_inicial[u]=0;
cg[u] = 0;
int corr = 100;
int sigma_cond, sigma_sup;

clrscr;
thick[0] =
cond_sup_stress(API[0]*1000,cond_depth,grad_water,d_aco,d_ext[0],1.5);
WPF[0] = 0.25*3.1415*(pow(d_ext[0]/12,2)-pow((d_ext[0]-
2*thick[0])/12,2))*65.5/0.13;

cout << endl << "*****ESTAGIO CONDUTOR*****" <<
endl << endl;
cout << "COTA INICIAL (TVD):" << setiosflags (ios::fixed) <<
setprecision(0) << LDA << " m" << endl;
cout << "COTA FINAL (TVD):" << cond_depth + LDA << " m" << endl <<
endl;
cout << "DENSIDADE DA LAMA:" << setprecision(2) << grad_water << "
ppg" << endl;
cout << "OD DO CASING:" << setprecision(2) << d_ext[0] << " in"
<< endl;
cout << "ESPESSURA DO CASING:" << thick[0] << " in" << endl;
cout << "TIPO DE ACO:" << "API " << API[0] << endl;

Code_output << endl << "*****ESTAGIO
CONDUTOR*****" << endl << endl;
Code_output << "COTA INICIAL (TVD):" << setiosflags (ios::fixed) <<
setprecision(0) << LDA << " m" << endl;
Code_output << "COTA FINAL (TVD):" << cond_depth + LDA << " m" <<
endl << endl;
Code_output << "DENSIDADE DA LAMA:" << setprecision(2) << grad_water
<< " ppg" << endl;
Code_output << "OD DO CASING:" << setprecision(2) << d_ext[0] <<
" in" << endl;
Code_output << "ESPESSURA DO CASING:" << thick[0] << " in" << endl;
Code_output << "TIPO DE ACO:" << "API " << API[0] << endl;

```

```

        drawing(cond_depth,25-0.5*d_ext[0],0,xpmax,ypmax,25-
0.5*d_stage[0],cond_depth,well_obj,thick[0],cond_depth,0,25-0.5*d_ext[0],0,0,25-
0.5*d_ext[v-1]);

drawing(cond_depth,25+0.5*d_ext[0],0,xpmax,ypmax,25+0.5*d_stage[0],cond_depth,well_obj
,thick[0],cond_depth,0,25+0.5*d_ext[0],0,0,25+0.5*d_ext[v-1]);

        contorno_drawing_cond (xpmax,ypmax,25-0.5*d_stage[0],0,25-
0.5*d_ext[0],cond_depth,well_obj);
        contorno_drawing_cond
(xpmax,ypmax,25+0.5*d_stage[0],0,25+0.5*d_ext[0],cond_depth,well_obj);

        if (v==1) {

                thick[1] =
cond_sup_stress(API[1]*1000,cond_depth,grad_water,d_aco,d_ext[1],1.0);
                WPF[1] = 0.25*3.1415*(pow(d_ext[1]/12,2)-pow((d_ext[1]-
2*thick[1])/12,2))*65.5/0.13;

                cout << endl << "*****ESTAGIO 2*****"
<< endl << endl;
                cout << "COTA INICIAL (TVD):          " << setiosflags (ios::fixed) <<
setprecision(0) << LDA << " m" << endl;
                cout << "COTA FINAL (TVD):          " << depth_new[p] << " m" << endl << endl;
                cout << "DENSIDADE DA LAMA:          " << setprecision(2) << grad_water << "
ppg" << endl;
                cout << "DIAMETRO DO ESTAGIO:        " << setprecision(2) << d_stage[1] << " in"
<< endl;
                cout << "OD DO CASING:              " << setprecision(2) << d_ext[1] << " in"
<< endl;
                cout << "ESPESSURA DO CASING:        " << thick[1] << " in" << endl;
                cout << "VOLUME DE CIMENTO:          " << pi/4*(pow(d_stage[1],2)-
pow(d_ext[1],2))*(depth_new[p]-LDA)*0.000625/0.159 << " bbl" << endl;
                cout << "COLUNA DE CIMENTO:          " << depth_new[p]-LDA << " m" << endl;
                cout << "TIPO DE ACO:              API " << API[1] << endl << endl;
                cout << "TEMPO DE PERFURACAO:         " << t_stage[N] << " + " << t_trig[N] << "
h" << endl;

                Code_output << endl << "*****ESTAGIO
2*****" << endl << endl;
                Code_output << "COTA INICIAL (TVD):          " << setiosflags (ios::fixed) <<
setprecision(0) << LDA << " m" << endl;
                Code_output << "COTA FINAL (TVD):          " << depth_new[p] << " m" << endl <<
endl;
                Code_output << "DENSIDADE DA LAMA:          " << setprecision(2) << grad_water
<< " ppg" << endl;
                Code_output << "DIAMETRO DO ESTAGIO:        " << setprecision(2) << d_stage[1]
<< " in" << endl;
                Code_output << "OD DO CASING:              " << setprecision(2) << d_ext[1] <<
" in" << endl;
                Code_output << "ESPESSURA DO CASING:        " << thick[1] << " in" << endl;
                Code_output << "VOLUME DE CIMENTO:          " << pi/4*(pow(d_stage[1],2)-
pow(d_ext[1],2))*(depth_new[p]-LDA)*0.000625/0.159 << " bbl" << endl;
                Code_output << "COLUNA DE CIMENTO:          " << depth_new[p]-LDA << " m" <<
endl;
                Code_output << "TIPO DE ACO:              API " << API[1] << endl << endl;
                Code_output << "TEMPO DE PERFURACAO:         (" << t_stage[N] << " + " <<
t_trig[N] << ") h" << endl;

                drawing(depth_new[N]-LDA,25-0.5*d_ext[0],0,xpmax,ypmax,25-
0.5*d_stage[1],depth_new[p]-LDA,well_obj,thick[1],cond_depth,0,25-
0.5*d_ext[1],0,corr,25-0.5*d_ext[v-1]);

```

```

        drawing(depth_new[N]-
LDA,25+0.5*d_ext[0],0,xpmax,ypmax,25+0.5*d_stage[1],depth_new[p]-
LDA,well_obj,thick[1],cond_depth,0,25+0.5*d_ext[1],0,corr,25+0.5*d_ext[v-1]);

        contorno_drawing(xpmax,ypmax,25-0.5*d_ext[0],0,25-
0.5*d_stage[1],cond_depth,25-0.5*d_ext[1],depth_new[p]-LDA,well_obj);
        contorno_drawing
(xpmax,ypmax,25+0.5*d_ext[0],0,25+0.5*d_stage[1],cond_depth,25+0.5*d_ext[1],depth_new[
p]-LDA,well_obj);

        p=p-1; N=N-1;
    }

    S:
    while (c < count)
    {
        d_ext1 = d_ext[v+1];
        d_stage1 = d_stage[v+1];

        if (v==0) { density[N] = grad_water; z_inicial[u+1]=0;
depth_ant1[u] = 0; cg[u+1] = 0; cim_prof_2[N] = comp_sup;
        thick[1] =
cond_sup_stress(API[1]*1000,cond_depth,grad_water,d_aco,d_ext[1],0.01);
        thick1 = thick[1];
        WPF[1] = 0.25*3.1415*(pow(d_ext[1]/12,2)-pow((d_ext[1]-
2*thick[1])/12,2))*65.5/0.13;
        }

        else {
            thick_teste = casing(API[v+1]*1000,density[N],density[N-
1],depth_new[N],depth_old[y],d_ext1,0.15,coef,d_aco,d_gas);
            thick[v+1] =
collapse(API[v+1]*1000,density[N],d_ext1,thick_teste,depth_new[N],coef,LDA,density[N-
1]);
            thick1 = thick[v+1];
            WPF[v+1] = 0.25*3.1415*(pow(d_ext[v+1]/12,2)-pow((d_ext[v+1]-
2*thick[v+1])/12,2))*65.5/0.13;
            }

        if (c_tipo[v+1] == "CASING" || c_tipo[v+1] == "C") {depth_ant[c] =
LDA; z_inicial[u+1]=0; cg[u+1]=0; depth_ant1[u+1]=depth_old[y+1];

            if (c_tipo[v] == "LINER" || c_tipo[v] == "L") {
                if (c_tipo[v-1] == "LINER" || c_tipo[v-1] == "L") {
                    cim_prof[N] = cemento(WOH,d_ext1,thick1,depth_new[N]-
depth_old[y+2]+corr,WPF[v+1],density[N],d_stage1,depth_new[N+1]-
LDA,d_ext[v],d_cim,d_aco,2);

                    cim_prof_2[N] = cemento_2(depth_new[N],cim_prof[N],d_cim,density[N-
1],LDA);
                    if (cim_prof_2[N] > 800) cim_prof_2[N] = 800; }

                else {
                    cim_prof[N] = cemento(WOH,d_ext1,thick1,depth_new[N]-
depth_old[y+1]+corr,WPF[v+1],density[N],d_stage1,depth_new[N+1]-
LDA,d_ext[v],d_cim,d_aco,2);

                    cim_prof_2[N] = cemento_2(depth_new[N],cim_prof[N],d_cim,density[N-
1],LDA);
                    if (cim_prof_2[N] > 800) cim_prof_2[N] = 800;} }

            else {

```

```

        cim_prof[N] = cimento(WOH,d_ext1,thick1,depth_new[N]-
LDA,WPF[v+1],density[N],d_stagel,depth_new[N+1]-LDA,d_ext[v],d_cim,d_aco,1);

        cim_prof_2[N] = cimento_2(cim_prof[N],depth_new[N],d_cim,density[N-
1],LDA);
        if (cim_prof_2[N] > 800) cim_prof_2[N] = 800;}

    }

    if (c_tipo[v+1] == "LINER" || c_tipo[v+1] == "L") {cg[u+1]=1;

        if (c_tipo[v] == "LINER" || c_tipo[v] == "L") {z_inicial[u+1] =
z_inicial[u]; depth_ant[c] = depth_old[y+1]-corr; depth_ant1[u+1] = depth_old[y+1];
        cim_prof[N] = cimento(WOH,d_ext1,thick1,depth_new[N]-
depth_old[y+1]+corr,WPF[v+1],density[N],d_stagel,depth_new[N+1]-
LDA,d_ext[v],d_cim,d_aco,2);

        cim_prof_2[N] = cimento_2(depth_new[N],cim_prof[N],d_cim,density[N-
1],LDA);
        if (cim_prof_2[N] > 800) cim_prof_2[N] = 800;}

        else {z_inicial[u+1] = depth_old[y]-LDA-corr; depth_ant[c] =
depth_old[y]-corr; depth_ant1[u+1] = depth_old[y];
        cim_prof[N] = cimento(WOH,d_ext1,thick1,depth_new[N]-
depth_old[y]+corr,WPF[v+1],density[N],d_stagel,depth_new[N+1]-
LDA,d_ext[v],d_cim,d_aco,2);

        cim_prof_2[N] = cimento_2(depth_new[N],cim_prof[N],d_cim,density[N-
1],LDA);
        if (cim_prof_2[N] > 800) cim_prof_2[N] = 800;}

        if (v==1) depth_ant1[u+1] = depth_old[y];
    }

    // PBOP_PF[c] = BOP_PF(density[N+1],depth_old[y],srt,coef,d_gas);
    PBOP_CR[c] = BOP_CR(API[v+1]*1000,d_ext1,thick1);
    //Code_output << PBOP_PF[c] << "    " << PBOP_CR[c] << endl;
//Teste definitivi BOP
    qnt_baritina[N] = baritina(density[N-1],density[N]);

    cout << endl << "*****ESTAGIO " << v+2 <<
"*****" << endl << endl;

    if (v!=0) cout << "TIPO DE REV:          " << c_tipo[v+1] << endl
<< endl;
    if (v==0) { cout << "COTA INICIAL (TVD):      " <<
setiosflags(ios::fixed) << setprecision(0) << LDA << " m" << endl; }
    else { cout << "COTA INICIAL (TVD):      " <<
setiosflags(ios::fixed) << setprecision(0) << depth_ant[c] << " m" << endl; }
    cout << "COTA FINAL (TVD):          " << depth_new[N] << " m" << endl
<< endl;
    cout << "DENSIDADE DA LAMA:          " << setprecision(2) << density[N]
<< " ppg" << endl;
    cout << "DIAMETRO DO ESTAGIO:        " << setprecision(2) << d_stagel
<< " in" << endl;
    cout << "OD DO CASING:              " << setprecision(2) << d_ext1 <<
" in" << endl;
    cout << "ESP. MINIMA CASING:            " << setprecision(2) << thick[v+1]
<< " in" << endl;
    cout << "PESO ESP. DO CASING:          " << WPF[v+1] << " lbf/ft" <<
endl;
    cout << "VOLUME DE CIMENTO:          " << setprecision(2) <<
CimV(d_stagel,d_ext1,cim_prof_2[N]) << " bbl" << endl;

```



```

        cout << "NIVEL DE CIMENTO:          " << setprecision(2) <<
cim_prof_2[N] << " m" << endl;
        cout << "TIPO DE ACO:              API " << API[v+1] << endl << endl;
        cout << "TEMPO DE PERFURACAO:      (" << t_stage[N] << " + " <<
t_trig[N] << ") h" << endl;

        Code_output << endl << "*****ESTAGIO " << v+2 <<
"*****" << endl << endl;
        if (v!=0) Code_output << "TIPO DE REV:          " << c_tipo[v+1] <<
endl << endl;
        if (v==0) { Code_output << "COTA INICIAL (TVD):      " <<
setiosflags(ios::fixed) << setprecision(0) << LDA << " m" << endl; }
        else { Code_output << "COTA INICIAL (TVD):          " <<
setiosflags(ios::fixed) << setprecision(0) << depth_ant[c] << " m" << endl; }
        Code_output << "COTA FINAL (TVD):              " << depth_new[N] << " m"
<< endl << endl;
        Code_output << "DENSIDADE DA LAMA:          " << setprecision(2) <<
density[N] << " ppg" << endl;
        Code_output << "DIAMETRO DO ESTAGIO:        " << setprecision(2) <<
d_stagel << " in" << endl;
        Code_output << "OD DO CASING:              " << setprecision(2) <<
d_extl << " in" << endl;
        Code_output << "ESP. MINIMA CASING:         " << setprecision(2) <<
thick[v+1] << " in" << endl;
        Code_output << "VOLUME DE CIMENTO:          " << setprecision(2) <<
CimV(d_stagel,d_extl,cim_prof_2[N]) << " bbl" << endl;
        Code_output << "NIVEL DE CIMENTO:          " << setprecision(2) <<
cim_prof_2[N] << " m" << endl;
        Code_output << "TIPO DE ACO:              API " << API[v+1] << endl
<< endl;
        Code_output << "TEMPO DE PERFURACAO:      (" << t_stage[N] << " + "
<< t_trig[N] << ") h" << endl;

        if (cond_depth!=0) {if (v+2==2) depth_old[y]=cond_depth+LDA;}
        drawing(depth_new[N]-LDA,25-
0.5*d_ext[v],z_inicial[u],xpmax,ypmax,25-
0.5*d_stage[v+1],cim_prof_2[N],well_obj,thick[v+1],depth_old[y]-LDA,depth_ant1[u]-LDA-
corr,25-0.5*d_ext[v+1],cg[u],corr,25-0.5*d_ext[v-1]);
        drawing(depth_new[N]-
LDA,25+0.5*d_ext[v],z_inicial[u],xpmax,ypmax,25+0.5*d_stage[v+1],cim_prof_2[N],well_ob
j,thick[v+1],depth_old[y]-LDA,depth_ant1[u]-LDA-
corr,25+0.5*d_ext[v+1],cg[u],corr,25+0.5*d_ext[v-1]);

        contorno_drawing (xpmax,ypmax,25-0.5*d_ext[v],z_inicial[u],25-
0.5*d_stage[v+1],depth_old[y]-LDA,25-0.5*d_ext[v+1],depth_new[N]-LDA,well_obj);
        contorno_drawing
(xpmax,ypmax,25+0.5*d_ext[v],z_inicial[u],25+0.5*d_stage[v+1],depth_old[y]-
LDA,25+0.5*d_ext[v+1],depth_new[N]-LDA,well_obj);

        N=N-1; y=y-1; c=c+1; v=v+1; u=u+1;
        }

        m = well_obj-final_depth/data;

        double thick_teste_final =
casing(API[v+1]*1000,density[0],Gf[0],final_depth+LDA,depth_new[0]+LDA-
srt,d_ext[v+1],0.15,coef,d_aco,d_gas);
        double thick_final =
collapse(API[v+1]*1000,density[0],d_ext[v+1],thick_teste_final,final_depth+LDA,coef,LDA
,Gf[0]);
        thick[v+1] = thick_final;
        WPF[v+1] = 0.25*3.1415*(pow(d_ext[v+1]/12,2)-pow((d_ext[v+1]-
2*thick_final)/12,2))*65.5/0.13;

```

```

        if (c_tipo[v+1] == "CASING" || c_tipo[v+1] == "C") {depth_final_i = LDA;
z_inicial_final_2=0, cg_final=0; depth_final_2 = well_obj;

        if (c_tipo[v] == "CASING" || c_tipo[v] == "C") {z_inicial_final_1=0;
cg[u]=0; depth_ant[c]=well_obj; depth_ant1[u]=depth_old[1];
        cim_prof_final =
cimento(WOH,d_ext[v+1],thick[v+1],well_obj,WPF[v+1],density[0],d_stage[v+1],depth[N]-
LDA,d_ext[v],d_cim,d_aco,2);

        cim_prof_final_2 = cimento_2(well_obj,cim_prof_final,d_cim,Gf[m],LDA);
if (cim_prof_final_2 > 800) cim_prof_final_2 = 800; }

        if (c_tipo[v] == "LINER" || c_tipo[v] == "L") {cg[u]=1;

        if (c_tipo[v-1] == "LINER" || c_tipo[v-1] == "L") {z_inicial_final_1 =
z_inicial[u]; depth_ant[c]=well_obj;
        cim_prof_final = cimento(WOH,d_ext1,thick1,depth_new[N]-
depth_old[y+2]+corr,WPF[v+1],density[N],d_stage1,depth_new[N+1]-
LDA,d_ext[v],d_cim,d_aco,2);

        cim_prof_final_2 = cimento_2(well_obj,cim_prof_final,d_cim,Gf[m],LDA);
if (cim_prof_final_2 > 800) cim_prof_final_2 = 800; }

        else { z_inicial_final_1 = depth_old[1]-LDA-corr; cg[u]=1;
depth_ant[c]=well_obj;
        cim_prof_final = cimento(WOH,d_ext[v+1],thick[v+1],depth_new[N]-
depth_old[y+1]+corr,WPF[v+1],density[N],d_stage[v+1],depth_new[N+1]-
LDA,d_ext[v],d_cim,d_aco,2);

        cim_prof_final_2 = cimento_2(well_obj,cim_prof_final,d_cim,Gf[m],LDA);
if (cim_prof_final_2 > 800) cim_prof_final_2 = 800; }
} }

        else if (c_tipo[v+1] == "LINER" || c_tipo[v+1] == "L") {cg_final=1;

        if (c_tipo[v] == "CASING" || c_tipo[v] == "C") {z_inicial_final_1 = 0;
depth_ant[c] = well_obj; depth_ant1[u+1] = depth_old[1]; depth_final_2 = well_obj-
depth_old[0]+corr+LDA;
        z_inicial_final_2 = depth_old[0]-LDA-corr; depth_final_i = depth_old[0]-
corr;
        cim_prof_final = cimento(WOH,d_ext[v+1],thick[v+1],well_obj-
depth_old[0]+corr+LDA,WPF[v+1],density[0],d_stage[v+1],depth_new[1]-
LDA,d_ext[v],d_cim,d_aco,2);

        cim_prof_final_2 = cimento_2(well_obj,cim_prof_final,d_cim,Gf[m],LDA);
if (cim_prof_final_2 > 800) cim_prof_final_2 = 800; }

        if (c_tipo[v] == "LINER" || c_tipo[v] == "L") {cg[u]=1;

        if (c_tipo[v-1] == "LINER" || c_tipo[v-1] == "L") {z_inicial_final_1 =
z_inicial[u]; depth_ant[c]=well_obj; depth_final_2 = well_obj-
depth_old[y+2]+corr+LDA;
        z_inicial_final_2 = depth_old[1]-LDA-corr; depth_final_i = depth_old[1]-
corr;
        cim_prof_final = cimento(WOH,d_ext1,thick1,well_obj-
depth_old[y+2]+corr+LDA,WPF[v+1],density[N],d_stage1,depth_new[N+1]-
LDA,d_ext[v],d_cim,d_aco,2);

        cim_prof_final_2 = cimento_2(well_obj,cim_prof_final,d_cim,Gf[m],LDA);
if (cim_prof_final_2 > 800) cim_prof_final_2 = 800; }

```

```

else { z_inicial_final_1 = depth_old[1]-LDA-corr; cg[u]=1;
depth_ant[c]=well_obj; depth_final_2 = well_obj-depth_old[y+1]+corr+LDA;
z_inicial_final_2 = depth_old[0]-LDA-corr; depth_final_i = depth_old[0]-
corr;
cim_prof_final = cimento(WOH,d_ext[v+1],thick[v+1],well_obj-
depth_old[y+1]+corr+LDA,WPF[v+1],density[N],d_stage[v+1],depth_new[N+1]-
LDA,d_ext[v],d_cim,d_aco,2);

cim_prof_final_2 = cimento_2(well_obj,cim_prof_final,d_cim,Gf[m],LDA);
if (cim_prof_final_2 > 800) cim_prof_final_2 = 800;}
} }

// PBOP_PF[c] = BOP_PF(density[N+1],depth_old[y],srt,coef,d_gas);
PBOP_CR[c] = BOP_CR(API[v+1]*1000,d_ext[v+1],thick[v+1]);
//Code_output << PBOP_PF[c] << " " << PBOP_CR[c] << endl; //Teste
definitivi BOP
cout << endl << "*****ESTAGIO " << v+2 <<
"*****" << endl << endl;
cout << "COTA INICIAL (TVD): " << setprecision(0) << depth_final_i << "
m" << endl;
cout << "COTA FINAL (TVD): " << well_obj1 << " m" << endl << endl;
cout << "DENSIDADE DA LAMA: " << setprecision(2) << density[0] << "
ppg" << endl;
cout << "DIAMETRO DO ESTAGIO: " << setprecision(2) << d_stage[v+1] << "
in" << endl;
cout << "OD DO CASING: " << setprecision(2) << d_ext[v+1] << " in"
<< endl;
cout << "ESP. MINIMA CASING: " << setprecision(2) << thick_final << "
in" << endl;
cout << "VOLUME DE CIMENTO: " << setprecision(2) <<
CimV(d_stage[v+1],d_ext[v+1],cim_prof_final_2) << " bbl" << endl;
cout << "NIVEL DE CIMENTO: " << setprecision(2) << cim_prof_final_2 <<
" m" << endl;
cout << "TEMPO DE PERFURACAO: " << t_stage[0] << " h" << endl;
cout << "TIPO DE ACO: API " << API[v+1] << endl << endl;
cout << "TEMPO DE PERFURACAO: (" << t_stage[0] << " + " << t_trig[0] <<
") h" << endl << endl;

Code_output << endl << "*****ESTAGIO " << v+2 <<
"*****" << endl << endl;
Code_output << "COTA INICIAL (TVD): " << setprecision(0) <<
depth_final_i << " m" << endl;
Code_output << "COTA FINAL (TVD): " << well_obj1 << " m" << endl <<
endl;
Code_output << "DENSIDADE DA LAMA: " << setprecision(2) << density[0]
<< " ppg" << endl;
Code_output << "DIAMETRO DO ESTAGIO: " << setprecision(2) << d_stage[v+1]
<< " in" << endl;
Code_output << "OD DO CASING: " << setprecision(2) << d_ext[v+1]
<< " in" << endl;
Code_output << "ESP. MINIMA CASING: " << setprecision(2) << thick_final
<< " in" << endl;
Code_output << "VOLUME DE CIMENTO: " << setprecision(2) <<
CimV(d_stage[v+1],d_ext[v+1],cim_prof_final_2) << " bbl" << endl;
Code_output << "NIVEL DE CIMENTO: " << setprecision(2) <<
cim_prof_final_2 << " m" << endl;
Code_output << "TEMPO DE PERFURACAO: " << t_stage[0] << " h" << endl;
Code_output << "TIPO DE ACO: API " << API[v+1] << endl << endl;
Code_output << "TEMPO DE PERFURACAO: (" << t_stage[0] << " + " <<
t_trig[0] << ") h" << endl << endl;

```

```

        drawing(well_obj,25-0.5*d_ext[v],z_inicial_final_1,xpmax,ypmax,25-
0.5*d_stage[v+1],cim_prof_final_2,well_obj,thick_final,depth_old[0]-LDA,depth_ant1[u]-
LDA-corr,25-0.5*d_ext[v+1],cg[u],corr,25-0.5*d_ext[v-1]);

drawing(well_obj,25+0.5*d_ext[v],z_inicial_final_1,xpmax,ypmax,25+0.5*d_stage[v+1],cim
_prof_final_2,well_obj,thick_final,depth_old[0]-LDA,depth_ant1[u]-LDA-
corr,25+0.5*d_ext[v+1],cg[u],corr,25+0.5*d_ext[v-1]);

        contorno_drawing (xpmax,ypmax,25-0.5*d_ext[v],z_inicial_final_1,25-
0.5*d_stage[v+1],depth_old[0]-LDA,25-0.5*d_ext[v+1],well_obj,well_obj);
        contorno_drawing
(xpmax,ypmax,25+0.5*d_ext[v],z_inicial_final_1,25+0.5*d_stage[v+1],depth_old[0]-
LDA,25+0.5*d_ext[v+1],well_obj,well_obj);

        drawing_final_casing(depth_final_2,25-
0.5*d_ext[v+1],z_inicial_final_2,xpmax,ypmax,thick_final,well_obj,corr,cg_final,25-
0.5*d_ext[v]);

drawing_final_casing(depth_final_2,25+0.5*d_ext[v+1],z_inicial_final_2,xpmax,ypmax,thi
ck_final,well_obj,corr,cg_final,25+0.5*d_ext[v]);

        PBOP_PP = BOP_PP(Gp_max,Dh,srt,coef,d_gas);
        PBOP_LDA = BOP_LDA(Gp_max,Dh,srt,coef,d_gas,LDA,grad_water);

        i=0;
        while (i<c)
        {
                // if (PBOP_PF[i] > PBOP_PF_max) PBOP_PF_max = PBOP_PF[i];
                if (PBOP_CR[i] > PBOP_CR_max) PBOP_CR_max = PBOP_CR[i];
                i=i+1;
        }

        Code_output << PBOP_PP << " " << PBOP_LDA << " " << PBOP_PF_max << " " <<
PBOP_CR_max << endl;

        i=0;
        double PBOP = BOP(PBOP_PP,PBOP_LDA,PBOP_PF_max,PBOP_CR_max);
        double t_total = t_drilling+t_manobra+t_cas_plac+t_desl;

        cout << "*****DISCRETIZACAO DO TEMPO DE
PERFURACAO*****" << endl << endl;
        Code_output << "*****DISCRETIZACAO DO TEMPO DE
PERFURACAO*****" << endl << endl;

        cout << "TEMPO TOTAL EM: " << endl;
        cout << "          - PERFURACAO: " << t_drilling << " h - " <<
setprecision(2) << t_drilling/t_total << endl;
        cout << "          - MANOBRA: " << t_manobra << " h - " <<
setprecision(2) << t_manobra/t_total << endl;
        cout << "          - ASS. CASING: " << t_cas_plac << " h - " <<
setprecision(2) << t_cas_plac/t_total << endl;
        cout << "          - DESLOCAMENTO: " << t_desl << " h - " <<
setprecision(2) << t_desl/t_total << endl;
        cout << endl;
        cout << "          - TOTAL: " << t_total << " h - " <<
t_total/t_total << endl << endl;
        cout << "*****" << endl <<
endl;

        Code_output << "TEMPO TOTAL EM: " << endl;
        Code_output << "          - PERFURACAO: " << t_drilling << " h - " <<
setprecision(2) << t_drilling/t_total << endl;

```

```

Code_output << "          - MANOBRA:          " << t_manobra << " h - " <<
setprecision(2) << t_manobra/t_total << endl;
Code_output << "          - ASS. CASING:    " << t_cas_plac << " h - " <<
setprecision(2) << t_cas_plac/t_total << endl;
Code_output << "          - DESLOCAMENTO:    " << t_desl << " h - " <<
setprecision(2) << t_desl/t_total << endl;
Code_output << endl;
Code_output << "          - TOTAL:          " << t_total << " h - " <<
setprecision(2) << t_total/t_total << endl << endl;
Code_output << "*****" <<
endl << endl;

while (i<hear_cont)
{
    if (bit_depth[i+1]==well_obj1) goto H;
    if (i+1==hear_cont) bit_depth[i+1]=well_obj1;
    m = bit_depth[i+1]-LDA;

    cout << "BIT No.: " << setw(4) << setprecision(0) << i+1 << "    DEPTH: " <<
    setw(6) << bit_depth[i+1] << " m    BIT TIME: " << setw(6) << setprecision(2) <<
    t_tooth[i] << " h    ACC. TIME: " << setw(7) << t_sum_plot[m] << " h" << endl;
    Code_output << "BIT No.: " << setw(4) << setprecision(0) << i+1 << "    DEPTH:
" << setw(6) << bit_depth[i+1] << " m    BIT TIME: " << setw(6) << setprecision(2) <<
    t_tooth[i] << " h    ACC. TIME: " << setw(7) << t_sum_plot[m] << " h" << endl;
    i=i+1;
}

H:
cout << endl; Code_output << endl;
cout << "*****DIMENSIONAMENTO DO BOP*****"
<< endl << endl;
Code_output << "*****DIMENSIONAMENTO DO
BOP*****" << endl << endl;

if (PBOP==0) {cout << "Solicitacao supera disponibilidade de BOP no mercado!" <<
endl << endl;
Code_output << "Solicitacao supera disponibilidade de BOP no
mercado!" << endl; }
else {cout << "PRESSAO MAXIMA DO BOP " << setprecision(0) << PBOP << "
psi" << endl << endl;
Code_output << "PRESSAO MAXIMO DO BOP " << setprecision(0) << PBOP <<
" psi" << endl << endl; }

G_total_input.close();
Code_output.close();
png_finish("C:/UFRJDrillingSimulator/Outputs/OutputWellProfile.png");

//IMPLEMENTACAO DOS CUSTOS
ofstream check_custo;

check_custo.open("C:/UFRJDrillingSimulator/Debug/DebugCosts.txt");

i=0; j=0; k=1;
double C_BOP, Cmom=0, Cmom_i=0, cim_vol, *C_sum_plot;
C_sum_plot = new double[well_obj_data];
C_sum_plot[0] = C_fixo;

while (i <= well_obj)
{
    if (depth[i] == bit_depth[j+1]) {Cmom = Cmom_i + C_var*t_trig_hear[j]/24 +
Cbit; Cmom_i=Cmom; j=j+1;}
    if (depth[i] == comp_sup)
        { cim_vol = CimV(d_stage[1],d_ext[1],cim_prof_2[1]);

```

```

        if (PBOP == 5000) C_BOP = CBOP_5000;
        if (PBOP == 10000) C_BOP = CBOP_10000;
        if (PBOP == 15000) C_BOP = CBOP_15000;
        Cmom = Cmom_i + C_BOP + Cwhe + (Cdrev*(alfa2 + beta2*depth[i])/24) +
(Ccpl*(alfa2 + beta2*depth[i])/24) + cim_vol*Ccim;
        Cmom_i = Cmom;
        w=w-1; k=k+1;
    }
    if (depth[i] == depth_new[w])
    { cim_vol = CimV(d_stage[k],d_ext[k],cim_prof_2[w]);
      Cmom = Cmom_i + (Cdrev*(alfa2 + beta2*depth[i])/24) + (Ccpl*(alfa2 +
beta2*depth[i])/24) + cim_vol*Ccim;
      Cmom_i = Cmom;
      w=w-1; k=k+1;
    }

    if (i > 0)
    C_sum_plot[i] = C_sum_plot[i-final_depth/data] + C_var*t[i]/24 + Cmom;

    check_custo << i+LDA << "    " << C_fixo << "    " << C_var << "    " << Cmom
<< "    " << C_sum_plot[i] << endl;
    Cmom=0; Cmom_i=0;
    i=i+final_depth/data;
}
m = well_obj-final_depth/data;

xpmax = 1028;
ypmax = 768;

png_start(xpmax,ypmax);

pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(20,20,xpmax-20,ypmax-20);
fgcolor("black");

gridding_COST(well_obj, xpmax, ypmax, C_sum_plot[m]);

i=final_depth/data;
while (i<=well_obj)
{
    cost_drawing(xpmax,ypmax,C_sum_plot[i],C_sum_plot[i-
final_depth/data],depth[i]-LDA,depth[i-final_depth/data]-LDA,well_obj,C_sum_plot[m]);
    i=i+final_depth/data;
}

png_finish("C:/UFRJDrillingSimulator/Outputs/OutputCostDepthDiagram.png");
check_custo.close();

//TOP VISION DRAWING

xpmax=700;
ypmax=700;
png_start(xpmax,ypmax);

pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(20,20,xpmax-20,ypmax-20);
fgcolor("black");
moveto(30,40); write_string("WELL TOP VIEW");

```

```

y_pos = 0, x_pos=12;
y_pixel = 20 + y_pos*(ypmax-40)/(final_depth);
x_pixel = 20 + x_pos*(xpmax-40)/50;

b=0; c=0;

circle(25,0.5*(well_obj),xpmax,ypmax,d_stage[0],1,final_depth,0.5,1);

while (c<=p)
{
d_ext1 = d_ext[b];
thick1 = thick[b];
circle(25,0.5*(well_obj),xpmax,ypmax,d_ext1,1,final_depth,thick1,0);
c=c+1; b=b+1;
}
circle(25,0.5*(well_obj),xpmax,ypmax,d_ext[b],2,final_depth,thick[b],0);

png_finish("C:/UFRJDrillingSimulator/Outputs/OutputWellTopView.png");

cout << "*****FIM DO PROJETO DE POCO*****" <<
endl << endl;
Code_output << "*****FIM DO PROJETO DE POCO*****";

delete C_sum_plot;
delete depth;
//delete WOB;

cout << " >>>> Digite qualquer botão para sair";
getch();
goto MAIN;
}
//FIM DO PROJETO E PLANEJAMENTO DE POÇO

if (opcao_main==2)
{
//INICIO DO MÓDULO DE TEMPO REAL

double q, Head, Nref, Pb1, Pb2, BHP, depth, depth_aux, *depth_aux1, dp_height,
Pf_dp1, Pf_hwdp1, Pf_dc1, Pf_dp2, Pf_hwdp2, Pf_dc2, *depth_aux2;
double dPf_dp1, dPf_dp2, dPf_hwdp1, dPf_hwdp2, dPf_dc1, dPf_dc2, *Pt_aux1,
*Pt_aux2, Pf_aux, P_bh1, P_bh2, P_bite, Psup, Pi, Psup_aux, ECD;
double time_bite, ff, time_acc = 0, *Gp, ROPs=0, downtime, time_casing,
time_total, Vmud, V_anulus, V_well, Awell, Gpml, HP, timestep, *Gfm, ff2;
double Aanulus, casing_depth_i, casing_depth_f, Hcim, t_rev, time_setting,
Vcasing, vel_casing, WOH1, Hmud, Vcoluna, timestep2, depth_ass, t_trip;
double WOBd, WOB2, ODwell, *t_sum_simul, time_ant, time_simul, depth_simul_ant,
*t_plot, *depth_plot, *ECD_plot, *density_plot, *ROP_plot;
double timein, depthin, P_bh2in, ECDin, HPin, Piin, WOHin, Vmudin, hearin, Gpin,
Gfin, qin, litin, depth_inicial, fract_volume, well_control_time;
int N, opcao_sim, depth_int, lit, well_obj_int, counter, counter_aux, x=0, y=0,
z=0;

double Pe, re, viscgas, hreserv, kperm, qgas, Vkick=0, Vpit, Hsecao, Pp,
time_bite_ant, tt, ROPant, Vperdas, area_fract, Awell1, Awell2;
double Ariser, Afase, Arev, IDrev, depth_casing, depth_fase, Vwell, time_bite2;
char ch, ch2;
string dp_cadast, litoName;
pipe dp, dc, hwdp;
fluid mud;
pump bomba;
bite bite;
 revesti mento rev;
ofstream p_teste, time_conf, simul_results, Gp_conf, casing_data, loop_conf,
plot_conf, report, Vmud_conf;

```

```

ifstream simul_input, Borgoyne_input, main_input;

main_input.open("C:/UFRJDrillingSimulator/Inputs/InputRealTime_Main.txt");
i=0;
main_input >> R;

while (i<R)
{
    main_input >> value[i];
    i=i+1;
}

timestep = value[0];           //time_step da simulação
ff = value[1];                 //fast_forward da simulação
timestep2 = value[2];         //time_step do assentamento da sapata
area_fract = value[3];       //area da fratura em m3
well_control_time = value[4]; //tempo do controle de poço
kperm = value[5];             //permeabilidade do reservatorio
viscgas = value[6];          //viscosidade do gas
hreserv = value[7];           //net-pay
re = value[8];                //raio do reservatorio
Vpit = value[9];              //volume de lama nos tanques iniciais

hreserv = hreserv*3.28;
re = re*3.28*12;

k = 0;
time_bite = 0;
//ff = 1000;
downtime = t_desl; //downtime já começa com tempo de deslocamento e assentamento
do condutor
well_obj_int = well_obj;
//fract_volume = 10;
//well_control_time = 20;
depth = LDA+cond_depth;

Pt_aux1 = new double[1000];
Pt_aux2 = new double[1000];
depth_aux1 = new double[1000];
depth_aux2 = new double[1000];
Gp = new double[6000];
Gfm = new double[6000];

simul_results.open("C:/UFRJDrillingSimulator/Outputs/OutputRealTime_Results.txt");
casing_data.open("C:/UFRJDrillingSimulator/Outputs/OutputRealTime_CasingPlacement.txt"
);
time_conf.open("C:/UFRJDrillingSimulator/Debug/DebugTime.txt");

Borgoyne_input.open("C:/UFRJDrillingSimulator/Inputs/InputBorgoyneParameters.txt");
report.open("C:/UFRJDrillingSimulator/Outputs/OutputRealTime_Report.txt");
Vmud_conf.open("C:/UFRJDrillingSimulator/Debug/DebugVmud.txt");
loop_conf.open("C:/UFRJDrillingSimulator/Debug/DebugBHP_conf.txt");

depth_int=0; j=0; i=1;

//gerando um array de Gp e Gf para modelo de ROP

while (j <= well_obj) //attributing the bit record
sections values from the whole well
{
    while (depth_int < i*final_depth/data)
    {

```



```

        Gp[depth_int] = Gpm[j];
        Gfm[depth_int] = Gf[j];
        lito[depth_int] = lito[j];
        depth_int = depth_int + 1;
    }

    i=i+1;
    j=j+final_depth/data;}
    Gp[depth_int] = Gpm[depth_int];

delete Gpm;
delete Gf;
t_plot = new double[6000];
depth_plot = new double[6000];
ECD_plot = new double[6000];
density_plot = new double[6000];
ROP_plot = new double[6000];
time_total = downtime;

// Leitura dos parâmetros de Borgoyne, da abrasividade
j=1;
while (j<=2)
{
    i=0;
    Borgoyne_input >> abras[j] >> WOBd_t[j];

    while (i<8)
    {
        Borgoyne_input >> a_borg[j][i];
        i=i+1;
    }
    j++;
}

report << " ***** RELATORIO DE PERFURACAO
*****" << endl << endl;

SIM_SCREEN:
clrscr;
opcao_sim = tela_simulacao();

if (opcao_sim == 1) { FLUID:
    clrscr;
    mud = start_fluid();
    if (mud.type == "Newtonian") report << time_total << " h   "
<< depth << " m   FLUIDO: " << mud.type << ";   DENSIDADE = " << mud.density << " ppg;
VISCOSIDADE = " << mud.visc << " cp;   VAZAO = " << mud.q << " gal/min" << endl;

        if (mud.type == "Bingham") report << time_total << " h   "
<< depth << " m   FLUIDO: " << mud.type << ";   DENSIDADE = " << mud.density << " ppg;
VISCOSIDADE PLASTICA = " << mud.p_visc << " cp;   TENSAO LIMITE = " << mud.T_0 << "
lbf/100ft;   VAZAO = " << mud.q << " gal/min" << endl;
        if (mud.type == "Power_Law") report << time_total << " h   "
<< depth << " m   FLUIDO: " << mud.type << ";   DENSIDADE = " << mud.density << " ppg;
PARAMETRO N = " << mud.n << ";   PARAMETRO DE CONSISTENCIA = " << mud.K << " eq cp;
VAZAO = " << mud.q << " gal/min" << endl;

        goto SIM_SCREEN;
    }

if (opcao_sim == 2) { PIPE:
    clrscr;

```

```

        cout << "Deseja cadastrar o drill pipe [sim/nao]? "; cin >>
dp_cadast;

        if (dp_cadast == "sim") { clrscr; dp = start_dp();
report << time_total << " h " << depth << " m DP: OD = "
<< dp.OD << " in; ID = " << dp.ID << " in; SECOES DE " << dp.lenght << " m; PESO
ESPECIFICO = " << dp.razao << " lbf/ft" << endl;
        }

        else {
clrscr;
cout << "*****DRILL PIPE
PARAMETERS*****" << endl << endl;
cout << "DP OUT DIAMETER [in]: " << dp.OD << endl;
cout << "DP THICKNESS [in]: " << dp.thick << endl;
cout << "DP SECTION LENGHT [m] " << dp.lenght <<
endl; //utilizar 3 strings = 30m
cout << "DP WEIGHT/LENGHT [lbf/ft]: " << dp.razao << endl;
cout << endl; }

        hwdp = start_hwdp();
report << time_total << " h " << depth << " m HWDP: OD = "
" << hwdp.OD << " in; ID = " << hwdp.ID << " in; SECOES DE " << hwdp.lenght << " m;
PESO ESPECIFICO = " << hwdp.razao << " lbf/ft; " << hwdp.n << " SECOES" << endl;

        dc = start_dc();
report << time_total << " h " << depth << " m DC: OD = "
<< dc.OD << " in; ID = " << dc.ID << " in; SECOES DE " << dc.lenght << " m; PESO
ESPECIFICO = " << dc.razao << " lbf/ft; " << dc.n << " SECOES" << endl;

        goto SIM_SCREEN;
        }

        if (opcao_sim == 3) { BIT:
clrscr;
bite = start_bite();
if (bite.type == "inicio") { time_bite2 = 0; }
if (bite.type == "nova") { time_bite2 = 0; }

        depth = depth;
WOHi = WOH;
timestep2 = 20;
t_trip = 0;
tt = (ts/60)/dp.lenght)*depth;

TRIP_comput:

        clrscr;

        Hmud = depth;
P_bh2 = 0.17*Hmud*mud.density;
ECD = P_bh2/(0.17*depth);
Pi = 0;
HP = 0;

        if (t_trip <= tt)
WOH = (WOHi+WOB2)*(depth-
dp.lenght/(ts/60)*t_trip)/depth;
        if (t_trip > tt)
WOH = (WOHi+WOB2)*(dp.lenght/(ts/60)*(t_trip-tt))/depth;
        if (WOH < 0) WOH = 0;

```

```

        hear = hear;
        ROPs = 0;

        //simul_results << "trip      " << " " << time_total - t_desl
<< " " << depth << " " << P_bh2 << " " << ECD << " " << Pi << " " << HP;
        //simul_results << " " << WOH << " " << Vmud << " " <<
hear*100 << " " << ROPs << " " << 0;
        //simul_results << " " << Gp[depth_int] << " " <<
Gfm[depth_int] << " " << lito[depth_int] << endl;

        simul_results << "trip      " << setiosflags(ios::fixed) <<
setprecision(2) << setw(7) << time_total << setprecision(2) << setw(8) << depth <<
setprecision(2) << setw(9) << P_bh2;
        simul_results << setprecision(2) << setw(7) << ECD <<
setprecision(2) << setw(9) << Pi << setprecision(2) << setw(8) << HP;
        simul_results << setprecision(2) << setw(7) << ROPs <<
setprecision(0) << setw(5) << 0 << setprecision(2) << setw(7) << WOH/1000 <<
setprecision(2) << setw(9) << Vmud << setprecision(2) << setw(7) << hear*100;
        simul_results << setprecision(2) << setw(12) << Vpit <<
setprecision(2) << setw(7) << Vperdas << setprecision(2) << setw(7) <<
x*area_fract*ROPant*(timestep/60)/.159 << setprecision(2) << setw(7) << Vkick;
        simul_results << setprecision(2) << setw(6) <<
Gp[depth_int] << setprecision(2) << setw(6) << Gfm[depth_int] << setprecision(2) <<
setw(3) << lit << endl;

        t_trip = t_trip + timestep2/60;
        downtime = downtime + timestep2/60;
        time_total = time_acc + downtime;

        if (t_trip <= 2*tt) goto TRIP_comput;
    }

    clrscr;

        report << time_total << " h      " << depth << " m      BROCA: OD
= " << bite.OD << " in; OD JATOS = " << bite.n1 << "/32 in; " << bite.n2 << "/32 in;
" << bite.n3 << "/32 in; ROTACAO = ";
        report << bite.N << " rpm; H1 = " << bite.H1 << "; H2 = "
<< bite.H2 << "; (WOB/d)max = " << bite.WOBdm << endl;

        goto SIM_SCREEN;
    }

    if (opcao_sim == 4) { CASING:
        clrscr;
        cout << "*****CASING SETTING
PARAMETERS*****" << endl << endl;
        rev = start_casing();
        //timestep2 = 150;

        Aanus = 0.25*pi*(pow(bite.OD,2)-
pow(rev.OD,2))*(0.000645);
        Hcim = rev.Vcim*0.159/Aanus; //m
        time_casing = alfa2 + beta2*depth;

        if (rev.type == "CASING" || rev.type == "C")
{casing_depth_i = LDA; casing_depth_f = depth; Vcasing = rev.A*(casing_depth_f -
casing_depth_i);}

        if (rev.type == "LINER" || rev.type == "L") {casing_depth_i
= casing_depth_f - 100; casing_depth_f = depth; Vcasing = rev.A*(casing_depth_f -
casing_depth_i);}

```

```

casing_data << casing_depth_i << " " << casing_depth_f <<
" " << rev.OD << " " << rev.thick << " " << Hcim << endl;

//definição dos parametros relevantes ao longo do
assentamento da sepata
depth = depth;
t_rev = 0;
tt = ((ts/60)/dp.lenght)*depth;
//ff2 = 4000;

REV_comput:

clrscr;
time_setting = time_casing - 2*tt;
Hmud = depth;
P_bh2 = 0.17*Hmud*mud.density;
ECD = P_bh2/(0.17*depth);
Pi = 0;
HP = 0;

if (t_rev <= tt || t_rev >= time_setting + tt)
{
if (t_rev <= tt)
WOH = (WOHi+WOB2)*(depth-dp.lenght/(ts/60)*t_rev)/depth;
if (t_rev > time_setting + tt)
WOH = (WOHi+WOB2)*(dp.lenght/(ts/60)*(t_rev -
time_setting - tt))/depth;
if (WOH < 0) WOH = 0;
}
else WOH = 0;

hear = hear;
ROPs = 0;

simul_results << "casing " << setiosflags(ios::fixed) <<
setprecision(2) << setw(7) << time_total << setprecision(2) << setw(8) << depth <<
setprecision(2) << setw(9) << P_bh2;
simul_results << setprecision(2) << setw(7) << ECD <<
setprecision(2) << setw(9) << Pi << setprecision(2) << setw(8) << HP;
simul_results << setprecision(2) << setw(7) << ROPs <<
setprecision(0) << setw(5) << 0 << setprecision(2) << setw(7) << WOH/1000 <<
setprecision(2) << setw(9) << Vmud << setprecision(2) << setw(7) << hear*100;
simul_results << setprecision(2) << setw(12) << Vpfit <<
setprecision(2) << setw(7) << Vperdas << setprecision(2) << setw(7) <<
x*(area_fract*ROPant*timestep) << setprecision(2) << setw(7) << Vkick;
simul_results << setprecision(2) << setw(6) <<
Gp[depth_int] << setprecision(2) << setw(6) << Gfm[depth_int] << setprecision(2) <<
setw(3) << lit << endl;

t_rev = t_rev + timestep2/60;
downtime = downtime + timestep2/60;
time_total = time_acc + downtime;

if (t_rev <= time_casing) goto REV_comput;

clrscr;

downtime = downtime + timestep2/60;
time_total = time_acc + downtime;

report << time_total << " h " << depth << " m
ASSENTAMENTO DO REVESTIMENTO: " << rev.type << "; OD = " << rev.OD << " in; ID = "
<< rev.ID << " in; GRAU DO ACO = ";

```

```

report << rev.API << " COLUNA DE CIMENTO = " << Hcim << "
m" << endl;

goto SIM_SCREEN;
}

if (opcao_sim == 5) { COMPLET:
clrscr;

Vpit = Vpit - Vperdas;
Vperdas = 0;
x = 0;

report << time_total << " h " << depth << " m ****
COMPLETADO O VOLUME DE LAMA NO POCO *****" << endl;

goto FLUID; }

if (opcao_sim == 6) { WELL_CONTROL:
clrscr;
Vkick = 0;
downtime = downtime + well_control_time;

report << time_total << " h " << depth << " m **** KICK
CIRCULADO *****" << endl;

goto FLUID; }

if (opcao_sim == 7) { clrscr; goto SIM_return; }

if (opcao_sim == 8) { clrscr;

depth = cond_depth+LDA;
depth_int = depth-LDA;
//timestep = 30;
casing_depth_f = cond_depth+LDA;
time_total = time_acc + downtime;

MAIN_simulator:

ROPant = ROPs;
Vmud = 0;
if (time_bite2 == 0) time_bite = 0;

//calculando o volume do poço
if (rev.ID < 1.0) { IDrev = 30; casing_depth_f =
cond_depth+LDA; }

else IDrev = rev.ID;

Ariser = 0.25*pi*pow(ODriser,2)*(0.000625);
Afase = 0.25*pi*pow(bite.OD,2)*(0.000625);
Arev = 0.25*pi*pow(IDrev,2)*(0.000625);
depth_casing = casing_depth_f - LDA;
depth_fase = depth - depth_casing - LDA;

Vwell = LDA*Ariser + depth_casing*Arev +
depth_fase*Afase;

//discrtizar os DP, HWDP e DC em 2 zonas para cada
secao

dp_height = depth - hwdp.height - dc.height;
Vcoluna = dp_height*dp.Ai + hwdp.height*hwdp.Ai +
dc.height*dc.Ai;

```

```

HID_begin:
//cout << "ready";
Pi = Pb2 + Psup_aux;

//P_bh por dentro da coluna
i=0;
//para os elementos discretizados no drill-pipe
if (mud.type == "Newtonian") dPf_dp1 =
newtonian_pipe(mud.q,mud.density,dp.ID,mud.visc);
if (mud.type == "Bingham") dPf_dp1 =
bingham_pipe(mud.q,mud.density,dp.ID,mud.T_0,mud.p_visc);
if (mud.type == "Power_Law") dPf_dp1 =
power_pipe(mud.q,mud.density,dp.ID,mud.K,mud.n);

//calculo da pressão em diferetes pontos para
verificar perfil de pressão ao longo do poço
depth_aux = depth_inicial; //garantindo que haverá
redução no nível de fluido

while (depth_aux<=dp_height)
{
Pf_aux = dPf_dp1*depth_aux*3.28;
Pt_aux1[i] = 0.17*depth_aux*mud.density + Pi -
Pf_aux;

depth_aux1[i]=depth_aux;
depth_aux=depth_aux+dp_height/100;

//loop_conf << "dp " << depth_aux << " " <<
depth_aux1[i] << endl;

if (depth_aux < depth_aux1[i]) {depth_aux =
depth_aux1[i]; goto Label1;} //forcando saido do loop
i=i+1;
}
Pf_dp1 = dPf_dp1*dp_height*3.28;

Label1:
//para os elementos discretizados no heave-weight
drill pipe
if (mud.type == "Newtonian") dPf_hwdp1 =
newtonian_pipe(mud.q,mud.density,hwdp.ID,mud.visc);
if (mud.type == "Bingham") dPf_hwdp1 =
bingham_pipe(mud.q,mud.density,hwdp.ID,mud.T_0,mud.p_visc);
if (mud.type == "Power_Law") dPf_hwdp1 =
power_pipe(mud.q,mud.density,hwdp.ID,mud.K,mud.n);

while (depth_aux<=dp_height+hwdp.height)
{
Pf_aux = dPf_hwdp1*(depth_aux-dp_height)*3.28;
Pt_aux1[i] = 0.17*depth_aux*mud.density + Pi -
(Pf_dp1+Pf_aux);

depth_aux1[i]=depth_aux;
depth_aux=depth_aux+hwdp.height/10;

//loop_conf << "hwdp " << depth_aux << " " <<
depth_aux1[i] << endl;

if (depth_aux < depth_aux1[i]) {depth_aux =
depth_aux1[i]; goto Label2;}

i=i+1;
}

```

```

Pf_hwdp1 = dPf_hwdp1*hwdp.height*3.28;

Label2:
//para os elementos discretizados no drill collar
if (mud.type == "Newtonian") dPf_dc1 =
newtonian_pipe(mud.q,mud.density,dc.ID,mud.visc);
if (mud.type == "Bingham") dPf_dc1 =
bingham_pipe(mud.q,mud.density,dc.ID,mud.T_0,mud.p_visc);
if (mud.type == "Power_Law") dPf_dc1 =
power_pipe(mud.q,mud.density,dc.ID,mud.K,mud.n);

while (depth_aux <= depth)
{
Pf_aux = dPf_dc1*(depth_aux-dp_height-
hwdp.height)*3.28;
Pt_aux1[i] = 0.17*depth_aux*mud.density + Pi -
(Pf_dp1+Pf_hwdp1+Pf_aux);
depth_aux1[i]=depth_aux;
depth_aux=depth_aux+dc.height/10;

//loop_conf << "dc " << depth_aux << " " <<
depth_aux1[i] << endl;

if (depth_aux < depth_aux1[i]) {depth_aux =
depth_aux1[i]; goto Label3;}
i=i+1;
}
Pf_dc1 = dPf_dc1*dc.height*3.28;

Label3:
//Calculo da pressão de fundo de poço
P_bh1 = 0.17*depth*mud.density + Pi -
(Pf_dp1+Pf_hwdp1+Pf_dc1);
Pt_aux1[i] = P_bh1;

P_bite = 8.311*pow(10,-
5)*mud.density*pow(mud.q/(0.95*bite.A),2);
P_bh2 = P_bh1 - P_bite;

//BHP PELO ANULAR DA COLUNA
i=0;

//para os elementos discretizados no drill collar
while (depth_aux>=depth-dc.height)
{
if (depth_aux < LDA) ODwell = ODriser;
else if (depth_aux > casing_depth_f) ODwell =
bite.OD;
else { if (rev.ID > 1) ODwell = rev.ID; else
ODwell = 30; }

if (mud.type == "Newtonian") dPf_dc2 =
newtonian_anulus(mud.q,mud.density,ODwell,dc.OD,mud.visc);
if (mud.type == "Bingham") dPf_dc2 =
bingham_anulus(mud.q,mud.density,ODwell,dc.OD,mud.T_0,mud.p_visc);
if (mud.type == "Power_Law") dPf_dc2 =
power_anulus(mud.q,mud.density,ODwell,dc.ID,mud.K,mud.n);

Pf_aux = dPf_dc2*(depth-depth_aux)*3.28;
Pt_aux2[i] = P_bh2 - 0.17*(depth-
depth_aux)*mud.density - (Pf_aux);

```

```

depth_aux2[i]=depth_aux;
depth_aux = depth_aux-dc.height/10;

//loop_conf << "dc_a " << depth_aux << " " <<
dPf_dc2 << " " << Pf_aux << endl;

depth_aux2[i]; goto Label4;}

if (depth_aux > depth_aux2[i]) {depth_aux =
i=i+1;
}
Pf_dc2 = dPf_dc2*dc.height*3.28;

Label4:
//para os elementos discretizados no heave-weight
drill pipe

while (depth_aux>=dp_height)
{
if (depth_aux < LDA) ODwell = ODriser;
else if (depth_aux > casing_depth_f) ODwell =
bite.OD;
else { if (rev.ID > 1) ODwell = rev.ID; else
ODwell = 30; }

if (mud.type == "Newtonian") dPf_hwdp2 =
newtonian_anulus(mud.q,mud.density,ODwell,hwdp.OD,mud.visc);
if (mud.type == "Bingham") dPf_hwdp2 =
bingham_anulus(mud.q,mud.density,ODwell,hwdp.OD,mud.T_0,mud.p_visc);
if (mud.type == "Power_Law") dPf_hwdp2 =
power_anulus(mud.q,mud.density,ODwell,hwdp.OD,mud.K,mud.n);

Pf_aux = dPf_hwdp2*(depth-dc.height-
depth_aux)*3.28;
Pt_aux2[i] = P_bh2 - 0.17*(depth-
depth_aux)*mud.density -(Pf_dc2+Pf_aux);

depth_aux2[i]=depth_aux;
depth_aux=depth_aux-hwdp.height/10;

//loop_conf << "hwdc_a " << depth_aux << " " <<
dPf_hwdp2 << " " << Pf_aux << endl;

depth_aux2[i]; goto Label5;}

if (depth_aux > depth_aux2[i]) {depth_aux =
i=i+1;
}
Pf_hwdp2 = dPf_hwdp2*hwdp.height*3.28;

Label5:
//para os elementos discretizados no drill-pipe

//calculo da pressão em diferetes pontos para
verificar perfil de pressão ao longo do poço
while (depth_aux>=depth_inicial)
{
if (depth_aux < LDA) ODwell = ODriser;
else if (depth_aux > casing_depth_f) ODwell =
bite.OD;
else { if (rev.ID > 1) ODwell = rev.ID; else
ODwell = 30; }

if (mud.type == "Newtonian") dPf_dp2 =
newtonian_anulus(mud.q,mud.density,ODwell,dp.OD,mud.visc);

```



```

        if (mud.type == "Bingham") dPf_dp2 =
bingham_anulus(mud.q,mud.density,ODwell,dp.OD,mud.T_0,mud.p_visc);
        if (mud.type == "Power_Law") dPf_dp2 =
power_anulus(mud.q,mud.density,ODwell,dp.OD,mud.K,mud.n);

        Pf_aux = dPf_dp2*(dp_height-depth_aux)*3.28;
        Pt_aux2[i] = P_bh2-0.17*(depth-
depth_aux)*mud.density - (Pf_aux+Pf_hwdp2+Pf_dc2);

        depth_aux2[i]=depth_aux;
        depth_aux=depth_aux-dp_height/100;

//loop_conf << "dp_a " << depth_aux << " " <<
dPf_dp2 << " " << Pf_aux << endl;

        if (depth_aux > depth_aux2[i]) {depth_aux =
depth_aux2[i]; goto Label6;}
        i=i+1;
        }
        Pf_dp2 = dPf_dp2*dp_height*3.28;

Label6:
//Calculo da pressão de superfície

        Psup = P_bh2 - 0.17*depth*mud.density -
(Pf_dp2+Pf_hwdp2+Pf_dc2);
        Psup_aux =
Pf_dp1+Pf_hwdp1+Pf_dc1+Pf_dp2+Pf_hwdp2+Pf_dc2 + P_bite - Pb2;

        //cout << Psup << endl;

//garantir que tenhamos uma pressão de superficiei
suficiente para vencer as perdas de carga
        if (abs(Psup) > 0.01) { goto HID_begin; }
        else (Psup = 0);

        //cout << "ready" << endl;

        Pt_aux2[i] = Psup;
        depth_aux1[i] = depth;

        j = i;
        i = 0;

        ECD = 5.8823*P_bh2/depth;
        HP = Pi*mud.q/1714;
        ECD_plot[counter] = ECD; //atribuindo para plot do
ECD
        density_plot[counter] = mud.density; //atribuindo
para plot densidade

p_teste.open("C:/UFRJDrillingSimulator/Outputs/OuputRealTime_HydraulicWell.txt");

        while (i <= j)
        {
                p_teste << setiosflags(ios::fixed) <<
setprecision(2) << setw(10) << depth_aux1[i] << setprecision(2) << setw(10) <<
Pt_aux1[i] << setprecision(2) << setw(10) << Pt_aux2[i] << endl;
                i++;
        }
        p_teste.close();

```

```

//Calculo do WOB
WOB2 = (hwdp.w + dc.w)*(1-mud.density/65.5)*(1/1.3);
//fator de seguranga para manter a linha neutra nos DCs
WOBd = WOB2/bite.OD;
J2 = (bite.WOBdm-(WOBd/1000))/(bite.WOBdm-
bite.WOBref/1000)*pow(60/bite.N,bite.H1)*(1/(1+0.5*bite.H2));

//Calculo do ROP instantaneo - Modelo de Borgoyne

TVD_ft = depth*3.28;
lit = lito[depth_int];

if (abs(abras[lit]) < 1) abras[lit] = abrasm;
hear = root(0.5*bite.H2,1,(-
1*time_bite2/(J2*abras[lit])));
Fj = 0.01823*0.95*mud.q*pow(mud.density*P_bite,0.5);

loop_conf << time_bite << " " << time_bite2 << " "
<< WOBd/1000 << " " << J2 << " " << abras[lit] << " " << hear*100 << endl;

F1 = exp(2.303*a_borg[lit][0]);
F2 = exp(2.303*a_borg[lit][1]*(10000-TVD_ft));
F3 =
exp(2.303*a_borg[lit][2]*pow(TVD_ft,0.69)*(Gp[depth_int]-9));
F4 = exp(2.303*a_borg[lit][3]*TVD_ft*(Gp[depth_int]-
ECD));
F5 = pow(((WOBd/1000-
WOBd_t[lit]/1000)/(bite.WOBref/1000-WOBd_t[lit]/1000)),a_borg[lit][4]);
F6 = pow(bite.N/60,a_borg[lit][5]);
F7 = exp(-1*a_borg[lit][6]*hear);
//F7 =
pow(1/(0.968125*pow(hear,2)+6*hear+1),a_borg[lit][6]);
F8 = pow((Fj/1000),a_borg[lit][7]);

ROPs = (F1*F2*F3*F4*F5*F6*F7*F8)/3.28;
ROP_plot[counter] = ROPs;

//Calculo do WOH
WOH = (dp_height*dp.razao*3.28 + hwdp.w + dc.w)*(1-
mud.density/65.5) - WOB2;

//Calculo do Volume de Lama no Poço
Vmud = (Vwell - Vcoluna)/0.159; //convertendo para
bbl

//considerando a abertura de fratura e perda de
fluido quando BHP > Gf
if (P_bh2 > 0.17*Gfm[depth_int]*depth) {x=1; Vperdas
= Vperdas + area_fract*ROPs*timestep/60; }

//corrigindo o Volume de lama no poco
Awell1 = (0.25*pi*pow(ODriser,2))*(0.000645);
Awell2 = (0.25*pi*pow(bite.OD,2))*(0.000645);

Vmud = Vmud - Vperdas/0.159;
depth_inicial = Vperdas/(Awell1-dp.A);

//considerando influxo se BHP < Gp
if (P_bh2 < 0.17*Gp[depth_int]*depth &
lito[depth_int] == 2)
{
    Pp = 0.17*Gp[depth_int]*depth;
}

```

```

                                qgas = 2.9525*pow(10,-
10)*(kperm*hreserv/viscgas)*(pow(Pp,2)-pow(P_bh2,2))/(2*log(re/(0.5*bite.OD)));
                                Vkick = Vkick + qgas*(timestep/60);
//convertendo timestep em dias
                                }

                                //definição do volume de lama nos tanques
                                Hsecao = ROPs*(timestep/60); //m
                                Vpit = Vpit - (Awell2-dc.A)*Hsecao/0.159 + Vkick;

                                //gerando arquivo texto com os resultados da
simulação

                                simul_results << "drilling" <<
setiosflags(ios::fixed) << setprecision(2) << setw(7) << time_total << setprecision(2)
<< setw(8) << depth << setprecision(2) << setw(9) << P_bh2;
                                simul_results << setprecision(2) << setw(7) << ECD <<
setprecision(2) << setw(9) << Pi << setprecision(2) << setw(8) << HP;
                                simul_results << setprecision(2) << setw(7) << ROPs
<< setprecision(0) << setw(5) << mud.q << setprecision(2) << setw(7) << WOH/1000 <<
setprecision(2) << setw(9) << Vmud << setprecision(2) << setw(7) << hear*100;
                                simul_results << setprecision(2) << setw(12) << Vpit
<< setprecision(2) << setw(7) << Vperdas/0.159 << setprecision(2) << setw(7) <<
x*(area_fract*ROPs*timestep/60)/0.159 << setprecision(2) << setw(7) << Vkick;
                                simul_results << setprecision(2) << setw(6) <<
Gp[depth_int] << setprecision(2) << setw(6) << Gfm[depth_int] << setprecision(2) <<
setw(3) << lit << endl;

                                time_conf << depth << " " << ROP << " " <<
a_borg[lit][0] << " " << a_borg[lit][1] << " " << a_borg[lit][2] << " " <<
a_borg[lit][3] << " " << a_borg[lit][4];
                                time_conf << " " << a_borg[lit][5] << " " <<
a_borg[lit][6] << " " << a_borg[lit][7] << endl;

                                clrscr;

                                SIM_return:
                                cout << "*****PERFURACAO POCO " << nome
<< "*****" << endl << endl;
                                cout << "FLUID:          " << mud.type << endl;
                                cout << "FLUID DENSITY:    " << mud.density << "
ppg" << endl;
                                cout << "PUMP RATE:        " << mud.q << " gal/min"
<< endl;
                                cout << "WOB:            " << WOBd << " lbf/in" <<
endl;
                                cout << "BITE ROTATION:    " << bite.N << " rpm" <<
endl << endl;

                                cout << "DEPTH:          " << setprecision(2) <<
depth << " m" << endl;
                                cout << "BOTTOM PRESSURE:  " << setprecision(2) <<
P_bh2 << " psia" << endl;
                                cout << "GP:            " << setprecision(2) <<
Gp[depth_int] << " ppg" << endl;
                                cout << "ECD:           " << setprecision(2) <<
ECD << " ppg" << endl;
                                cout << "GF:            " << setprecision(2) <<
Gfm[depth_int] << " ppg" << endl;
                                cout << "PUMP PRESSURE:   " << setprecision(2) <<
Pi << " psia" << endl;
                                cout << "PUMP POWER:     " << setprecision(2) <<
HP << " HP" << endl;

```

```

WOH << " lbf" << endl;
Vmud << " bbl" << endl;
Vpit << " bbl" << endl;
hear*100 << " %" << endl;
ROPs << " m/h" << endl;
time_acc << " h" << endl;
downtime << " h" << endl;
time_total/24 << " dias (" << time_total << " h)" << endl;

cout << "WEIGHT ON HOOK:      " << setprecision(2) <<
cout << "MUD VOLUME - WELL:  " << setprecision(2) <<
cout << "MUD VOLUME - PIT:   " << setprecision(2) <<
cout << "TOOTH WEAR:         " << setprecision(2) <<
cout << "ROP:                " << setprecision(2) <<
cout << "DRILLING TIME:       " << setprecision(2) <<
cout << "DOWNTIME:           " << setprecision(2) <<
cout << "TIME:              " << setprecision(2) <<

//Atribuição do nome da formação
if (lito[depth_int] == 1) litoName = "FOLHELHO";
if (lito[depth_int] == 2) litoName = "ARENITO";

z=0;
report << setiosflags(ios::fixed) << time_total << "
h " << setprecision(2) << setw(5) << depth << " m " << litoName << endl;
if (P_bh2 > 0.95*0.17*Gfm[depth_int]*depth) {report
<< setiosflags(ios::fixed) << time_total << " h " << setprecision(2) << setw(5) <<
depth << " m " << setprecision(2) << setw(5) << "**** A PRESSÃO DE FUNDO ATINGIU " <<
P_bh2/(0.17*Gfm[depth_int]*depth)*100 << "% DA PRESSÃO DE FRATURA ****" << endl; z=1;}
if (P_bh2 < 1.05*0.17*Gp[depth_int]*depth) {report
<< setiosflags(ios::fixed) << time_total << " h " << setprecision(2) << setw(5) <<
depth << " m " << setprecision(2) << setw(5) << "**** A PRESSÃO DE FUNDO ATINGIU " <<
P_bh2/(0.17*Gp[depth_int]*depth)*100 << "% DA PRESSÃO DE POROS ****" << endl; z=1;}
if (hear > 0.9) {report << time_total <<
setiosflags(ios::fixed) << " h " << setprecision(2) << setw(5) << depth << " m " <<
setprecision(2) << setw(5) << "**** O DESGASTE NA BROCA ATINGIU " << hear*100 << "%
****" << endl; z=1;}

if (kbhit()) {
ch = getch();
if (ch == 'm') goto SIM_SCREEN;
if (ch == 'f') goto FLUID;
if (ch == 'b') goto BIT;
if (ch == 't') goto PIPE;
if (ch == 'r') goto CASING;
if (ch == 'c') goto COMPLET;
if (ch == 'w') goto WELL_CONTROL;
if (ch == 'p') { ch2 = getch(); if (ch2 == 's') goto
SIM_resume; }

if (ch == 'z')
{
xpmax = 1024;
png_start(xpmax, ypmax);
//plot_conf.open("C:/VICTOR/WELL PROJECT
PROGRAM/plot_conf.txt");

pensize(4);
bgcolor("white");
fgcolor("black");
rectangle(20,20,xpmax-20,ypmax-20);
fgcolor("black");

```

```

//gridding_TIME_simul(well_obj, xpmax, ypmax,
t_sum_plot[well_obj_int]);
gridding_TIME(well_obj, xpmax, ypmax, t_sum_plot[well_obj_int], ROPmax);

        i=cond_depth-final_depth/data;
        while (i<=well_obj)
        {
            time_drawing(xpmax, ypmax, ROP[i], ROP[i-
final_depth/data], t_sum_plot[i], t_sum_plot[i-final_depth/data], i, i-
final_depth/data, well_obj, t_sum_plot[well_obj_int], ROPmax, 1);
            i=i+final_depth/data;
        }

        counter_aux = 1;

time_drawing_simul(xpmax, ypmax, t_desl, 0, cond_depth, 0, well_obj, t_sum_plot[well_obj_int]
, 2);

        while (counter_aux <= counter)
        {
            if (depth_plot[counter_aux] >=
depth_plot[counter_aux-1])

//time_drawing_simul(xpmax, ypmax, t_plot[counter_aux], t_plot[counter_aux-
1], depth_plot[counter_aux]-LDA, depth_plot[counter_aux-1]-
LDA, well_obj, t_sum_plot[well_obj_int], 2);

time_drawing(xpmax, ypmax, ROP_plot[counter_aux], ROP_plot[counter_aux-
1], t_plot[counter_aux], t_plot[counter_aux-1], depth_plot[counter_aux]-
LDA, depth_plot[counter_aux-1]-LDA, well_obj, t_sum_plot[well_obj_int], ROPmax, 2);
            plot_conf << t_plot[counter_aux] << " " <<
t_plot[counter_aux-1] << " " << depth_plot[counter_aux]-LDA << " " <<
depth_plot[counter_aux-1]-LDA << endl;
            counter_aux = counter_aux+1;
        }

        a=sem_reg;
        while(a <= well_obj)
        {
            if (a==sem_reg)
litology_drawing(xpmax, ypmax, lito[a], sem_reg, sem_reg, well_obj);
            else litology_drawing(xpmax, ypmax, lito[a], a, a-
final_depth/data, well_obj);
            a=a+final_depth/data;
        }

png_finish("C:/UFRJDrillingSimulator/Outputs/OutputRealTime_TimeDepthDiagram.png");
        goto SIM_resume;
    }

    if (ch == 'x')
    {
        xpmax = 1024;
        png_start(xpmax, ypmax);
        pensize(4);
        bgcolor("white");
        fgcolor("black");
        rectangle(30, 20, xpmax-20, ypmax-20);
    }

```

```

gridding(final_depth,10,ypmax,ypmax,Gfm[well_obj_int],8.5);

        i=final_depth/data;
        while (i<=well_obj)
        {
            grad_drawing(xpmax,ypmax,Gp[i]-error,Gp[i-
final_depth/data]-error,Gfm[i]+error,Gfm[i-final_depth/data]+error,0,0,i,i-
final_depth/data,well_obj,8.35,Gfm[well_obj_int],0);
            i=i+final_depth/data;
        }

        counter_aux = 1;
        while (counter_aux <= counter)
        {
            if (depth_plot[counter_aux] >=
depth_plot[counter_aux-1])

grad_drawing(xpmax,ypmax,ECD_plot[counter_aux],ECD_plot[counter_aux-
1],density_plot[counter_aux],density_plot[counter_aux-1],0,0,depth_plot[counter_aux]-
LDA,depth_plot[counter_aux-1]-LDA,well_obj,8.35,Gfm[well_obj_int],2);

                //plot_conf << t_plot[counter_aux] << " " <<
t_plot[counter_aux-1] << " " << depth_plot[counter_aux]-LDA << " " <<
depth_plot[counter_aux-1]-LDA << endl;
                counter_aux = counter_aux+1;
        }

        a=sem_reg;
        while(a <= well_obj)
        {
            if (a==sem_reg)
litology_drawing(xpmax,ypmax,lito[a],sem_reg,sem_reg,well_obj);
            else litology_drawing(xpmax,ypmax,lito[a],a,a,
final_depth/data,well_obj);
            a=a+final_depth/data;
        }

png_finish("C:/UFRJDrillingSimulator/Outputs/OutputRealTime_GradientDiagram.png");
        goto SIM_resume;
    }

    else goto SIM_SCREEN;
}

SIM_resume:
Sleep(60000*timestep/ff);
//getch();
time_bite_ant = time_bite;

time_acc = time_acc + (timestep)/60;
time_bite = time_bite + (timestep)/60;
time_bite2 = time_bite2 + timestep/60;

time_total = time_acc + downtime;
t_plot[counter] = time_total;
depth_plot[counter] = depth;
counter = counter+1;

if (time_bite = time_bite_ant) time_bite = time_bite
+ timestep/60;

```

```
        Psup_aux=0;
        depth = depth + (timestep/60)*ROPs;
        depth_int = depth-LDA;

        if (depth - LDA <= well_obj) goto MAIN_simulator;

        cout << endl << "-----PERFURACAO CONCLUIDA
COM SUCESSO-----" << endl;
    }

}
    getch();
    return 0;
}
```