



UNIVERSIDADE FEDERAL DO PARÁ
CENTRO DE GEOCIÊNCIAS
CURSO DE PÓS-GRADUAÇÃO EM GEOFÍSICA

DISSERTAÇÃO DE MESTRADO

INTERFACES GRÁFICAS DE ENTRADA E SAÍDA DA
eGs (electrical GEOPHYSICS suite)

RODRIGO ERASMO DA CONCEIÇÃO SILVA

BELÉM-PARÁ

2005

UNIVERSIDADE FEDERAL DO PARÁ
CENTRO DE GEOCIÊNCIAS
CURSO DE PÓS-GRADUAÇÃO EM GEOFÍSICA

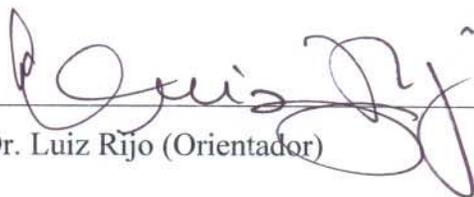
INTERFACES GRÁFICAS DE ENTRADA E SAÍDA DA
eGs (electrical GEOPHYSICS suite)

DISSERTAÇÃO APRESENTADA POR
RODRIGO ERASMO DA CONCEIÇÃO SILVA
COMO REQUISITO PARCIAL À OBTENÇÃO DE GRAU DE
MESTRE EM CIÊNCIAS NA ÁREA DE GEOFÍSICA

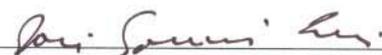
DATA DE APROVAÇÃO: 14/03/2005

NOTA: 93% (EXCELENTE)

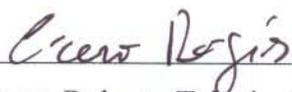
COMITÊ DE TESE:



Dr. Luiz Rijo (Orientador)



Dr. José Gouvêa Luiz



Dr. Cícero Roberto Teixeira Régis

BELÉM

2005

*Às minhas duas mães
e aos meus dois pais.*

AGRADECIMENTOS

Agradeço ao professor Luiz Rijo por todo o incentivo e paciência. Aos colegas Glauco Pereira, Marcos Welby, Victor Tocantins, Frayser Almeida e Gilberto Vogado, pela ajuda e descontração. Ao corpo docente do CPGf, pelo esforço para ofertar cursos de qualidade. Aos membros da secretaria do curso, em especial a Benildes Lopes, por toda a ajuda e boa vontade com relação às questões administrativas. Ao CNPq, pela concessão de bolsa de estudo, e ao PRH/ANP – 6, por auxiliar na infraestrutura computacional do PROEM (Laboratório de Processamento eletromagnético do CG/UFPa). Finalmente, aos meus familiares por todo o incentivo e compreensão.

SUMÁRIO

DEDICATÓRIA	i
AGRADECIMENTOS	ii
LISTA DE ILUSTRAÇÕES	vii
RESUMO	1
ABSTRACT	2
1 – INTRODUÇÃO	3
2 – DIRETRIZES NO DESENVOLVIMENTO DA eGs 3.0	6
2.1 – O SISTEMA OPERACIONAL.....	6
2.2 – A LINGUAGEM DE PROGRAMAÇÃO.....	7
2.3 – O USO DOS PROGRAMAS FONTES JÁ EXISTENTES.....	7
2.4 – PÚBLICO ALVO.....	9
2.5 – AS FERRAMENTAS PARA FACILITAR A CRIAÇÃO DE MODELOS GEOFÍSICOS 2D E 3D.....	10
2.6 – UM PROGRAMA AUTO-SUSTENTÁVEL.....	11
2.7 – ACRÉSCIMO DE NOVOS MÉTODOS PARA ÀS VERSÕES SEGUINTEs.....	11
3 – CARACTERÍSTICAS DA INTERFACE GRÁFICA DE ENTRADA E SAÍDA DA eGs	13
3.1 – FACILIDADE PARA CRIAÇÃO DE MODELOS GEOFÍSICOS.....	13
3.1.1 – Modelos 2D	13
3.1.1.1 – A janela para criação e modificação de modelos geofísicos 2D.....	13
3.1.1.2 – As opções para modificar as cores das células.....	14
3.1.1.3 – O ajuste das dimensões da malha.....	18
3.1.2 – Modelos 3D	19
3.1.2.1 – A janela para criação de modelos geofísicos 3D.....	19
3.1.2.2 – Facilidade para visualização de modelos geofísicos tridimensionais.....	20
3.2 – FACILIDADE PARA VISUALIZAR OS DADOS DE SAÍDA.....	30

3.2.1 – Interface para visualização de gráficos de curvas.....	30
3.2.1.1 – Quantidade de gráficos.....	30
3.2.1.2 – As escalas dos gráficos.....	31
3.2.1.3 – Os intervalos de visualização dos gráficos.....	33
3.2.2 – Interface para visualização de mapas de contorno.....	34
3.2.3 – Interface para visualização de pseudo-seções.....	34
3.2.4 – Interface para gerenciamento de arquivos de saídas do método MCSEM.....	36
3.3 – AUTOVERIFICAÇÃO DE ERROS.....	37
3.4 – UM PROGRAMA QUE NÃO DEPENDA DE OUTROS PROGRAMAS.....	38
3.5 – EXEMPLOS.....	39
4 – INFORMAÇÕES AO PROGRAMADOR.....	45
4.1 – INTRODUÇÃO	45
4.2 – INTERFACE PARA CRIAÇÃO E VISUALIZAÇÃO DE MODELOS GEOFÍSICOS 2D E 3D.....	46
4.2.1 – Os arquivos de entrada no processamento.....	46
4.2.2 – O armazenamento dos dados.....	48
4.2.3 – A estrutura gráfica das janelas.....	48
4.2.3.1 – Janela para geração de modelos geofísicos 2D ou 3D.....	49
4.2.3.2 – Janela para modificar o tamanho do modelo geofísico.....	50
4.2.3.3 – Janela para visualizar o modelo geofísico em três dimensões.....	50
4.2.4 – A estrutura de modificação dos dados.....	51
4.2.4.1 – Modificação das cores das células.....	52
4.2.4.2 – Os itens <i>Desfazer</i> e <i>Refazer</i>	54
4.2.4.3 – Geração da figura para a visualização em 3D.....	56
4.2.4.3.1 – Determinação das coordenadas dos nós.....	56
4.2.4.3.2 – Visualização por cortes.....	58
4.2.4.3.3 – Visualização por cores escolhidas.....	59
4.2.4.3.4 – Rotação do paralelepípedo.....	59
4.3 – INTERFACE PARA VISUALIZAÇÃO DE GRÁFICOS DE CURVAS.....	61
4.3.1 – Os arquivos usados na interface.....	62

4.3.2 – Armazenamento dos dados usados nos gráficos.....	64
4.3.3 – As janelas que compõem a interface.....	66
4.3.4 – O processo de geração de gráficos.....	67
4.3.5 – Discussão a respeito das principais opções.....	68
4.4 – INTERFACE PARA VISUALIZAÇÃO DE MAPAS DE CONTORNO.....	72
4.4.1 – Janela usada para visualização de mapas de contorno.....	72
4.4.2 – O armazenamento dos dados exibidos.....	73
4.4.3 – O funcionamento da interface.....	74
4.5 – INTERFACE PARA VISUALIZAÇÃO DE PSEUDO-SEÇÕES.....	75
4.5.1 – Os arquivos de dados do tipo IP.....	76
4.5.2 – A janela usada na interface.....	76
4.5.3 – O armazenamento dos dados.....	77
4.5.4 – O funcionamento da interface.....	78
4.5.4.1 – Os <i>Registros</i> criados para a interface.....	78
4.5.4.2 – Os passos para pintar a pseudo-seção (sem interpolação).....	79
4.5.4.3 – Os passos para obter a pseudo-seção suavizada (interpolação).....	79
4.6 – INTERFACE PARA GERENCIAMENTO DE ARQUIVOS DE SAÍDA DO MÉTODO MCSEM.....	81
4.6.1 – Os arquivos de resposta de modelos (arquivos de saída).....	82
4.6.2 – A estrutura da janela usada.....	83
4.6.3 – Armazenamento dos dados.....	84
4.6.4 – O funcionamento da interface.....	85
4.7 – SUGESTÕES PARA IMPLEMENTAÇÃO.....	87
4.7.1 – Personalização das cores.....	87
4.7.2 – Visualização dos resultados de inversão em 3D.....	87
4.7.3 – Armazenamento dos arquivos plotados.....	88
4.7.4 – Opção para plotar os dados em ordem decrescente.....	88
5 – CONCLUSÃO.....	89
REFERÊNCIAS BIBLIOGRÁFICAS.....	91

APÊNDICE A – MANUAL DO USUÁRIO	92
A.1 – ONDE OBTER A eGs 3.0.....	92
A.2 – INSTALAÇÃO DA eGs 3.0 (Windows 2000 ou XP).....	92
A.3 – COMO USAR A eGs 3.0.....	96
A.3.1 – 1D (uma dimensão)	96
A.3.1.1 – Magnetotelúrico (Problema Direto).....	96
A.3.2 - 2D (duas dimensões)	118
A.3.2.1 – IP e Resistividade 2D – Problema Direto.....	118

LISTA DE ILUSTRAÇÕES

Figura 2.1: Etapas da modelagem e interpretação de dados geofísicos.....	8
Figura 2.2: Atividades executadas na eGs pelos seus usuários.....	9
Figura 3.1: Janela usada para criar e modificar modelos geofísicos bidimensionais.....	14
Figura 3.2: Ilustração do funcionamento da opção <i>criar corpo retangular</i>	15
Figura 3.3: Ilustração do funcionamento da opção <i>mudar a cor de um corpo</i>	16
Figura 3.4: Ilustração do funcionamento da opção <i>mudar a cor dos corpos de mesma propriedade</i>	17
Figura 3.5: Ilustração do funcionamento da opção <i>copiar uma região retangular</i>	18
Figura 3.6: Ilustração do funcionamento da opção <i>colar região copiada</i>	18
Figura 3.7: Janela usada para criar e modificar modelos tridimensionais.....	20
Figura 3.8: Seções bidimensionais do <i>modelo 1</i> (tridimensional).....	21
Figura 3.9: Visualização do <i>modelo 1</i> em 3D através de <i>cortes</i> nos eixos.....	22
Figura 3.10: Visualização das células do <i>modelo 1</i> através de <i>cores selecionadas</i>	23
Figura 3.11: Segunda seção bidimensional do <i>modelo 2</i> (tridimensional).....	24
Figura 3.12: Visualização do <i>modelo 2</i> por <i>cortes</i>	25
Figura 3.13: Visualização do <i>modelo 2</i> por <i>cores selecionadas</i>	26
Figura 3.14: Seções bidimensionais do <i>modelo 3</i> (tridimensional).....	27
Figura 3.15: Visualização do <i>modelo 3</i> por <i>cortes</i>	28
Figura 3.16: Visualização do <i>modelo 3</i> por <i>cores selecionadas</i>	29
Figura 3.17: Ilustração de quatro gráficos de uma só vez.....	31
Figura 3.18: Ilustração de gráficos do tipo Linear-Linear e Log-Log.....	32
Figura 3.19: Ilustração do funcionamento da opção <i>Limites Automáticos</i>	33
Figura 3.20: Visualização da resposta do modelo através de mapas de contorno.....	34
Figura 3.21: Visualização da resposta do modelo ou dos dados observados do tipo IP.....	35
Figura 3.22: Suavização dos dados do tipo IP.....	36
Figura 3.23: Janela que permite escolher os arquivos, o modo de apresentação e a componente do campo elétrico dos dados de saída do método MCSEM.....	37
Figura 3.24: Alerta dado ao usuário em relação ao preenchimento incorreto dos campos onde são determinadas as dimensões da malha de elementos finitos.....	38

Figura 3.25: Alerta dado ao usuário em relação ao possível preenchimento incorreto dos campos onde são declaradas as propriedades físicas do modelo geoeletrico.....	38
Figura 3.26: Ilustração do modelo (com $h = 10$ metros) na janela que permite modificar as cores das células.....	39
Figura 3.27: Resposta do modelo com $h = 10$ metros e sua comparação com os dados observados.....	41
Figura 3.28: Resposta do modelo com $h = 20$ metros e sua comparação com os dados observados.....	42
Figura 3.29: Resposta do modelo com $h = 50$ metros e sua comparação com os dados observados.....	43
Figura 4.1: Janela usada para gerar modelos geofísicos 2D ou 3D.....	49
Figura 4.2: Janela usada para modificar o tamanho do modelo.....	50
Figura 4.3: Janela usada para visualizar o modelo geofísico tridimensionalmente.....	51
Figura 4.4: Ilustração dos pontos de fuga e dos nós que determinam todos os demais nós da figura.....	57
Figura 4.5: Ilustração dos nós nas faces XZ e YZ do paralelepípedo, e das retas usadas para determinar a posição desses nós.....	58
Figura 4.6: Ilustração dos nós do paralelepípedo através dos pontos de intersecção das retas usadas para determina-los.....	58
Figura 4.7: Janela usada para mostrar os gráficos.....	66
Figura 4.8: Janela usada para declarar quais colunas do arquivo serão usadas para gerar os gráficos.....	66
Figura 4.9: Janela usada para declarar os intervalos de visualização dos gráficos nas escalas <i>Linear-Linear</i> , <i>Log-Linear</i> e <i>Log-Log</i>	67
Figura 4.10: Janela usada para exibir mapas de contorno.....	73
Figura 4.11: Janela usada no método IP.....	77
Figura 4.12 : Ilustração do processo de interpolação linear dos dados do tipo IP.....	81
Figura 4.13: Janela usada para gerenciar os arquivos de resposta de modelos do método MCSEM.....	84
Figura A.1: Ícone de instalação da eGs 3.0	93
Figura A.2: Janela de boas-vindas para instalação do programa	93

Figura A.3: Informação para instalação	93
Figura A.4: Termo de licença	94
Figura A.5: Informações do diretório, do espaço necessário e livre no disco rígido	94
Figura A.6: Janela de confirmação da instalação	95
Figura A.7: Janela que informa a finalização da instalação do programa	95
Figura A.8: Janela Métodos	96
Figura A.9: Janela “Seleção” para abrir um <i>modelo geoeletrico</i> novo	97
Figura A.10: Abrir um <i>modelo geoeletrico</i> novo através do Menu	98
Figura A.11: Janela “Modelo” para início da edição de um <i>modelo geoeletrico</i>	98
Figura A.12: <i>Modelo geoeletrico</i> 1D usado como exemplo	99
Figura A.13: (a) Janela “Modelo” com permeabilidade magnética relativa igual a 1 em todas camadas.(b) Janela “Modelo” para alteração da permeabilidade magnética relativa	100
Figura A.14: Opção Salvar Modelo do Menu	101
Figura A.15: Janela de aviso quando um <i>modelo geoeletrico</i> está sendo fechado, mas as alterações não foram salvas	101
Figura A.16: Janela “Seleção” para abrir um <i>modelo geoeletrico</i> salvo	102
Figura A.17: Abrir um <i>modelo geoeletrico</i> salvo através do Menu	102
Figura A.18: Janela amigável para abertura de arquivos	103
Figura A.19: Janela “Modelo” aberta para reedição de um arquivo salvo anteriormente.....	103
Figura A.20: Janela “Seleção” para abrir a janela “Dados Observados” nova	104
Figura A.21: Abrir uma janela “Dados Observados” nova através do Menu	105
Figura A.22: Nova janela “Dados Observados” para início da edição.....	105
Figura A.23: Nova janela “Dados Observados” editada.....	106
Figura A.24: Opção <i>Salvar Dados Observados</i> do Menu.....	107
Figura A.25: Janela de aviso quando os <i>dados observados</i> estão sendo fechados, mas as alterações não foram salvas.....	107
Figura A.26: Janela “Seleção” para abrir <i>dados observados</i> salvos.....	108
Figura A.27: Abrir os <i>dados observados</i> salvos através do Menu.....	108
Figura A.28: Janela amigável para abertura de arquivos.....	109
Figura A.29: Janela “Dados Observados” aberta para reedição de um arquivo salvo anteriormente.....	109

Figura A.30: (a) Janela “Seleção” com o botão <i>Executar</i> ativado. (b) Itens <i>Executar Com Gráfico</i> e <i>Executar Sem Gráfico</i>	110
Figura A.31: Resultado da modelagem numérica com o item <i>Executar Com Gráfico</i> ativado	110
Figura A.32: Ilustração de uma modelagem usando o arquivo Mt1D.MGE como <i>modelo geoeletrico</i>	111
Figura A.33: Item Abrir Arquivo.....	112
Figura A.34: Sub-menus do menu <i>Gráfico</i>	113
Figura A.35: Janela que permite escolher as escalas de cada gráfico.....	114
Figura A.36: Janela que permite escolher quais colunas do arquivo serão usadas para gerar o gráfico.....	114
Figura A.37: Janela que permite escolher os intervalos de visualização dos gráficos.....	115
Figura A.38: Janela que permite escolher os textos que identificam a janela, os eixos e as curvas.....	116
Figura A.39: Janela que permite escolher os símbolos que representarão os pontos nos gráficos.....	116
Figura A.40: Janela Métodos.....	118
Figura A.41: Janela “Seleção” para abrir um <i>modelo geoeletrico</i> novo.....	119
Figura A.42: Abrir um <i>modelo geoeletrico</i> novo através do Menu.....	120
Figura A.43: Janela “Seleção” para abrir um <i>modelo geoeletrico</i> salvo.....	120
Figura A.44: Abrir um <i>modelo geoeletrico</i> salvo através do Menu.....	121
Figura A.45: Janela usada para editar o modelo 2D.....	121
Figura A.46: Barra de ferramentas da janela usada para desenhar o <i>modelo geoeletrico</i>	122
Figura A.47: Janela usada para definir as dimensões da malha usada no <i>modelo geoeletrico</i> do método IP e Resistividade 2D.....	124
Figura A.48: Ilustração da forma do <i>modelo geoeletrico</i> usado na modelagem usando o método IP e Resistividade 2D.....	125
Figura A.49: Campos destinados à declaração das propriedades físicas do <i>modelo geoeletrico</i>	125
Figura A.50: Opção <i>Salvar Modelo</i> do Menu.....	126

Figura A.51: Janela de aviso quando um <i>modelo geométrico</i> está sendo fechado, mas as alterações não foram salvas.....	126
Figura A.52: Janela “Seleção” para abrir a janela “Dados Observados” nova.....	127
Figura A.53: Abrir uma janela “Dados Observados” nova através do Menu.....	127
Figura A.54: Nova janela “Dados Observados” para início da edição.....	128
Figura A.55: Nova janela “Dados Observados” editada.....	128
Figura A.56: Escolha para edição da resistividade aparente, percentual do efeito da frequência e fator metálico ou apenas da resistividade aparente.....	129
Figura A.57: Escolha do tipo de propriedade que o usuário deseja editar.....	129
Figura A.58: Opção <i>Salvar Dados Observados</i> do Menu.....	130
Figura A.59: Janela de aviso quando os <i>dados observados</i> estão sendo fechados, mas as alterações não foram salvas.....	130
Figura A.60: Janela “Seleção” para abrir <i>dados observados</i> salvos.....	131
Figura A.61: Abrir os <i>dados observados</i> salvos através do Menu.....	131
Figura A.62: Janela para abertura de arquivos.....	132
Figura A.63: Janela “Dados Observados” aberta para reedição de um arquivo salvo anteriormente.....	132
Figura A.64: (a) Janela “Seleção” com o botão <i>Executar</i> ativado. (b) Itens <i>Executar Com Gráfico</i> e <i>Executar Sem Gráfico</i>	133
Figura A.65: Resultado da modelagem numérica com o item <i>Executar Com Gráfico</i> ativado.....	134
Figura A.66: Resposta do <i>modelo geométrico</i> do método IP e resistividade.....	135
Figura A.67: Opção para interpolação das pseudo-seções de Resistividade e IP (deve estar marcada a opção ”sim”).....	136

RESUMO

Apresentamos uma interface gráfica para o programa eGs (electrical GEOPHYSICS suíte) usado na modelagem e interpretação de dados elétricos e eletromagnéticos na exploração geofísica. Esta dissertação corresponde a parcela do programa que trata das interfaces de entrada de dados de modelos geoeletricos bi e tridimensionais e das interfaces de apresentação dos resultados da modelagem como também dos dados observados. O programa contém interfaces para criação, modificação e visualização de modelos geoeletricos bi e tridimensionais, e para apresentação da resposta do modelo através de gráficos e imagens. Este trabalho resultou num programa com diversas opções para alterar a geometria e as propriedades físicas do modelo geoeletricos e recursos de visualização para facilitar a interpretação de dados geofísicos. A interface gráfica pode ser usada para criar modelos geoeletricos dos métodos elétrico ou eletromagnético mais usados e analisar as respostas dos modelos e de dados observados em campanha de campo. O programa é suficientemente flexível de modo que novas interfaces gráficas de outros métodos elétricos e eletromagnéticos podem ser facilmente implementadas.

ABSTRACT

We present a graphic interface for the program eGs (electrical GEOPHYSICS suite) used for modeling and interpretation of electric and electromagnetic data in geophysical exploration. This master degree dissertation corresponds the portion of the program that treats the interfaces for bi and three-dimensional input data of geoelectrical models together with interfaces for the presentation of the modeling results and for the presentation of observed data as well. The program contains interfaces for creation, modification and visualization of bi and three-dimensional geoelectrical models and for illustrations of the calculated results through graphs and images. This work gives a program with several options to alter the geometry and the physical properties of the geoelectrical models, including facilities for three dimensions visualization in order to help geophysical data interpretation. The graphic interfaces can be used to create geoelectrical models associated with the most used electric and electromagnetic geophysical methods and to analyze the answers obtained by numerical modeling associated with different types of observed data in field campaign. The program is sufficiently flexible so that new graphic interfaces of other electric and electromagnetic methods can be implemented easily.

1 – INTRODUÇÃO

A geofísica tem como uma de suas principais características ser uma área multidisciplinar. A geofísica abrange conhecimentos de Matemática, Física, Geologia e Computação. A computação é largamente usada em geofísica seja no processamento de dados de campo, seja na modelagem e interpretação de dados processados. A construção de boa parte dos softwares para processamento, modelagem e interpretação de dados geofísicos é realizada nas universidades por professores e alunos de pós-graduação. Todavia, muitos desses programas são usados apenas por um pequeno grupo de pessoas, geralmente aquelas associadas ao projeto que motivou o desenvolvimento do software. Ademais, esses programas desenvolvidos nas universidades são, geralmente, mal documentados, mal estruturados e muito dependentes de programas comerciais, como, por exemplo, o MATLAB. É desejável, portanto, que os programas de computadores desenvolvidos nos cursos de pós-graduação sejam acessados pelo maior número possível de usuários e, na medida do possível, independam de programas comerciais, que comumente são muito caros para serem adquiridos. É dentro desses objetivos que se enquadra o conjunto de programas chamado eGs (electrical GEOPHYSICS suite).

A eGs é uma suite de programas (de acesso livre) para modelagem e interpretação de dados geofísicos elétricos e eletromagnéticos, através de uma interface simples e amigável. No início, na década de setenta, o uso dos programas que agora compõem a eGs era muito difícil, trabalhoso e muito dependente de outros programas para representar graficamente os resultados. O objetivo desta dissertação é contribuir com o desenvolvimento de interfaces gráficas que tornam a eGs mais simples de ser usada por um maior número de usuários, em particular nas disciplinas dos métodos elétricos e eletromagnéticos nos cursos de graduação e pós-graduação em geofísica e geologia, assim como no desenvolvimento de TCCs e dissertações de mestrado.

Na década de setenta, foi iniciada pelo professor Luiz Rijo (quando este ainda era aluno de doutorado na Universidade de Utah) a construção de uma biblioteca de programas para modelagem numérica de dados geofísicos elétricos e eletromagnéticos chamada EGSLIB (Electrical Geophysics Software Library). A EGSLIB foi continuamente modificada no decorrer dos anos em função das transformações sofridas pelo setor computacional, e em seu último estágio (no fim da década de noventa), era formada por uma biblioteca de programas FORTRAN para MS-DOS utilizados conjuntamente com o pacote MATLAB.

A metodologia usada no último estágio da EGSLIB tinha dois inconvenientes: a geração dos *inputs* que só podia ser feita manualmente através de um editor de texto ASCII, e a dependência de outros programas para a apresentação dos resultados (o que implicava também na necessidade de adquirir licenças de programas comerciais, nem sempre de baixo custo). Para contornar essas dificuldades, foram dados os primeiros passos na tentativa de criar uma interface gráfica que possibilitasse a geração de um programa que fosse independente de outros programas, e que facilitasse a geração de modelos geoeletricos usados na interpretação dos dados de campo.

Os primeiros resultados foram obtidos por Rodrigo Valle em 1998, que no seu trabalho de iniciação científica orientado pelo professor Luiz Rijo, construiu uma interface gráfica para gerar os arquivos que continham as informações do modelo geoeletrico* usados como *input* para os programas da agora chamada eGs (Electrical GEOPHYSICS Suite). No entanto, esta versão não atendia ainda ao requisito de ser independente de outros programas, que seria concretizada somente alguns anos depois.

Em 2000 foi construída uma interface gráfica (Monteiro, 2001) que foi além do trabalho citado acima, permitindo que o seu usuário construísse modelos geoeletricos em uma, duas ou três dimensões, e também inversão de dados unidimensionais. Infelizmente, a carência de documentação do programa não possibilitou a continuação desse trabalho, o que nos levou a reiniciá-lo para poder prosseguir com o projeto.

Em 2002, foi iniciada a construção da versão mais atual da eGs: a eGs 3.0. Este é o resultado de um trabalho feito em conjunto por mim e Glauco L. Pereira (também aluno de mestrado), e que teve como objetivo construir um programa com alto grau de visualização, facilidade para criação de modelos geoeletricos, uma variedade de opções para a construção dos dados de entrada** e apresentação dos dados de saída**, e independente de outros programas.

* Um *modelo geoeletrico* é uma descrição simplificada da distribuição de propriedades elétricas na sub-superfície que se deseja determinar. Nesta dissertação não será feita nenhuma distinção entre modelo geofísico e modelo geoeletrico.

** A estrutura da eGs consta de três módulos. O primeiro módulo é o de entrada dos dados, o segundo módulo é o engenho que efetivamente executa a computação científica e o terceiro módulo o de apresentação de dados de saída. Neste trabalho chamaremos o primeiro módulo de *módulo de entrada* ou *módulo de pré-processamento*, o segundo módulo chamaremos de *kernel* ou *módulo de processamento* e o terceiro módulo chamaremos de *módulo de saída* ou *módulo de pós-processamento*. Aqui o termo processamento se refere aos processos computacionais executados pelo *kernel*, e não tem o significado mais comum em geofísica, ou seja, o de tratamento de dados e filtragem.

Esta dissertação faz parte de um projeto em que foram escritas duas dissertações: a dissertação escrita por Glauco L. Pereira (Pereira, 2005), a qual trata da *reformulação e ampliação da eGs com ênfase nas interfaces de entrada de dados observados e de modelos 1D, e visualização de inversão em 1D*, e esta dissertação, que trata das *interfaces gráficas de entrada de modelos bi e tridimensionais, e de apresentação dos resultados da modelagem*. Estas duas dissertações são complementares no que diz respeito ao entendimento da eGs como um todo. Para se ter uma melhor compreensão desta dissertação, é aconselhável que o leitor leia a dissertação de Pereira (2005) antes desta. Esta dissertação tem como objetivo dar uma visão geral do que é a eGs 3.0, dando ênfase à parcela do programa da qual fiquei com a responsabilidade de construir.

Devido ao seu aspecto não convencional, esta dissertação está dividida em três partes. A primeira (Capítulos 1, 2 e 3) e a terceira (Apêndice A) partes são voltadas principalmente para pessoas que queiram fazer uso da eGs 3.0 na modelagem e interpretação de dados. A segunda parte (Capítulo 4) é voltada para programadores interessados no código fonte da eGs a fim de dar continuidade ao seu desenvolvimento.

Em resumo, no Capítulo 2, é dada uma descrição do processo de desenvolvimento de *software* do programa, justificando o uso do sistema operacional e da linguagem de programação, e ainda descrevendo as estratégias usadas. No Capítulo 3 é dada uma visão dos resultados deste trabalho, mostrando as principais opções, vantagens e facilidades de uso das interfaces. O Capítulo 4 é destinado aos programadores que tenham interesse em saber dicas de como foi desenvolvida a eGs 3.0, e que venham no futuro a entrar no seu projeto de desenvolvimento. O Apêndice A é um manual destinado aos usuários que desejam ter um primeiro contato com o programa.

2 – DIRETRIZES NO DESENVOLVIMENTO DA eGs 3.0

A análise feita neste capítulo dá prioridade aos aspectos da eGs 3.0 descritos nesta dissertação. Para obter maiores detalhes sobre como foram construídas as interfaces e suas ferramentas o leitor deve consultar o Capítulo 4, e para uma análise mais profunda da eGs 3.0 por completo o leitor deve consultar também Pereira (2005).

No desenvolvimento de um *software*, alguns fatores devem ser levados em consideração a fim de obter um produto final de boa qualidade, fácil de ser manuseado, que seja usado por um grande número de pessoas e com facilidade para que novos programadores se juntem ao seu projeto de desenvolvimento. Para atender a estas necessidades, foram levados em consideração os seguintes fatores no desenvolvimento da eGs 3.0 :

- Em que sistema operacional a eGs 3.0 deveria ser desenvolvida?
- Qual a linguagem de programação mais apropriada para o desenvolvimento e a manutenção do programa?
- Como aproveitar os programas já existentes (gerados em FORTRAN) nas interfaces gráficas?
- Que tipos de usuários são esperados para o programa e quais as principais ferramentas usadas por eles?
- O que se deve fazer para facilitar a criação de modelos geofísicos?
- O que se deve fazer para deixar a eGs 3.0 independente de outros programas?
- Que critérios devem ser usados para facilitar o acréscimo de novos métodos geofísicos em futuras versões da eGs?

A discussão destes fatores é feita a seguir.

2.1 – O SISTEMA OPERACIONAL

A escolha do sistema operacional onde um determinado software será executado deve ser feita levando-se em consideração a sua abrangência em termos de número de usuários, tentando evitar assim que um indivíduo precise aprender a usar um novo sistema operacional para ter acesso ao software. Dentre os sistemas operacionais existentes no mercado, o MS-Windows é o mais usado em todo o mundo atualmente, e por este motivo, a eGs 3.0 está sendo construída para

ser executada neste sistema operacional. Além disso, as versões anteriores da eGs foram desenvolvidas para o MS-Windows.

2.2 – A LINGUAGEM DE PROGRAMAÇÃO

Na escolha da linguagem de programação usada para gerar o executável da eGs 3.0 optou-se por usar critérios que facilitassem a geração de uma interface gráfica, e que aumentasse as chances de se encontrar novos programadores para entrar no projeto. Sendo assim, esta linguagem de programação deveria ter:

- Facilidade para gerar programas de interfaciamento gráfico;
- Opções diversificadas de componentes para montar uma interface gráfica;
- Um razoável número de programadores.

O uso do ambiente de desenvolvimento DELPHI da Borland é uma excelente opção para atender as necessidades descritas acima. Montar programas contendo janelas, botões, caixas de texto e outros objetos, não traz muitas dificuldades neste ambiente, além do que, há um amplo leque de opções de componentes que podem enriquecer o aspecto visual de uma aplicação. Este ambiente de desenvolvimento é bastante usado para este fim, o que traz consigo a vantagem de ter vários programadores que têm afinidade com a linguagem Object Pascal (que é a linguagem usada nos códigos fonte dos programas feitos em DELPHI), facilitando assim encontrar novos indivíduos que se interessem pelo projeto e possam dar continuidade ao desenvolvimento do programa. Além do mais, as versões anteriores da eGs também foram desenvolvidas em DELPHI.

2.3 – O USO DOS PROGRAMAS FONTES JÁ EXISTENTES

A eGs é composta por uma biblioteca de programas para modelagem e interpretação de dados geofísicos elétricos e eletromagnéticos. Todos os programas desta biblioteca funcionam da seguinte maneira: Primeiramente, os dados que contêm as características do modelo geoeletrico precisam ser armazenados em um arquivo, para que posteriormente possam ser usados pelo programa. A etapa de geração e armazenamento desses dados é executada pelo *módulo de entrada* ou *módulo de pré-processamento*. Em seguida, estes dados são lidos pelo programa, que

então calcula a resposta do modelo geoeletrico e armazena os resultados em arquivo. Esta etapa é executada pelo *kernel* ou *módulo de processamento*. Finalmente, os dados gerados pelo *kernel*, ou seja, os dados de saída ou *resposta do modelo*, que serão usados na interpretação de dados de campo, ou seja, os *dados observados*. A etapa de apresentação da resposta do modelo geoeletrico é executada pelo *módulo de saída* ou *módulo de pós-processamento*. A Figura 2.1 ilustra as três etapas discutidas acima representadas pelos seus respectivos módulos:

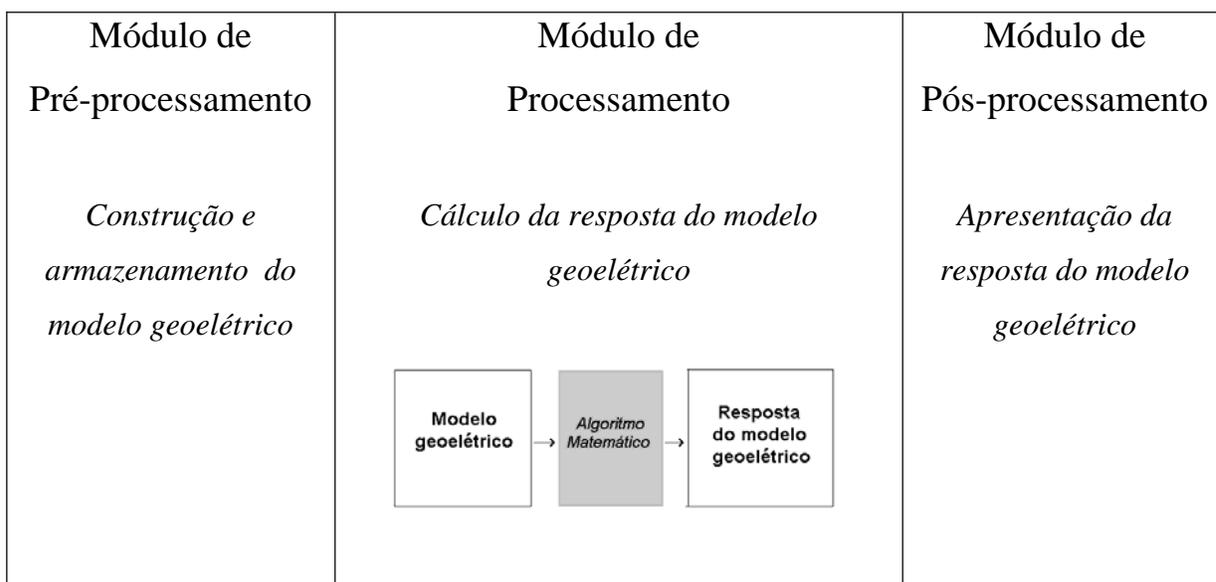


Figura 2.1: Etapas da modelagem e interpretação de dados geofísicos.

A computação numérica executada pelo *kernel* da eGs é feita por técnicas avançadas de solução numérica das equações de Maxwell, como por exemplo, *transformadas integrais*, *elementos finitos* e *equações integrais* (Rijo, 1992). Com base nessas técnicas, uma grande variedade de programas foi desenvolvida em FORTRAN para diversos métodos geofísicos elétricos e eletromagnéticos. Para fazer uso desses programas na interface gráfica da eGs, foram geradas a partir deles bibliotecas de ligação dinâmica (DLL) que tornam possível a integração com a estrutura gráfica da eGs.

2.4 – PÚBLICO ALVO

A eGs 3.0 é um software de uso principalmente acadêmico. Ela é destinada principalmente a alunos de graduação ou de pós-graduação dos cursos de geofísica ou geologia, como complemento às disciplinas de métodos elétricos e eletromagnéticos. Ela pode também ser usada na interpretação de dados de campo relacionados a TCCs e dissertações de mestrado. Naturalmente, outros profissionais (geofísicos ou geólogos que trabalham com métodos elétricos e eletromagnéticos) podem utilizar a eGs para fazer interpretação de dados de campo. A Figura 2.2 mostra um resumo sobre os usuários da eGs e as atividades desenvolvidas com o programa.

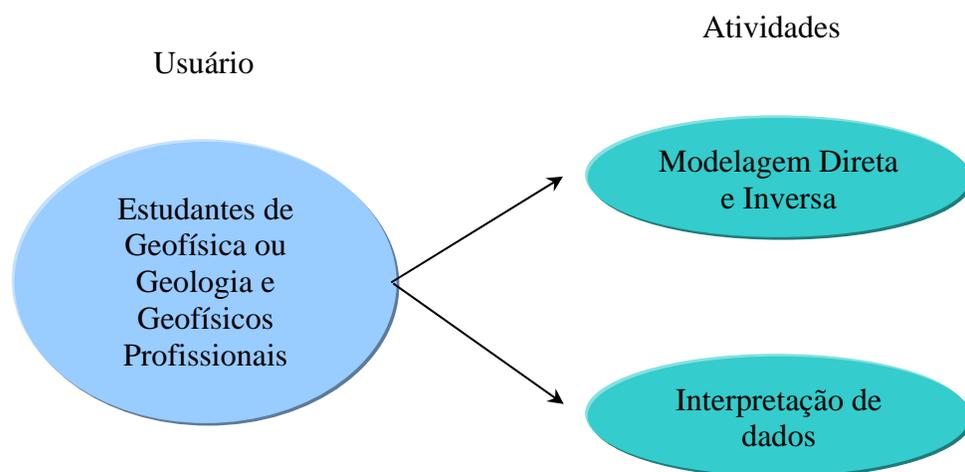


Figura 2.2: Atividades executadas na eGs pelos seus usuários.

Como se pode observar, a eGs 3.0 é útil para uma variedade de usuários com interesse em geofísica. Esta abrangência com relação aos tipos de usuários do programa foi um dos fatores mais relevantes na sua construção, pois isso é importante tanto para a sua divulgação, quanto para o seu aprimoramento.

2.5 – AS FERRAMENTAS PARA FACILITAR A CRIAÇÃO DE MODELOS GEOFÍSICOS 2D e 3D

Para facilitar a criação de modelos geofísicos, o programa deve dispor de opções para auxiliar o usuário nesta tarefa. Por isso, a interface para criação de modelos geofísicos 2D e 3D foi construída com uma variedade de opções para modificar as propriedades físicas das células que compõem o modelo geoeletrico, assim como para a visualização destes modelos em três dimensões*.

Como já foi citado anteriormente, o processamento dos dados da eGs é feito por programas que usam, por exemplo, o método dos *Elementos Finitos*. Neste método o modelo é subdividido em células (*elementos*), dentro das quais as propriedades físicas são consideradas constantes (Rijo, 1977). Na interface gráfica as células são apresentadas com cores constantes, e estas cores são associadas aos valores das propriedades físicas. Para ajudar o usuário a determinar qual propriedade física será atribuída a cada célula da malha de elementos finitos, a eGs 3.0 dispõe de opções que permitem fazer as seguintes alterações no modelo:

- Atribuir uma mesma cor às células de uma região retangular;
- Modificar as cores das células de um corpo;
- Modificar, simultaneamente, as cores de todas as células que apresentam uma mesma cor;
- Copiar as cores de uma região retangular da malha para colar em outras posições.

A construção de modelos geofísicos 3D na eGs é feita através de um método bastante prático, no qual altera-se as propriedades físicas das células em seções bidimensionais que representam grupos de fatias do modelo geofísico tridimensional. Porém, este tipo de modelagem dificulta a visualização do modelo em três dimensões. Por isso, a eGs 3.0 dispõe de uma interface que permite a visualização em 3D de modelos geofísicos bi e tridimensionais.

*Neste trabalho vamos nos deter aos modelos 2D e 3D. Os modelos 1D são tratados em Pereira (2005).

2.6 – UM PROGRAMA AUTO-SUSTENTÁVEL

De acordo com o que foi apresentado no Capítulo 1, uma das características propostas para a eGs 3.0 é a de ser um programa independente de outros. De acordo com a descrição feita naquele capítulo, para a eGs ser independente de outros programas ela precisa de sua própria interface para apresentação dos resultados da modelagem de dados, ou seja, uma interface para gerar gráficos (de curvas e imagens (mapas de contorno)) dos dados observados e dos dados de resposta dos modelos geoeletricos (*outputs* do processamento), permitindo a comparação entre estes dados feita, muitas vezes, através de ajuste de gráficos. Portanto, a eGs 3.0 foi construída com interfaces para fazer este tipo de comparação. A interface que permite exibir os dados através de gráficos de curvas tem diversas opções para fazer gráficos em escala linear ou logarítmica, comparação entre dados, entre outros. A interface que permite exibir os dados através de imagens também tem opções de visualização em escala linear e logarítmica. Para dados (de campo e/ou resposta do modelo geoeletrico) do tipo IP (Induced Polarization) foi construída uma interface para apresenta-los na forma de pseudo-seções.

2.7 – ACRÉSCIMO DE NOVOS MÉTODOS PARA AS VERSÕES SEGUINTE

A eGs tem dentro de sua concepção a vantagem de usar uma mesma estrutura para muitos dos arquivos usados como *inputs* no processamento de seus dados, e dos *outputs* gerados neste processamento. Há poucas mudanças na estrutura de arquivos de métodos diferentes, o que simplifica a geração dos *inputs* e análise dos *outputs*, tanto para o usuário quanto para o programador.

Graças à semelhança na maioria dos seus *inputs*, é quase sempre possível acrescentar um novo método usando interfaces gráficas feitas a partir de um mesmo tipo de formulário (janela) já existente, sem grandes mudanças.

Os *outputs* são, em sua maioria, analisados através de gráficos de curvas ou de imagens, o que permite, em geral, o uso de uma mesma estrutura de formulários para fazer esta análise. Entretanto, em alguns casos a forma do *output* é específica, como é o caso das pseudo-seções de IP.

Uma boa documentação do programa também é um fator que facilita a inclusão de novos métodos, além de evitar que o projeto fique impossibilitado de ser continuado e conseqüentemente evitando o desperdício de tempo e verbas no seu prosseguimento. Por isso, a eGs 3.0 vem sendo construída com a preocupação de deixar o programa com uma documentação adequada para a sua manutenção e ampliação. O Capítulo 4 desta dissertação também serve de guia aos programadores que venham a dar continuidade ao projeto eGs.

3 – CARACTERÍSTICAS DA INTERFACE GRÁFICA DE ENTRADA E SAÍDA DA eGs

Neste capítulo serão mostradas as opções da interface gráfica da eGs, com relação às interfaces feitas para criação e visualização de modelos geofísicos 2D e 3D, e para apresentação da resposta do modelo geofísico e sua comparação com os dados observados.

3.1 – FACILIDADE PARA CRIAÇÃO DE MODELOS GEOFÍSICOS

Ao contrário de como se fazia no passado, criar um modelo geofísico bi ou tridimensional na eGs 3.0 agora é uma tarefa muito fácil. Enquanto que antes era complicado alterar e visualizar modelos desse tipo (através do *prompt* da tela do monitor do computador), agora não são necessários mais do que alguns clicks e o preenchimento de alguns poucos campos.

3.1.1 – Modelos 2D

3.1.1.1 – A janela para criação e modificação de modelos geofísicos 2D

Os modelos geofísicos bidimensionais da eGs são processados através de uma malha do método dos Elementos Finitos, a qual é subdividida em várias células retangulares (na verdade são duas células triangulares que juntas formam um retângulo). Cada uma dessas células representa uma região com propriedade física constante, e que na interface gráfica é representada por uma região de cor constante. A Figura 3.1 ilustra a janela usada para criar e modificar estes modelos. Nesta janela, cada cor usada no modelo geofísico pode representar um tipo de rocha, a qual é caracterizada, nos métodos elétricos e eletromagnéticos, pela resistividade e pela permeabilidade magnética relativa. Para cada cor usada no modelo está reservado um espaço para o preenchimento dessas propriedades físicas (os campos no canto direito inferior da Figura 3.1).

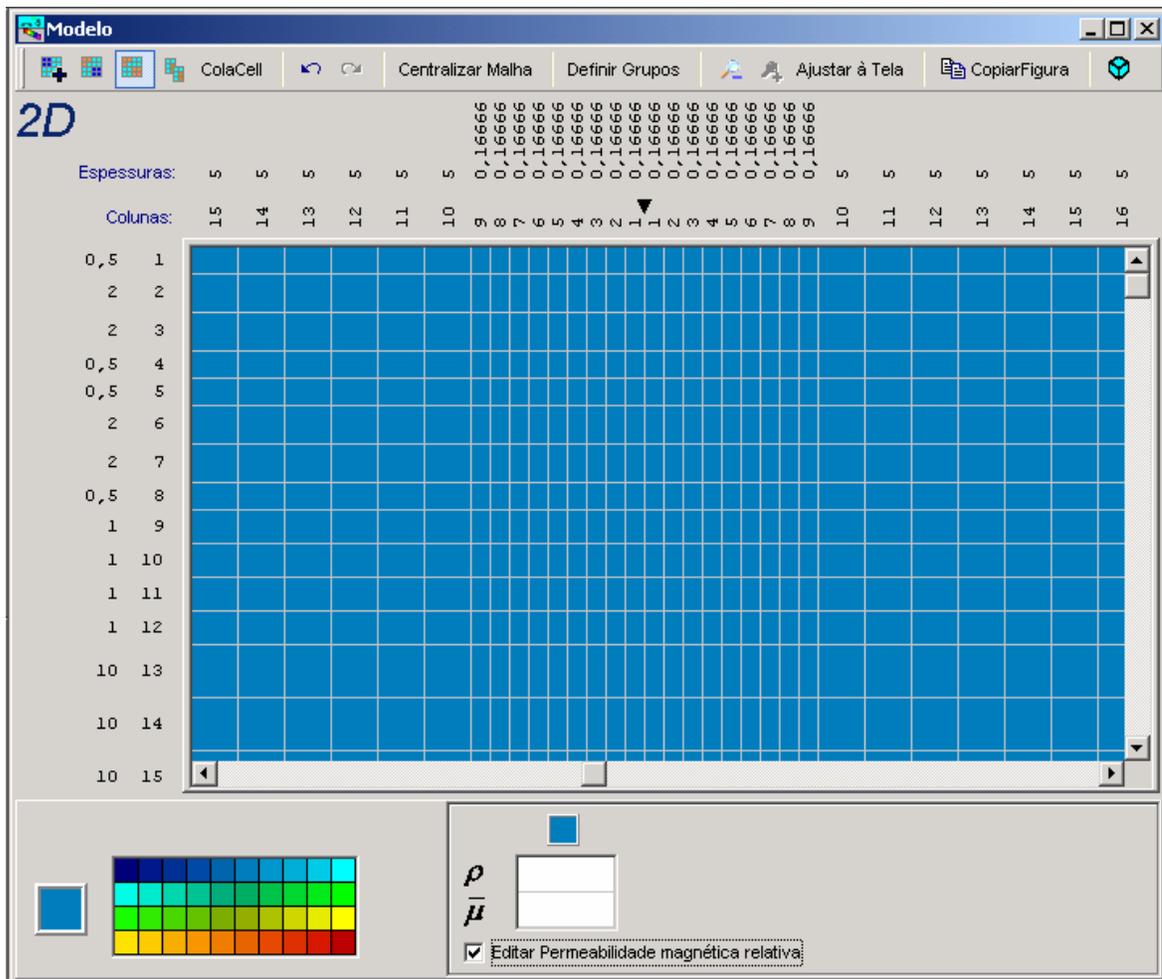


Figura 3.1: Janela usada para criar e modificar modelos geofísicos bidimensionais.

As malhas usadas para identificar os modelos geofísicos têm, em geral, uma grande variação nos tamanhos de suas células (as células próximas da borda do modelo são cerca de mil vezes maiores que as centrais). Para facilitar a visualização destes modelos a eGs 3.0 exibe as dimensões dessas células em escala logarítmica, permitindo visualizar um maior intervalo da malha de uma só vez, sem perder totalmente a heterogeneidade dos tamanhos das células.

3.1.1.2 – As opções para modificar as cores das células

Para facilitar a criação dos modelos geofísicos, esta interface possui quatro opções que permitem modificar as cores das células. O funcionamento destas opções será explicado e ilustrado a seguir.

Para pintar de uma mesma cor as células de uma região retangular, basta usar a opção *criar corpo retangular* (o botão que ativa esta opção está ilustrado nos exemplos no Apêndice A). Para pintar as células basta escolher uma cor e arrastar o mouse de um canto ao outro de uma região retangular que contenha estas células, da mesma maneira como se seleciona vários ícones no Windows apenas com o mouse. A Figura 3.2 ilustra três camadas e dois corpos construídos com o auxílio desta ferramenta.

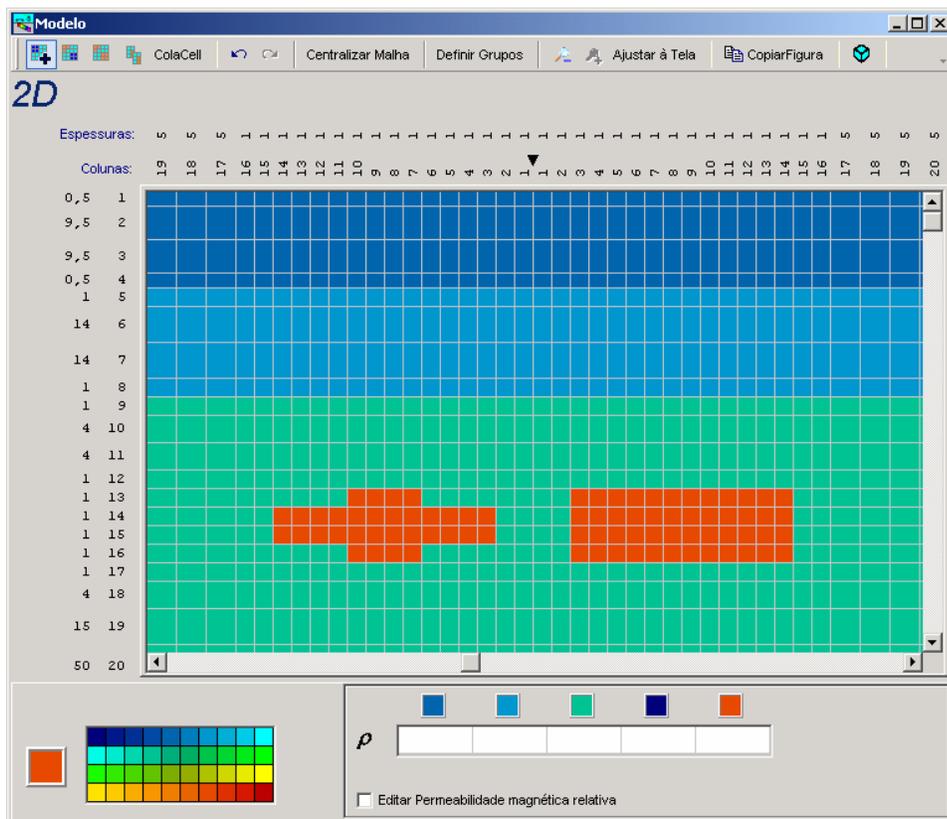


Figura 3.2: Ilustração do funcionamento da opção *criar corpo retangular*. Foram inseridos três camadas (cores azul, azul claro e verde) e dois corpos (cor alaranjada).

Caso o usuário precise modificar as cores de um conjunto de células de mesma cor e que estejam isoladas das demais células da mesma cor, basta acionar a opção *mudar a cor de um corpo* (o botão que ativa esta opção está ilustrado nos exemplos no Apêndice A). Com esta função ativada, basta escolher a nova cor e clicar em cima de uma das células deste conjunto de células. A Figura 3.3 mostra o resultado do uso desta ferramenta em um dos corpos da cor alaranjada, que passa a ser de cor azul claro.

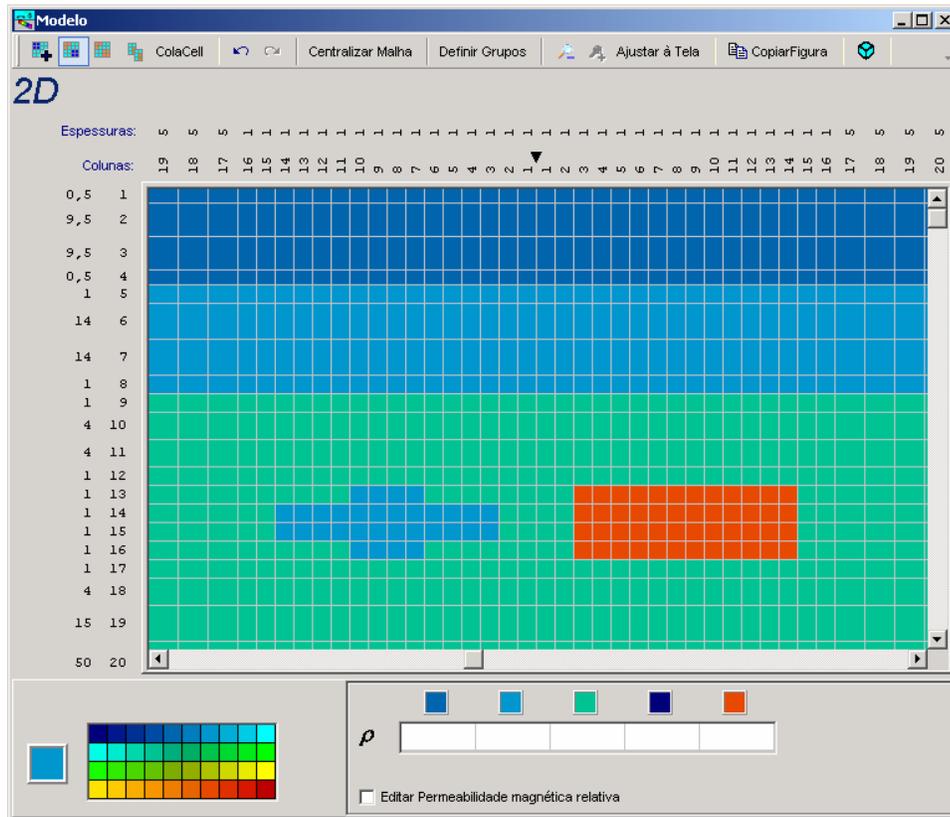


Figura 3.3: Ilustração do funcionamento da opção *mudar a cor de um corpo*. Um dos corpos que era da cor alaranjada passou a ser da cor azul claro.

Para modificar as cores de todas as células de uma mesma cor, usa-se a opção *mudar a cor dos corpos de mesma propriedade* (o botão que ativa esta opção está ilustrado nos exemplos no Apêndice A). Basta selecionar a nova cor e clicar em uma das células que tenha a cor que se deseja trocar. A Figura 3.4 mostra o funcionamento desta ferramenta. Nesta figura, as células de cor azul claro passam a ser da cor alaranjada.

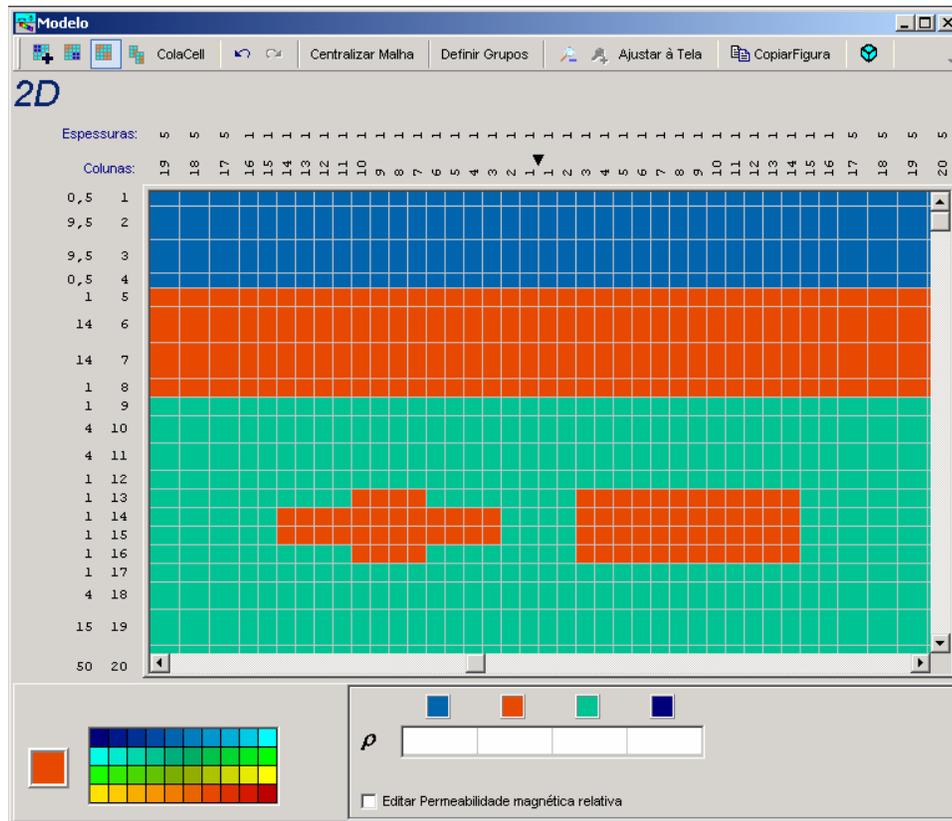


Figura 3.4: Ilustração do funcionamento da opção *mudar a cor dos corpos de mesma propriedade*. Os corpos que eram da cor azul claro passaram a ser da cor alaranjada.

Em alguns casos é conveniente copiar as cores das células de uma região para colar em outra. Por isso esta interface dispõe das opções *copiar células* e *colar células* (os botões que ativam estas opções estão ilustrados nos exemplos no Apêndice A) que permitem copiar e colar, respectivamente, as cores de células de uma região retangular. As Figuras 3.5 e 3.6 mostram como estas opções são úteis para a construção de um modelo que possui um corpo inclinado. Na Figura 3.5 construiu-se um pequeno trecho do corpo. A Figura 3.6 mostra como ficou o modelo após copiar a região retangular que continha o trecho já construído, e colado nas devidas posições.

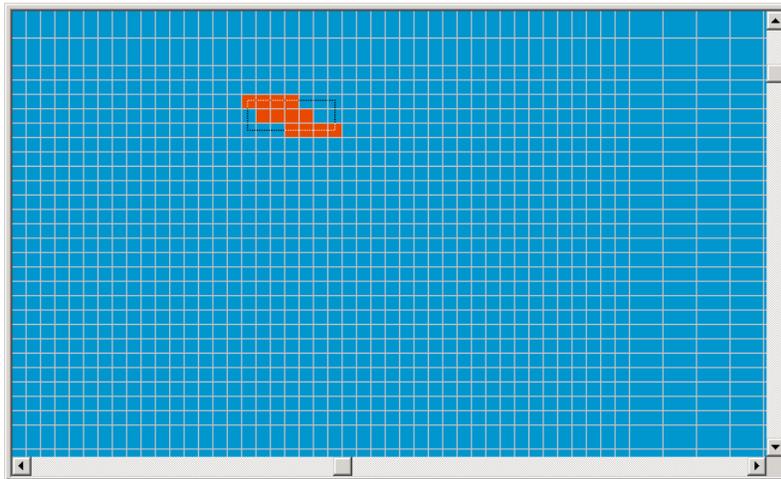


Figura 3.5: Ilustração do funcionamento da opção *copiar uma região retangular*. Foram copiadas as cores das células limitadas pela linha retangular.

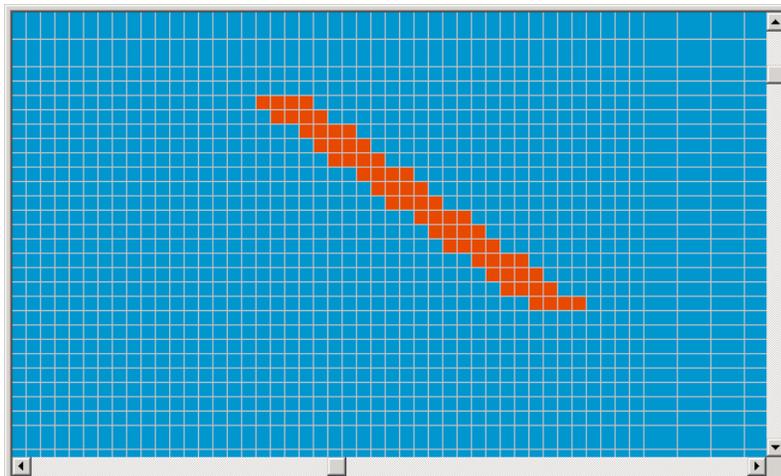


Figura 3.6: Ilustração do funcionamento da opção *colar região copiada*. O trecho copiado foi colado quatro vezes para formar o corpo por inteiro.

3.1.1.3 – O ajuste das dimensões da malha

No início, antes da introdução de uma interface gráfica, as dimensões da malha de elementos finitos usada na modelagem dos dados na eGs eram declarados diretamente num editor de texto. Na eGs 3.0 estes dados são declarados em uma janela com diversos campos que aparecem ou não de acordo com o método geofísico usado, a dimensão do modelo geofísico (2D

ou 3D), e de outros dados declarados na própria janela e que regulam o número de campos que devem ser preenchidos.

3.1.2 – Modelos 3D

A criação e modificação de modelos geofísicos tridimensionais são feitas na mesma janela usada para modelos bidimensionais, mas com ligeiras alterações. Todas as opções mostradas durante a apresentação da interface para criação e modificação de modelos bidimensionais também são válidas para modelos tridimensionais. Nesta seção, será apresentada também a interface para visualização de modelos tridimensionalmente, a qual pode ser usada para visualizar tanto modelos bidimensionais quanto tridimensionais.

3.1.2.1 – A janela para criação de modelos geofísicos 3D

A construção e a modificação de modelos geofísicos tridimensionais são feitas em seções bidimensionais que representam um determinado grupo de fatias (perpendiculares ao eixo Y) do modelo tridimensional. Estas seções são visualizadas na mesma janela usada para modelos geofísicos 2D, com a diferença de que aqui aparecem botões que permitem alternar entre as seções. A Figura 3.7 mostra a nova aparência da janela.

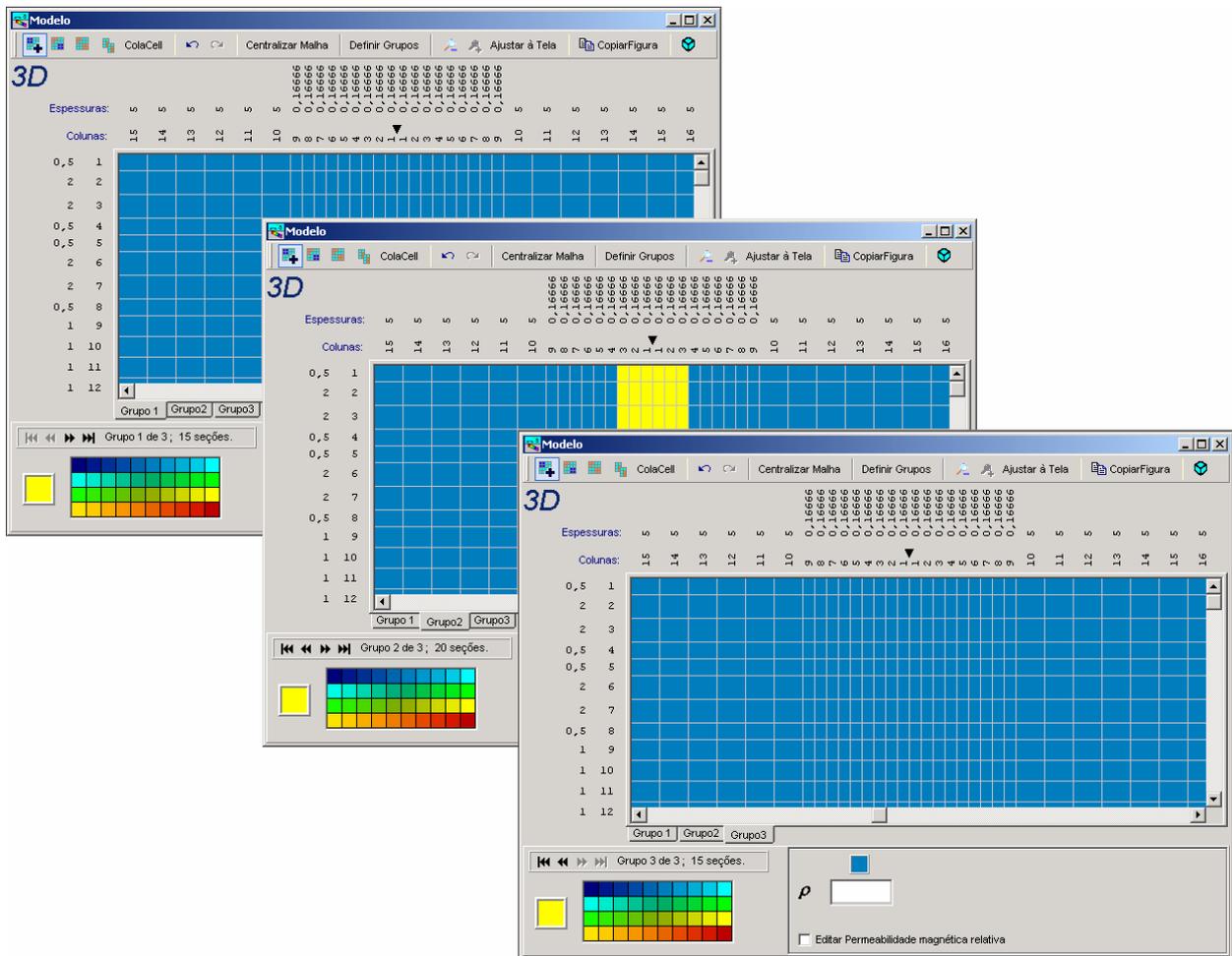


Figura 3.7: Janela usada para criar e modificar modelos tridimensionais. A janela é a mesma da Figura 3.1, mas com ligeiras alterações. Para facilitar o entendimento, aqui as seções são mostradas em janelas separadas, porém estas seções são alternadas em uma mesma janela.

3.1.2.2 – Facilidade para visualização de modelos geofísicos tridimensionais

Nas versões anteriores da eGs os modelos geofísicos tridimensionais só poderiam ser visualizados através de seções bidimensionais que representam um grupo de fatias do modelo total. Cada seção representa um corte perpendicular ao eixo Y, sendo que o plano da seção é o XZ. Para facilitar a visualização do modelo a eGs 3.0 possibilita a visualização do mesmo em três dimensões. Para ilustrar o uso desta ferramenta serão analisadas as geometrias de três modelos. Os dois primeiros (*modelo 1* e *modelo 2*) são modelos tipicamente usados no método MCSEM (Marine Controlled Source Electromagnetic), o qual é um método geofísico empregado na

exploração de hidrocarbonetos em águas profundas. O método MCSEM usa com fonte um dipolo elétrico horizontal móvel e um conjunto de receptores distribuídos no fundo do mar para medir as componentes do campo elétrico. O terceiro modelo (*modelo 3*) é mais comum na área de prospecção mineral.

Antes de ilustrar o uso desta ferramenta, será mostrado, na janela de implementação de modelos geoeletricos, o modelo que será usado para tal fim. A Figura 3.8 mostra as seções do *modelo 1*, tipicamente usado no método MCSEM.

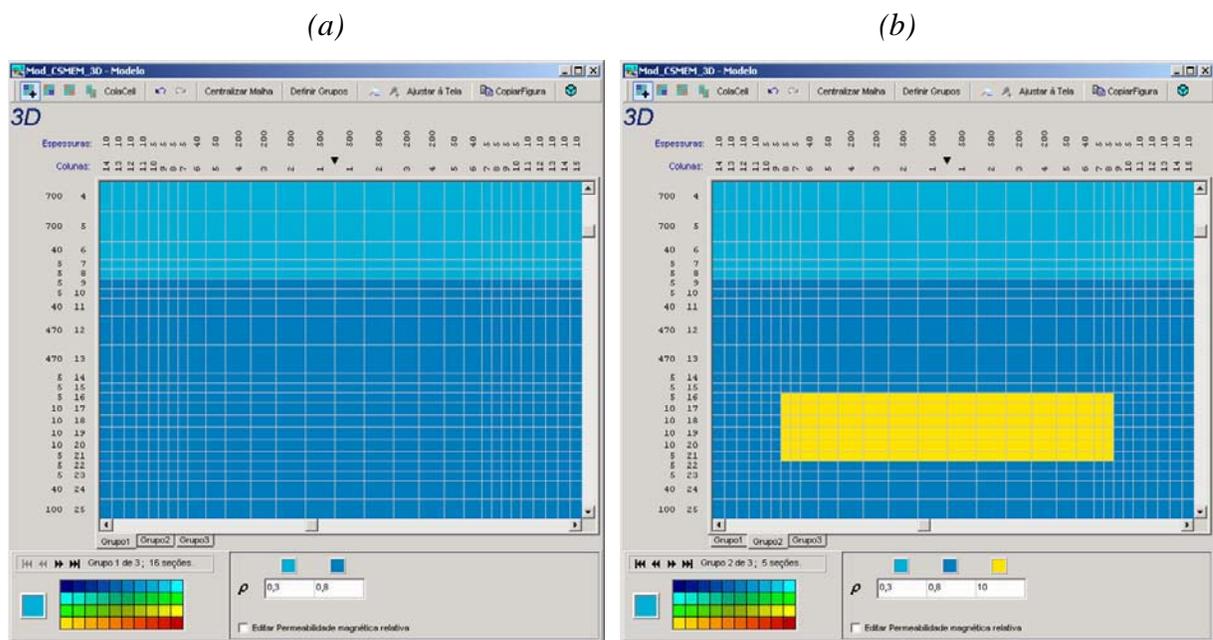


Figura 3.8: Seções bidimensionais do *modelo 1* (tridimensional). A figura (a) representa a primeira e terceira seção (somente o meio estratificado). A figura (b) representa a segunda seção (meio estratificado e corpo juntos).

Agora vamos visualizar este modelo em três dimensões. Esta interface permite visualizar o modelo de duas formas: por *cortes* ou por *cores selecionadas*.

A opção para *visualizar por cortes* permite enxergar o interior do modelo por meio de cortes nos três eixos. A Figura 3.9 ilustra o uso dessa opção.

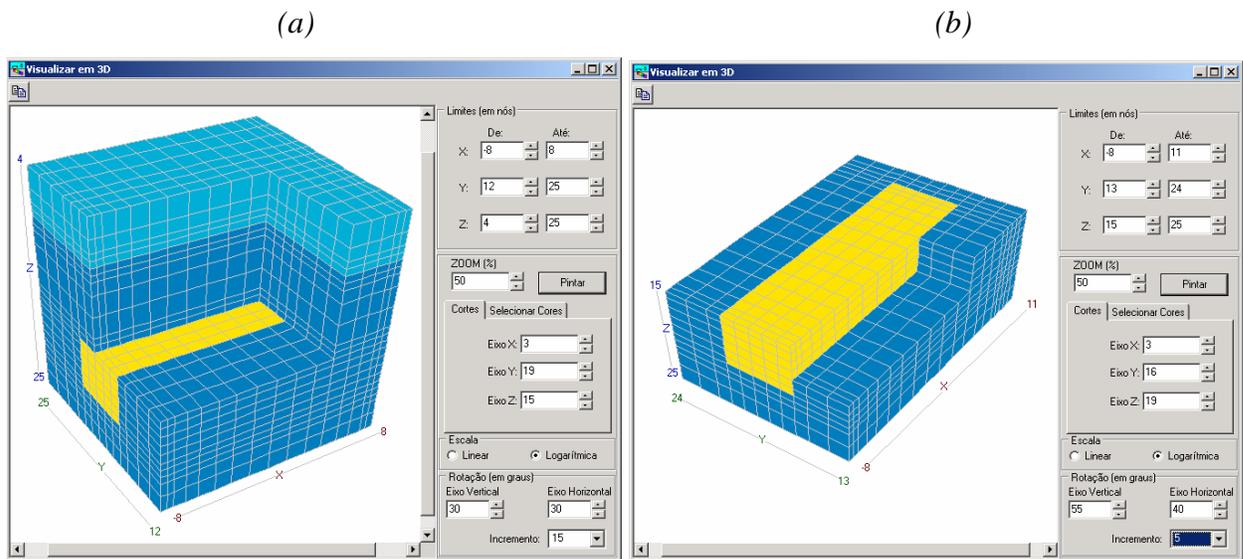


Figura 3.9: Visualização do *modelo 1* em 3D através de *cortes* nos eixos. A figura (a) exibe o trecho do modelo em que os nós variam do nó 4 ao nó 25 na direção Z. Na figura (b) a variação na direção Z vai do nó 15 (no topo do corpo) ao nó 25 (um pouco abaixo do corpo).

Além de poder visualizar o modelo por *cortes*, o usuário pode visualizar somente as células de cores previamente escolhidas. Para isso, basta acionar a opção *visualizar por cores selecionadas* e escolher as cores das células as quais se deseja visualizar. A Figura 3.10 mostra como fica a visualização.

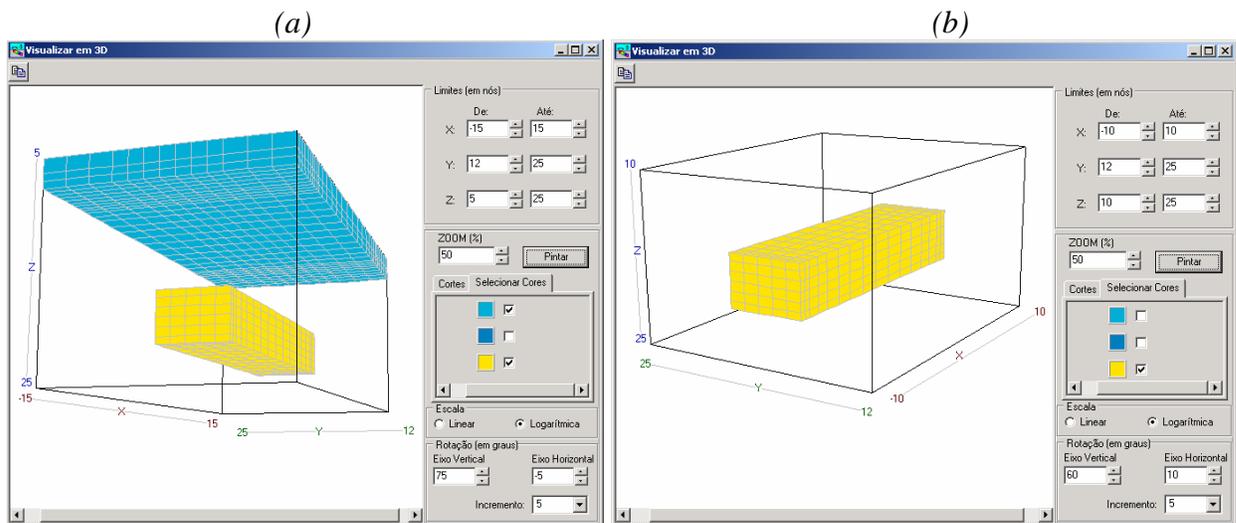


Figura 3.10: Visualização das células do *modelo 1* através de cores selecionadas. Na figura (a) são mostradas as células das cores amarelo e azul claro. A figura (b) mostra somente as células de cor amarela. Nas figuras (a) e (b) as células da rocha encaixante (cor azul) ficaram invisíveis (é bom lembrar que as cores estão associadas aos valores de resistividades de cada parte do modelo).

O *modelo 2* é semelhante ao *modelo 1*, porém contém dois corpos ao invés de um só. A Figura 3.11 mostra a segunda seção do *modelo 2*, onde está localizado o corpo (as seções 1 e 3, que contêm o meio estratificado, são do mesmo tipo do *modelo 1*).

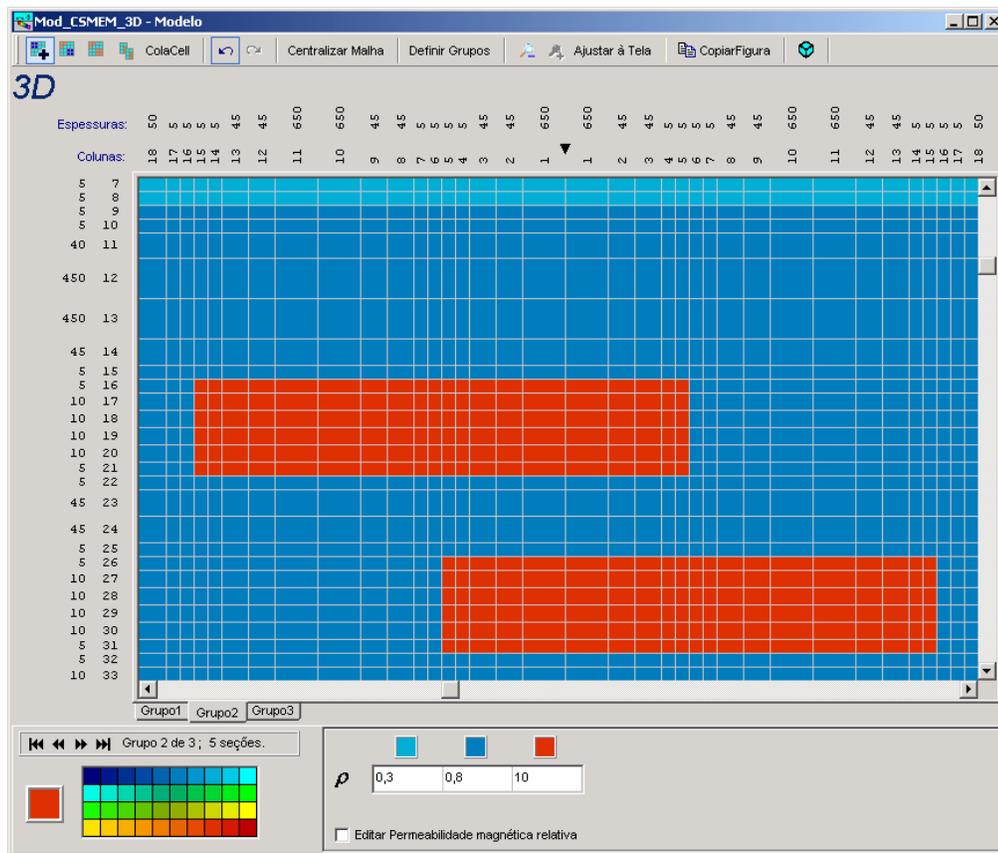


Figura 3.11: Segunda seção bidimensional do *modelo 2* (tridimensional). As seções 1 e 3, que representam o meio estratificado, são do mesmo tipo da Figura 3.8(a).

Como no *modelo 1*, na seqüência temos a visualização do modelo tridimensionalmente. Em primeiro lugar vem a visualização por *cortes* ilustrada na Figura 3.12.

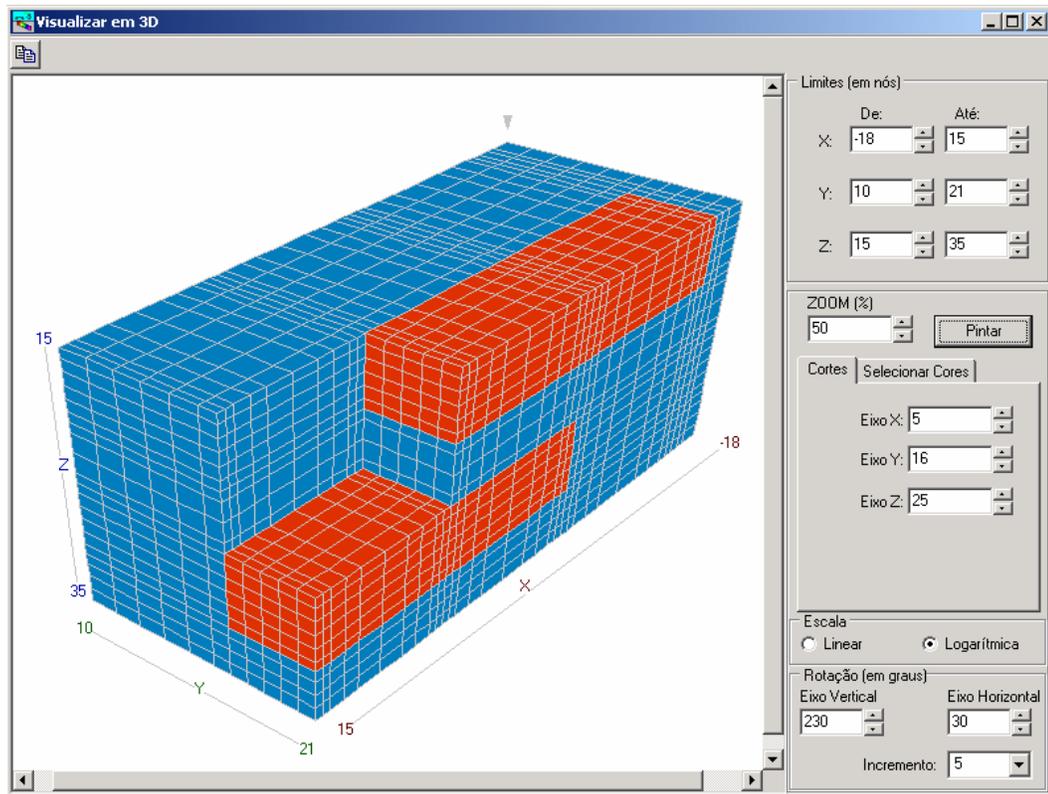


Figura 3.12: Visualização do *modelo 2* por *cortes*. Aqui são exibidos a rocha encaixante (células de cor azul) e os corpos (células de cor vermelha). Para facilitar a visualização foram feitos cortes nas extremidades dos corpos.

A seguir temos a visualização feita por *cores selecionadas*. A Figura 3.13 exhibe somente as células da cor vermelha (células do corpo)

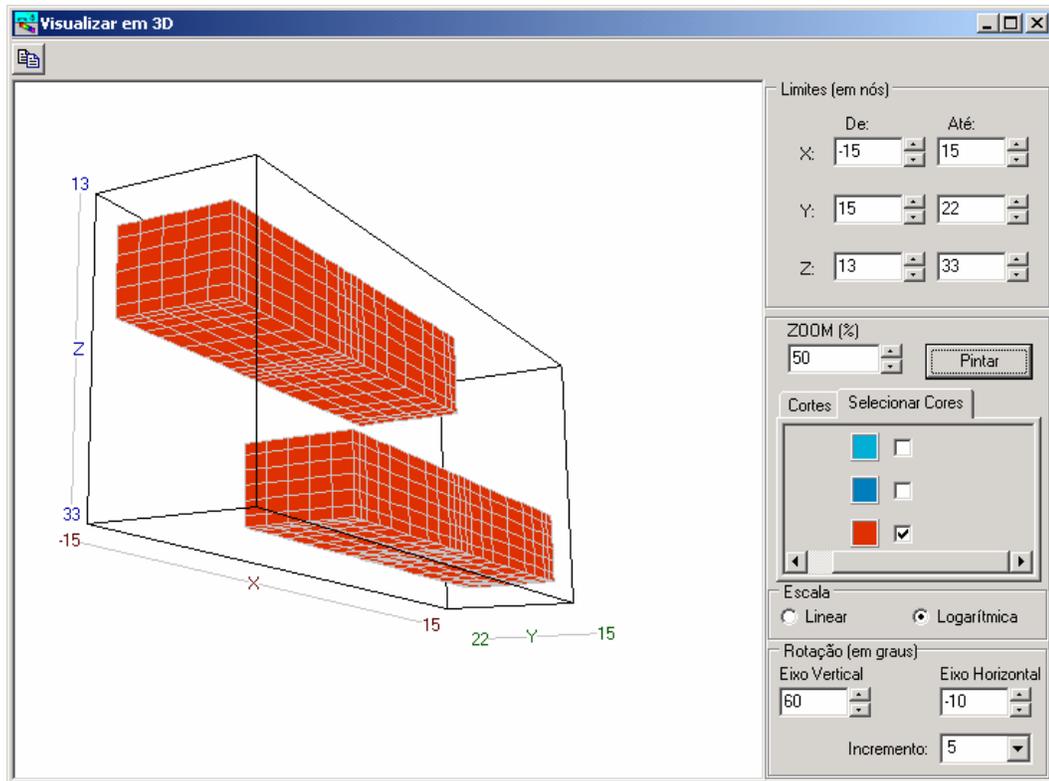
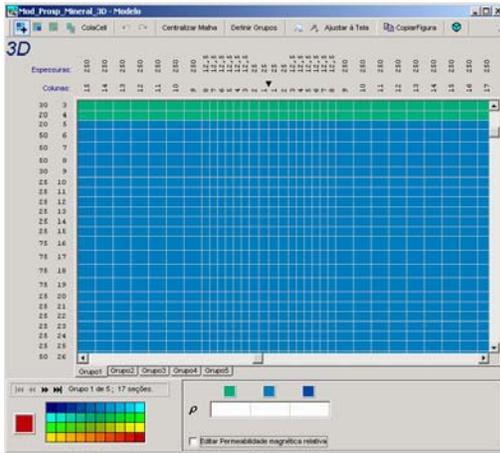


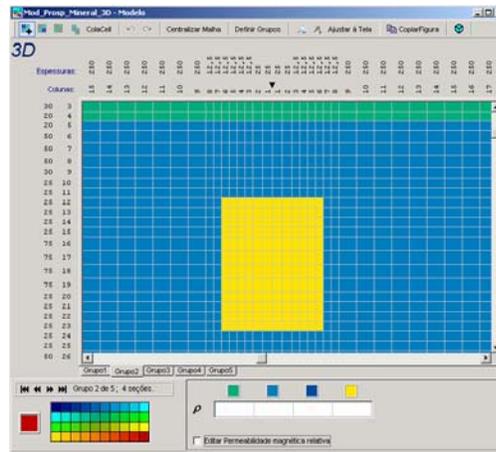
Figura 3.13: Visualização do *modelo 2* por *cores selecionadas*. As células da rocha encaixante (cor azul) ficaram transparentes (invisíveis). As células de cor azul claro não aparecem neste intervalo do modelo.

O *modelo 3* é formado por duas camadas e por um corpo que contém um corpo interno e uma auréola externa, simulando o halo de um corpo mineralizado. A Figura 3.14 mostra as seções deste modelo.

(a)



(b)



(c)

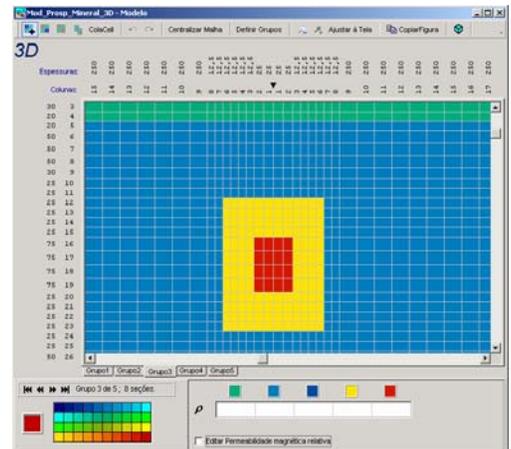
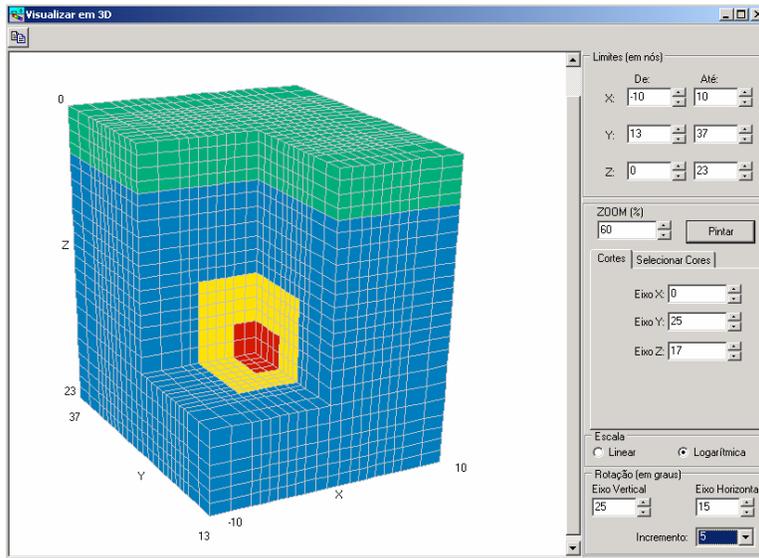


Figura 3.14: Seções bidimensionais do *modelo 3* (tridimensional). A figura (a) ilustra a primeira e a quinta seção do modelo (somente o meio estratificado). A figura (b) ilustra a segunda e a quarta seção do modelo (aqui ainda não aparece o corpo interno). A figura (c) mostra a terceira seção (aqui aparece o corpo interno).

Como nos *modelos 1* e *2*, na seqüência temos a visualização do modelo tridimensionalmente. Em primeiro lugar vem a visualização por *cortes* ilustrada na Figura 3.15.

(a)



(b)

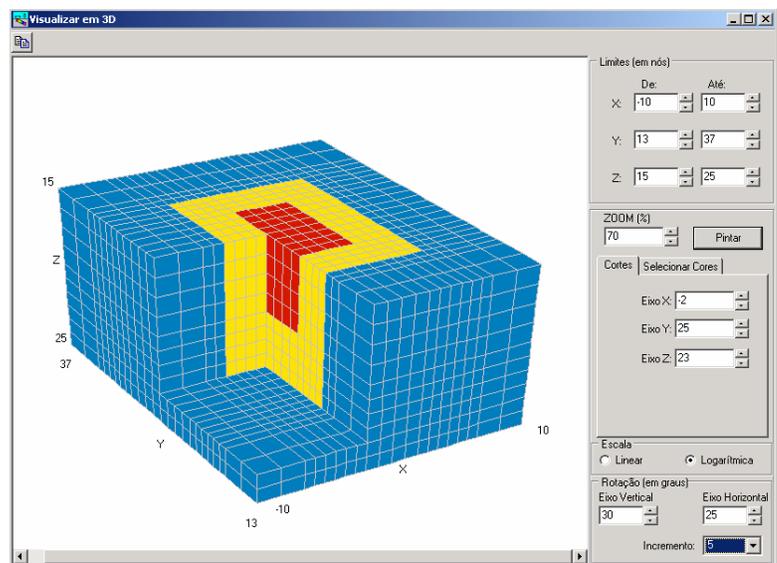
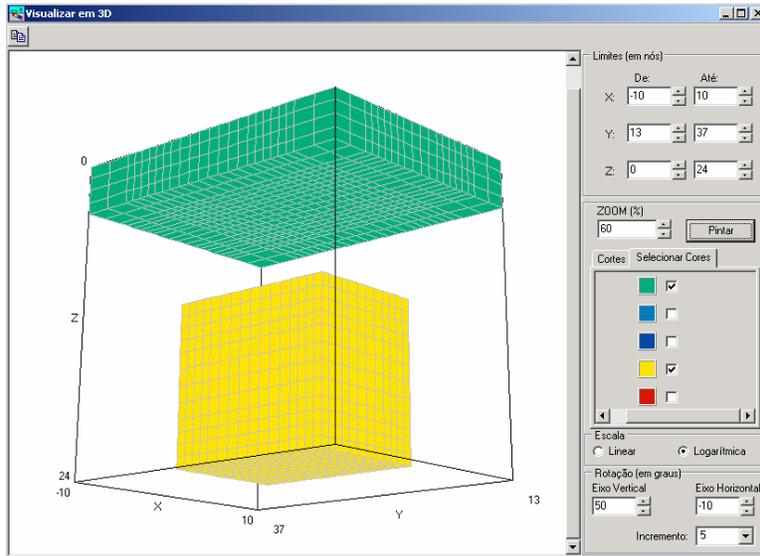


Figura 3.15: Visualização do *modelo 3* por *cortes*. A figura (a) exhibe as camadas e o interior do corpo. A figura (b) ilustra mais claramente a distribuição do corpo através de um corte feito no topo do corpo interno (células vermelhas).

A seguir temos a visualização feita por *cores selecionadas*. A Figura 3.16 ilustra o modelo por dois pontos de vista.

(a)



(b)

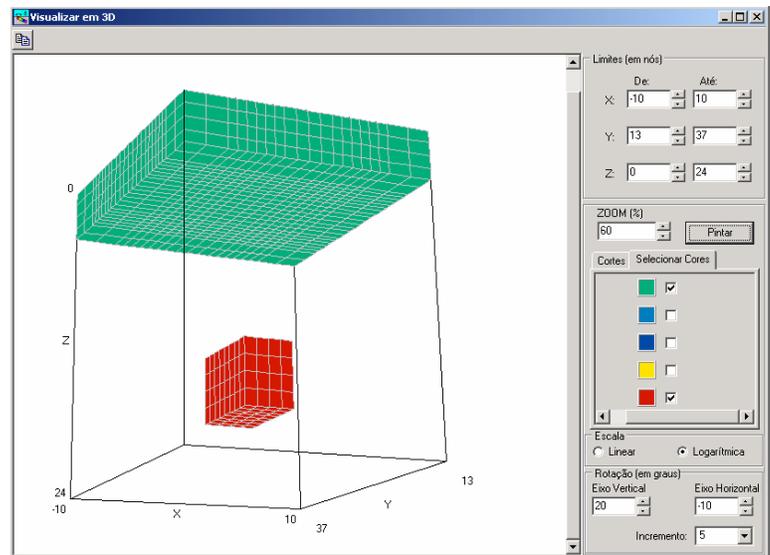


Figura 3.16: Visualização do modelo 3 por cores selecionadas. Na figura (a) as células de cor azul e as de cor vermelha ficaram invisíveis. Na figura (b) as células de cor azul e as de cor amarela ficaram invisíveis.

Analisando as figuras que mostram os modelos geofísicos (*modelos 1, 2 e 3*) tridimensionalmente, fica claro que há casos em que a visualização por *cortes* é mais conveniente que a visualização por *cores selecionadas*, e vice-versa. Nos *modelos 1 e 2* (Figuras 3.8 e 3.11) a visualização por *cores selecionadas* (Figuras 3.10 e 3.13) se mostrou mais eficaz que a visualização por *cortes* (Figuras 3.9 e 3.12). Já no *modelo 3* (Figura 3.14), ocorreu o contrário. A

visualização por *cortes* (Figura 3.15) se mostrou mais eficiente que a visualização por *cores selecionadas* (Figura 3.16). Aplicando-se esses dois modos de visualização tem-se melhor controle da forma e característica de um modelo 3D.

3.2 – FACILIDADE PARA VISUALIZAR OS DADOS DE SAÍDA

Depois de construir o modelo geoeétrico e executar a modelagem numérica, chega o momento de avaliar a sua resposta (dados de saída). Precisamos saber se o modelo corresponde a uma boa aproximação das características geoeétricas da subsuperfície ou se precisa ser modificado. Para fazer este tipo de análise, o *módulo de saída* da eGs 3.0 possui dois grupos de interfaces. O primeiro grupo é composto pela interface para visualização de gráficos de curvas. O segundo grupo é composto pelas interfaces para visualização de imagens. A este segundo grupo pertencem as interfaces para visualização de mapas de contorno e para visualização de pseudo-seções.

3.2.1 – Interface para visualização de gráficos de curvas

A interface da eGs que permite comparar os dados da resposta do modelo geoeétrico com dados observados tem diversas opções. As opções escolhidas para fazer os gráficos podem ser manuais ou automáticas. Neste último caso, que é o caso padrão, o gráfico é gerado automaticamente após o cálculo da resposta do modelo. Dentre outras opções à disposição do usuário tem-se: quantidade de gráficos; escala linear ou logarítmica e intervalos de visualização de gráficos.

3.3.1.1 – Quantidade de gráficos

Os arquivos que contêm os dados observados e os que contêm os dados da resposta do modelo geoeétrico são, em geral, na forma de uma matriz cuja primeira coluna é plotada no eixo x , e as demais são plotadas no eixo y . Estas últimas, normalmente, aparecem aos pares (resistividade aparente e fase; elipticidade e ângulo de inclinação, amplitude e fase, etc), por isso a interface permite visualizar um, dois ou quatro gráficos de uma só vez, permitindo assim avaliar

os tipos de dados isoladamente ou aos pares, em uma única janela ou em janelas separadas. A Figura 3.17 ilustra quatro gráficos em uma só janela.

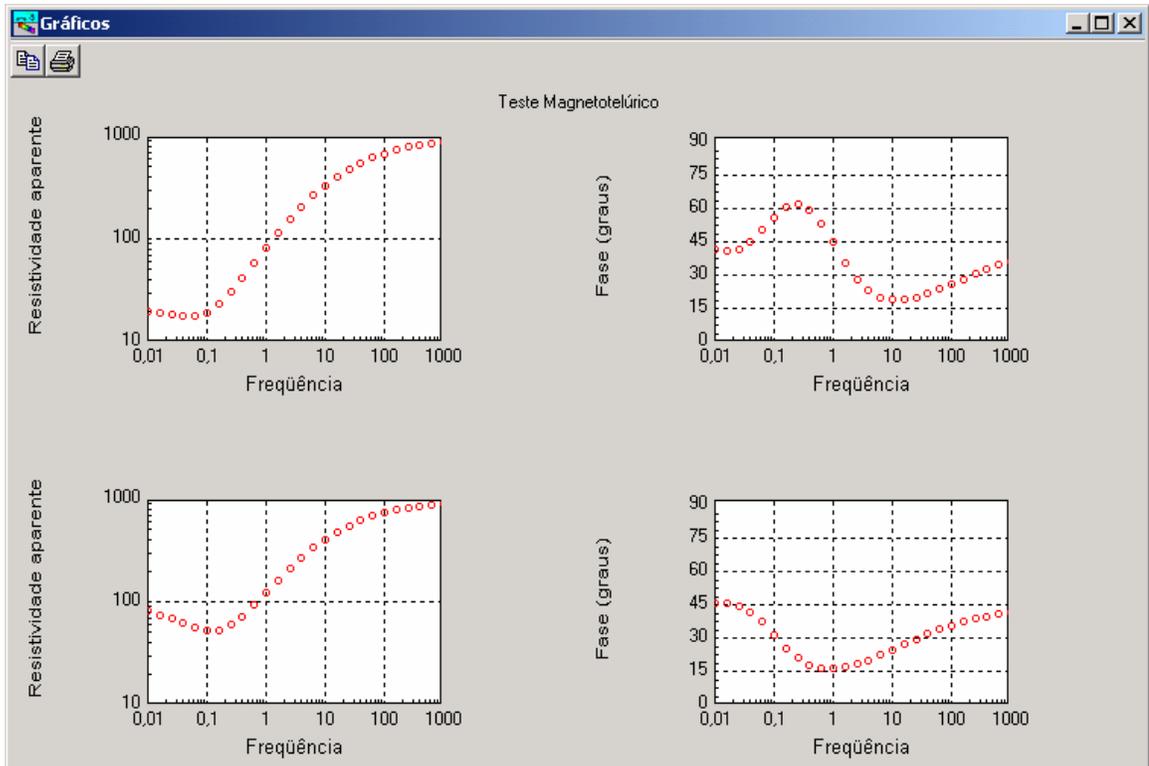
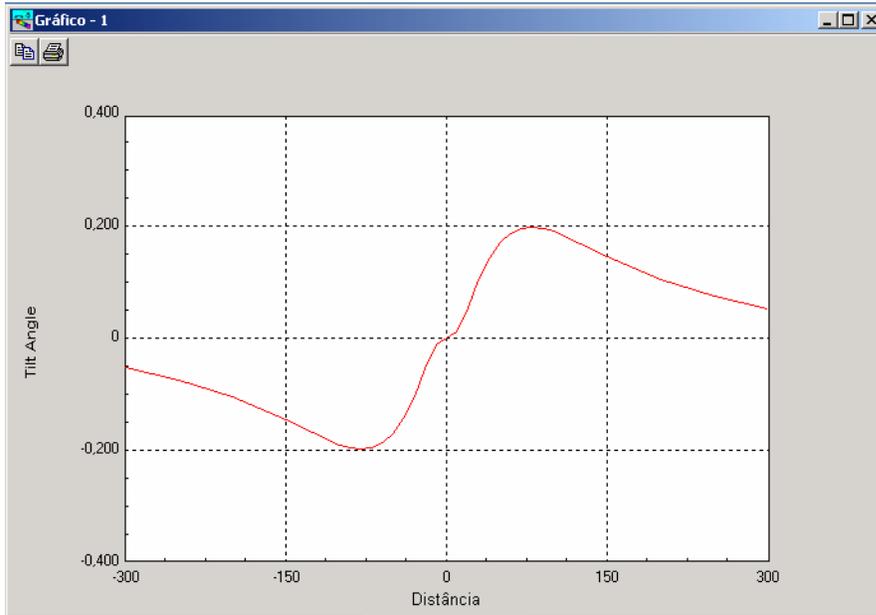


Figura 3.17: Ilustração de quatro gráficos de uma só vez.

3.2.1.2 – As escalas dos gráficos

Esta interface permite fazer gráficos de três tipos com relação às escalas usadas em cada eixo: Linear-Linear (linear nos eixos x e y), Log-Linear (logarítmica no eixo x e linear no eixo y) e Log-Log (logarítmica nos eixos x e y). A Figura 3.18 mostra gráficos dos tipos Linear-Linear e Log-Log.

(a)



(b)

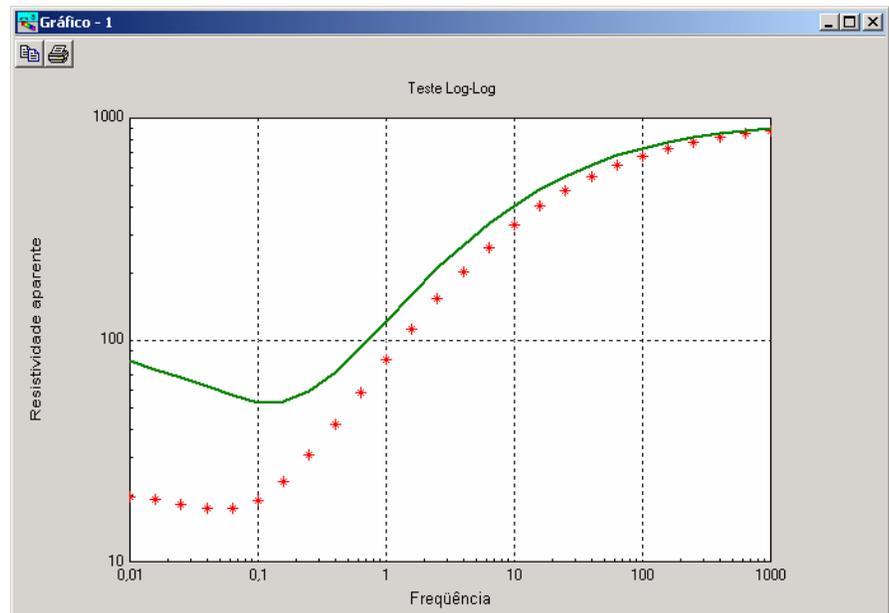


Figura 3.18: Ilustração de gráficos do tipo Linear-Linear (a) e Log-Log (b).

As escalas usadas nos gráficos podem ser escolhidas manualmente pelo usuário, ou escolhidas automaticamente pela interface. Para a escolha automática basta usar a opção *limites automáticos*.

3.2.1.3 – Os intervalos de visualização dos gráficos

Para visualizar os gráficos o usuário pode escolher os intervalos de visualização que desejar. Porém, se preferir visualizar os gráficos por completo, e na escala que melhor os represente, basta usar a opção *Limites Automáticos*. Assim o usuário não precisa se preocupar em ajustar os intervalos de visualização dos gráficos a cada alteração feita nos mesmos. A Figura 3.19 mostra, respectivamente, um mesmo gráfico feito com e sem a ativação desta opção.

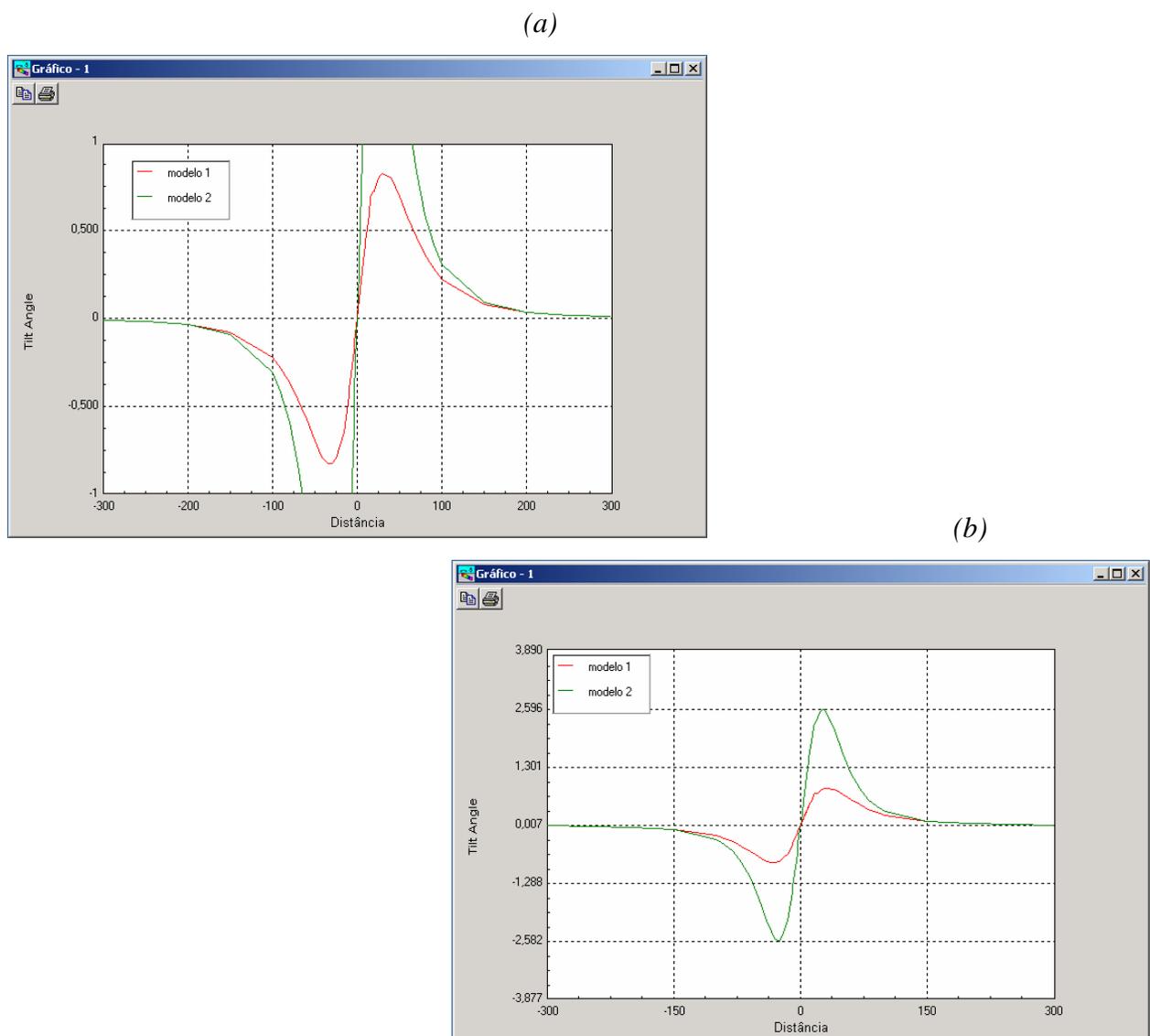


Figura 3.19: Ilustração do funcionamento da opção *Limites Automáticos*. No gráfico da figura (a) não foi usada a opção, enquanto que no da figura (b) foi usada.

3.2.2 – Interface para visualização de mapas de contorno

Para casos em que a resposta do modelo está distribuída de forma bidimensional (como por exemplo o campo elétrico no fundo do mar de dados MCSEM), a eGs dispõe de uma interface para visualizar estes dados na forma de mapas de contorno. Esta interface possui a opção de visualização dos dados em escala linear ou logarítmica. A Figura 3.20 exibe dados através de mapa de contorno em escala linear e logarítmica.

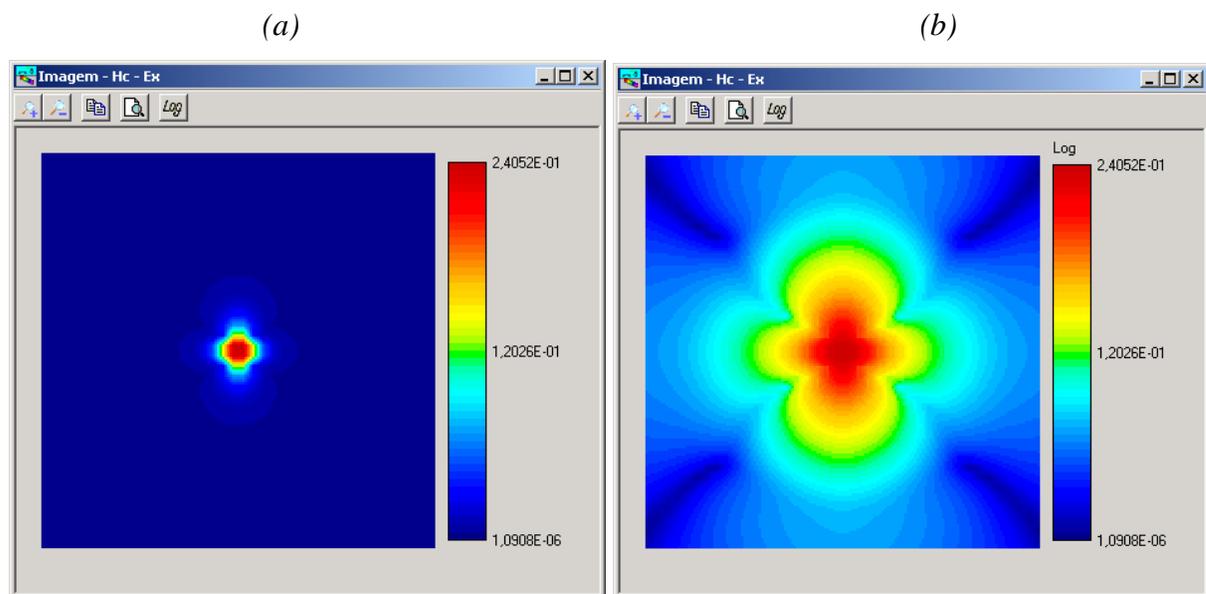


Figura 3.20: Visualização da resposta do modelo através de mapas de contorno: (a) dados em escala linear; (b) dados em escala logarítmica.

3.2.3 – Interface para visualização de pseudo-seções

Para facilitar a análise dos dados observados e os de resposta do modelo geoeletrico do tipo IP esta interface oferece duas opções para visualizar estes dados com base num espectro de cores: visualizando os dados *brutos* e visualizando os dados *suavizados*.

Os dados *brutos* são os declarados nas células do editor de pseudo-seções da interface, e que são provenientes de arquivos [de dados de campo (observados) ou de resposta do modelo geoeletrico] ou digitados diretamente (Pereira, 2005). Estes dados podem ser visualizados com

base num espectro de cores que facilita identificar como os valores estão distribuídos. A Figura 3.21 exibe os dados ainda no editor de pseudo-seções e os dados identificados por cores.

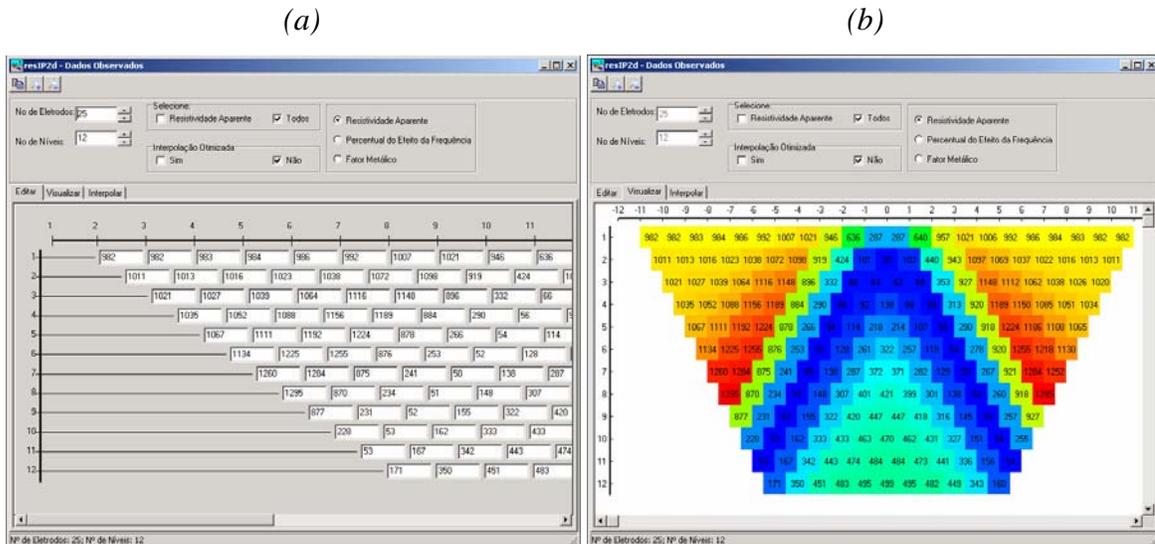


Figura 3.21: Visualização da resposta do modelo ou dos dados observados do tipo IP. A figura (a) mostra os dados ainda no editor de pseudo-seções e a figura (b) mostra a distribuição dos dados identificados por cores (o menor valor é identificado pela cor azul e o maior pela cor vermelha).

Há casos em que a visualização de dados *suavizados* facilita a sua interpretação. Por isso, a interface dispõe de uma opção para gerar uma figura desse tipo através de interpolação linear. A Figura 3.22 mostra a Figura 3.21(b) de forma contínua.

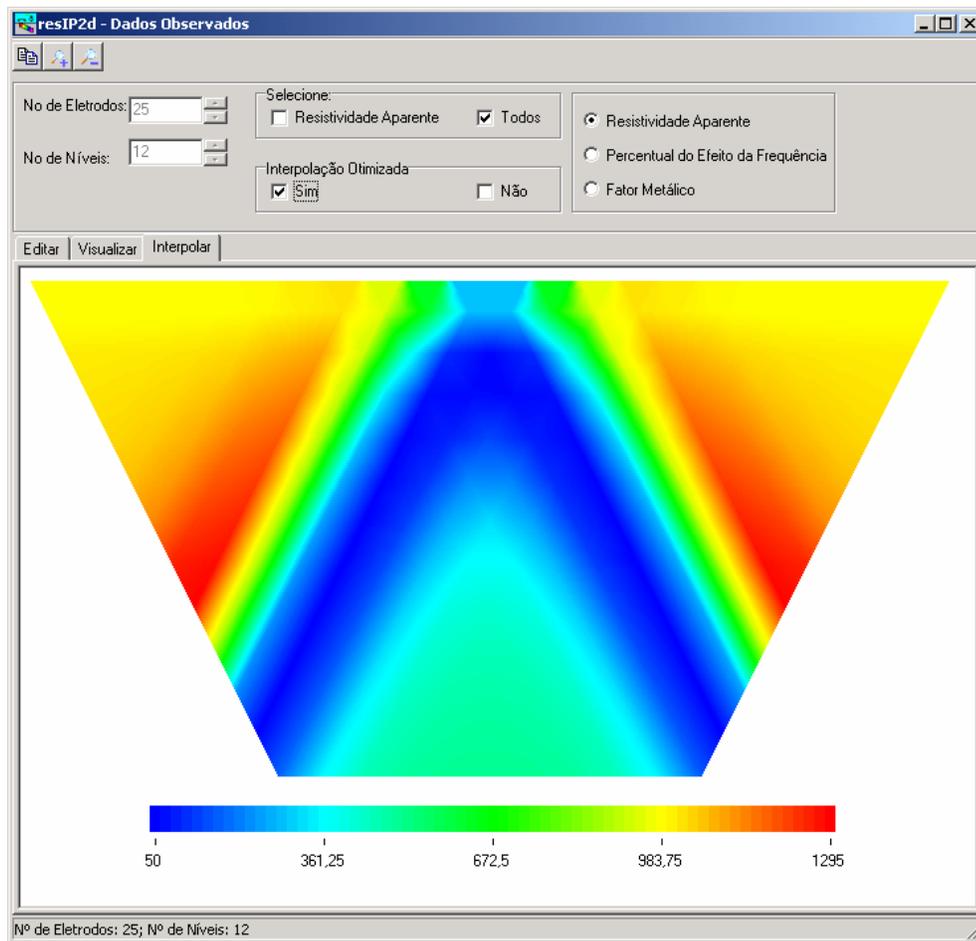


Figura 3.22: Suavização dos dados do tipo IP. Esta figura mostra a suavização dos dados mostrados na Figura 3.21(b).

3.2.4 – Interface para gerenciamento de arquivos de saídas do método MCSEM

Os dados de saída do método MCSEM (Marine Controlled Source Electromagnetic) são apresentados normalmente através de mapas de contorno que mostram a distribuição das componentes do campo elétrico nas direções X, Y e Z separadamente, e normalizadas pelos campos gerados na ausência de hidrocarbonetos na subsuperfície. Porém, há casos em que se torna mais conveniente visualizar os mapas de contorno de dados sem normalização, ou somente dos dados na ausência de hidrocarbonetos. A eGs dispõe de uma interface para que o usuário escolha os arquivos que contêm os dados gerados a partir de modelos na presença e na ausência de hidrocarbonetos, quais desses dados (de modelos na presença de hidrocarbonetos, na ausência dos mesmos, ou dos primeiros normalizados pelos últimos) e componentes do campo elétrico

serão visualizados. A Figura 3.23 mostra a janela que permite fazer as escolhas das opções apresentadas acima.

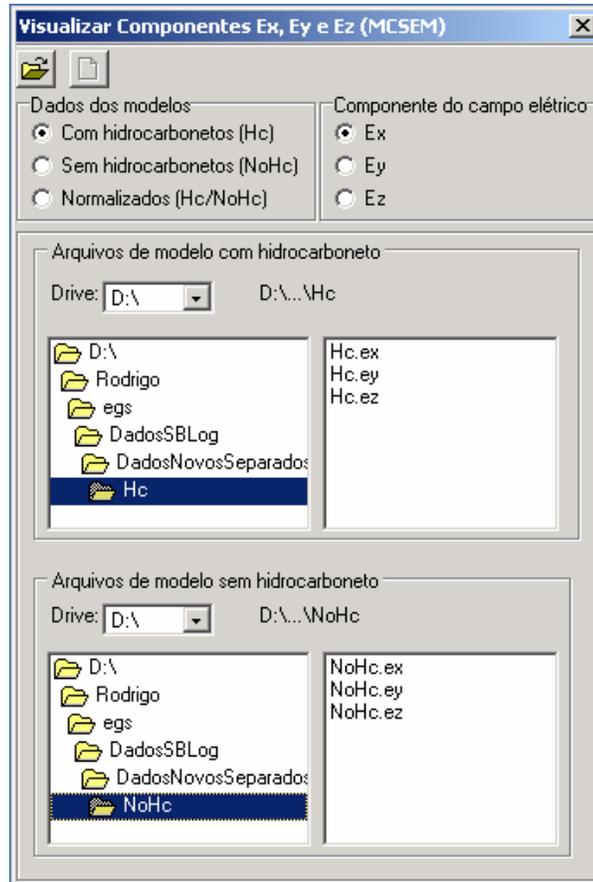


Figura 3.23: Janela que permite escolher os arquivos, o modo de apresentação e a componente do campo elétrico dos dados de saída do método MCSEM.

3.3 – AUTOVERIFICAÇÃO DE ERROS

Uma interface que propõe facilitar a geração, a visualização e a interpretação de modelos não pode deixar de alertar ao usuário os erros cometidos na entrada de dados. Por isso a eGs 3.0 avisa seus usuários dos possíveis erros cometidos por escolhas de opções incorretas ou de preenchimento incorreto de campos, procurando sempre que possível mostrar uma alternativa para corrigir os erros, ou em alguns casos corrigir automaticamente. A Figura 3.24 mostra uma mensagem de alerta mostrada quando há um preenchimento incorreto dos campos que determinam as dimensões da malha.



Figura 3.24: Alerta dado ao usuário em relação ao preenchimento incorreto dos campos onde são determinadas as dimensões da malha de elementos finitos.

Em alguns casos os dados podem até ter sido declarados corretamente, mas o alerta é dado para que o usuário esteja ciente dos valores declarados. A Figura 3.25 dá um exemplo desse tipo de atitude.

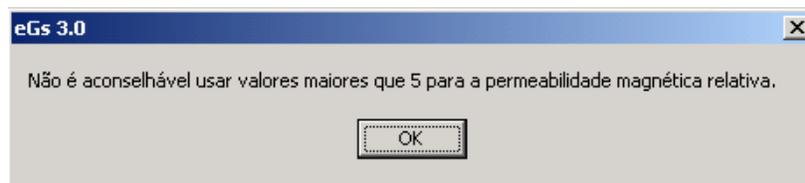


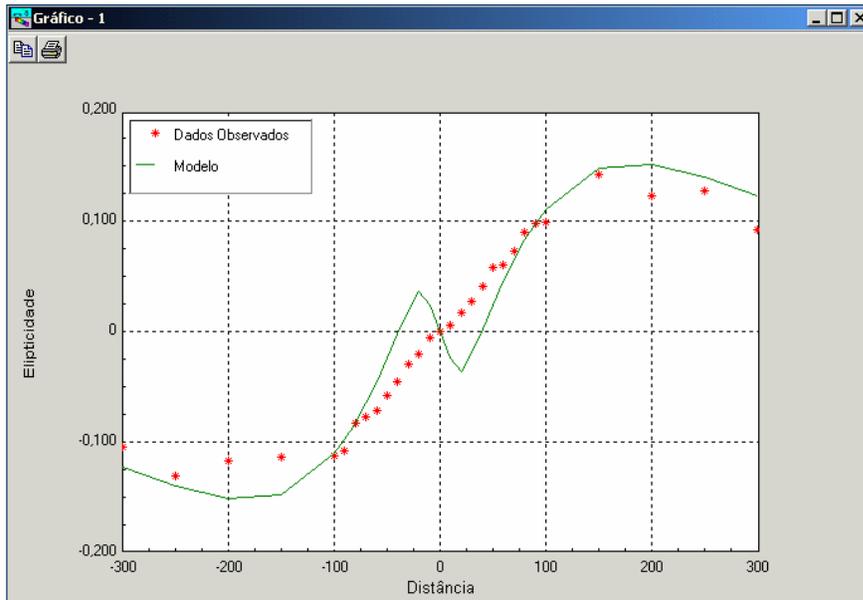
Figura 3.25: Alerta dado ao usuário em relação ao possível preenchimento incorreto dos campos onde são declaradas as propriedades físicas do modelo geológico.

3.4 – UM PROGRAMA QUE NÃO DEPENDA DE OUTROS PROGRAMAS

Uma das características fundamentais da eGs 3.0 no seu projeto de desenvolvimento é a de ser um programa independente de outros programas, o que implica basicamente em ter a sua própria interface de entrada e apresentação dos dados de saída. A interface gráfica para visualização dos dados de saída é a peça da eGs 3.0 que a torna independente de outros programas de visualização, e que tem as suas ferramentas construídas em função do restante da interface e dos arquivos gerados e usados na eGs. O item 3.2 mostra algumas das opções desta interface. Para os usuários que mesmo assim preferirem usar outros programas para fazer seus gráficos, é possível usar os dados gerados pela eGs, já que os mesmos são guardados em arquivo.

Para fazer uma modelagem usando a eGs 3.0, o usuário deve executar os passos a seguir (os detalhes desta seqüência de passos são encontrados no Apêndice A, em que o item A.3.2.1 dá um exemplo do método VLF). Primeiramente, o usuário deve escolher o método geofísico elétrico ou eletromagnético que será usado na modelagem. Neste exemplo deve-se escolher dimensão 2D, método VLF e modelo direto. A seguir, o usuário deve declarar o arquivo de dados observados. Este passo é fundamental porque este arquivo normalmente contém duas ou mais colunas. Na primeira coluna têm-se informações sobre as posições das estações de observação, e nas demais colunas os valores dos dados observados. Na modelagem de inversão todas as colunas são necessárias porque as demais colunas são usadas durante o processo de inversão. Na modelagem direta apenas a primeira coluna é usada, as demais colunas não têm nenhum efeito. Em seguida, deve ser declarado o modelo geofísico. O usuário pode criar um modelo novo ou abrir um já existente em arquivo, podendo modifica-los com as ferramentas que permitem modificar as propriedades elétricas e as dimensões das células. A Figura 3.26 mostra o modelo geofísico na janela que permite modifica-lo e na janela que permite visualiza-lo tridimensionalmente. Posteriormente, deve-se salvar o modelo e executar o programa, para que seja gerado o arquivo que contém a resposta do modelo geofísico. Finalmente, o usuário pode avaliar a resposta do modelo geofísico através de gráficos. Se os gráficos não aparecerem automaticamente após se clicar em executar, o usuário deve abrir o arquivo que contém os dados da resposta do modelo geofísico e plotar os gráficos manualmente. Os gráficos das respostas dos modelos geofísicos junto com o dos dados observados estão ilustrados nas Figuras 3.27, 3.28 e 3.29.

(a)



(b)

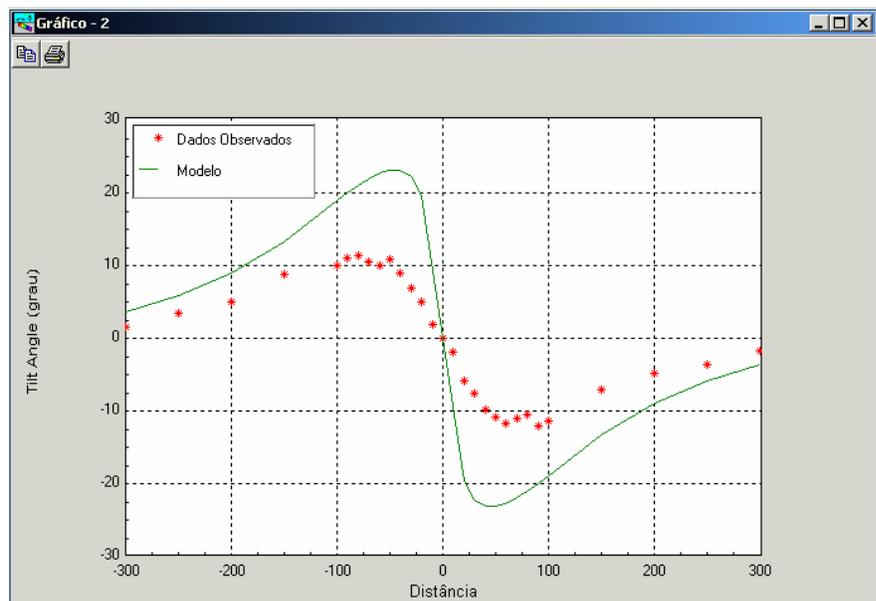
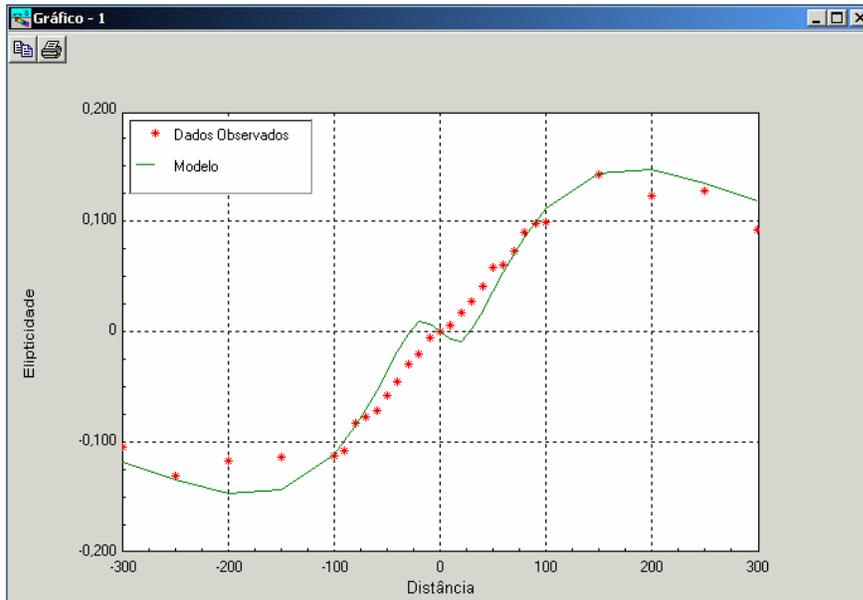


Figura 3.27: Resposta do modelo com $h = 10$ metros (linha verde) e sua comparação com os dados observados (asteriscos vermelhos).

(a)



(b)

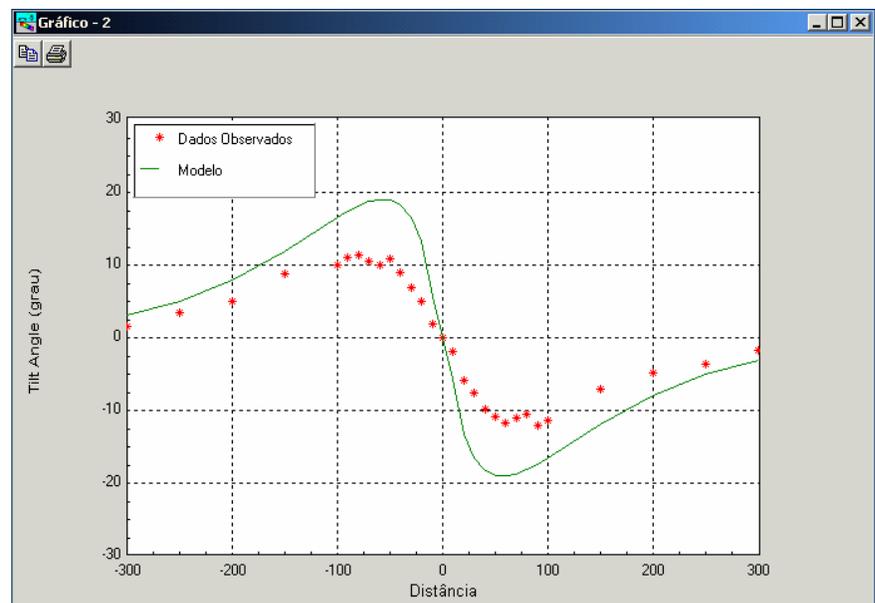
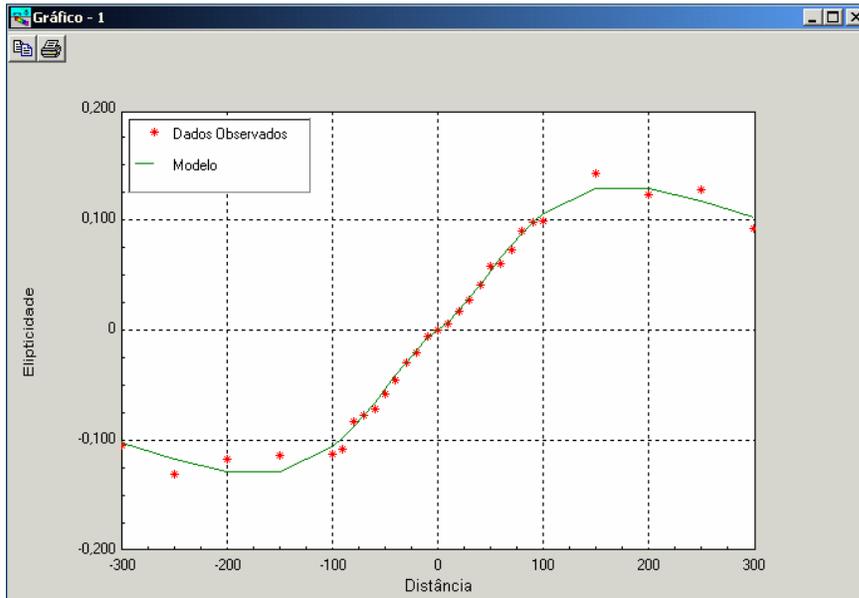


Figura 3.28: Resposta do modelo com $h = 20$ metros (linha verde) e sua comparação com os dados observados (asteriscos vermelhos).

(a)



(b)

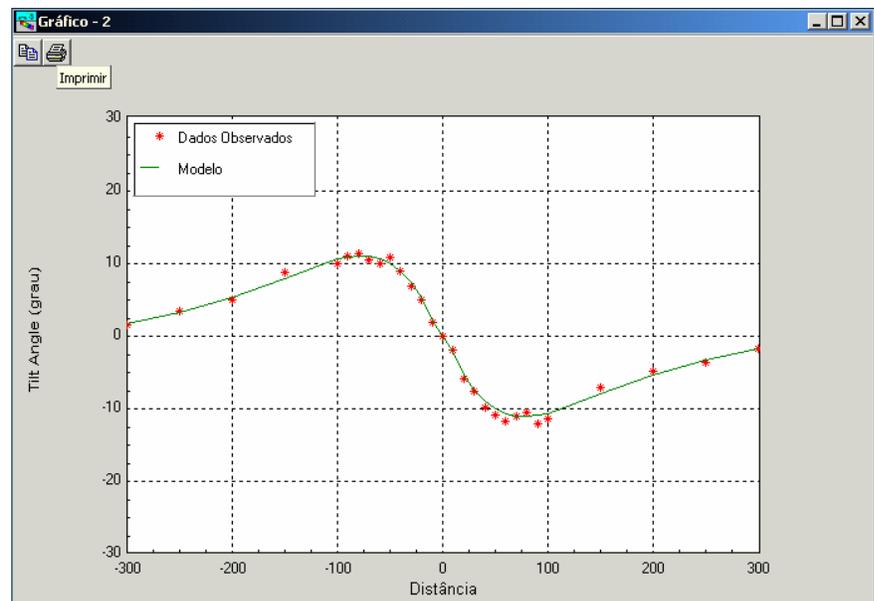


Figura 3.29: Resposta do modelo com $h = 50$ metros (linha verde) e sua comparação com os dados observados (asteriscos vermelhos).

A modelagem descrita acima nos permite tirar algumas conclusões a respeito da distribuição de propriedades geolétricas na subsuperfície que supostamente produziu os dados observados. Nos casos em que a resposta do modelo geofísico não se ajustou aos dados observados ($h = 10$ metros e $h = 20$ metros ilustrados nas Figuras 3.27 e 3.28, respectivamente), pode-se concluir que a distribuição de propriedades geolétricas da subsuperfície é diferente das

mostradas nos modelos que geraram estas respostas. Já no caso em que a resposta do modelo geofísico se ajustou aos dados observados ($h = 50$ metros ilustrado na Figura 3.29) podemos concluir que este modelo (caracterizado por uma distribuição de propriedades geoelétricas da subsuperfície) é uma candidata a solução do problema de determinação desta distribuição de propriedades geoelétricas [não podemos garantir que esta é a única solução do problema, pois o problema geofísico é mal-posto (Silva et al, 2001)]. Para saber se este modelo é de fato a solução (simplificada por se tratar de um modelo) deste problema, se faz necessário o uso de informações adicionais (informações a priori) sobre a estrutura geológica da localidade onde supostamente os dados observados foram coletados.

De maneira análoga a este exemplo, a eGs trata dos outros métodos geofísicos elétricos e eletromagnéticos. O manual do usuário no Apêndice A, mostra outros exemplos envolvendo outros métodos geofísicos elétricos e eletromagnéticos.

4 – INFORMAÇÕES AO PROGRAMADOR

4.1- INTRODUÇÃO

Como já foi dito antes, este trabalho consta de três partes. A primeira (Capítulos 1, 2 e 3) e a terceira (Apêndice A) partes se destinam ao usuário final. A segunda parte (Capítulo 4) se destina ao programador que irá dar continuidade ao desenvolvimento e manutenção da eGs. Nesse trabalho de dissertação fizemos um grande esforço para construir uma metodologia eficaz e transparente que venha facilitar o trabalho dos futuros programadores da eGs. Este é um trabalho de bastidores, que só os que estão envolvidos sabem da sua importância. É importante enfatizar que para se usar a eGs não é preciso saber o conteúdo deste capítulo. Para o usuário da eGs, embora o manual no Apêndice A seja, provavelmente, a parte principal dessa dissertação, o Capítulo 4 é, do ponto de vista dos construtores da eGs, a contribuição maior desta dissertação. De fato, este capítulo fornece todas as informações para se dar continuidade a eGs. Lembre-se que um dos objetivos principais dessa dissertação é contornar o problema de se escrever repetidamente, os mesmos códigos de compilação por gerações diferentes de alunos de pós-graduação.

Para um melhor entendimento deste capítulo, é aconselhável construir e rodar alguns modelos geofísicos na eGs 3.0, usando as instruções dadas no Apêndice A, antes de ler este texto, e também ter alguns conhecimentos de programação em Delphi. Neste texto serão abordados tópicos da estrutura de manipulação dos dados, dos métodos usados na pintura das figuras de modelos geofísicos e gráficos, dicas a respeito tanto da estrutura do código fonte quanto das estratégias usadas na construção deste programa e sugestões para o melhoramento do programa. Por se tratar de um texto voltado principalmente para programadores, todas as figuras de janelas mostradas aqui foram capturadas em tempo de projeto. As interfaces gráficas a serem discutidas são as usadas no (a):

- Criação e visualização de malhas de elementos finitos de modelos geoeletricos 2D e 3D;
- Visualização dos dados de resposta do modelo geoeletrico e de campo através de gráficos de curvas;

- Visualização dos dados de resposta do modelo geoeletrico e de campo através de mapas de contorno;
- Visualização de pseudo-seções de IP;
- Gerenciamento de arquivos de saída do método MCSEM.

4.2 – INTERFACE PARA CRIAÇÃO E VISUALIZAÇÃO DE MODELOS GEOFÍSICOS 2D E 3D

Para mostrar o funcionamento desta interface, vamos dividi-lo em quatro etapas:

- Estrutura dos arquivos usados como entrada no processamento dos modelos geofísicos diretos;
- Estrutura de armazenamento dos dados;
- Estrutura gráfica das janelas;
- Estrutura de modificação dos dados.

4.2.1 - Os arquivos de entrada no processamento

Na fase de processamento, a eGs 3.0 usa arquivos de entrada, os quais têm as informações sobre o modelo geofísico, e gera arquivos de saída, que contêm a resposta do modelo geofísico e são comparados aos dados observados para verificar se o modelo é satisfatório. Estes arquivos de entrada são gerados pela interface gráfica e obedecem à estrutura mostrada a seguir (nos exemplos de arquivos, os textos iniciados pelo caractere “!” são comentários, e portanto, não fazem parte do arquivo, e são usados aqui somente para explicar o que significam os dados nas respectivas linhas do arquivo):

4.2.2 - O armazenamento dos dados

Durante a fase de construção de um modelo geofísico 2D ou 3D há algumas variáveis que armazenam os dados que caracterizam esse modelo, e outras usadas somente como ferramenta para facilitar a geração do mesmo.

Dentre as variáveis que caracterizam o modelo geoeletrico, existem as que guardam as dimensões da malha (como, por exemplo, *NoNosHorX_G*, que guarda o número de nós na direção X), as que guardam as propriedades elétricas usadas (*Ro* e *Mi*, que caracterizam a resistividade e a permeabilidade magnética relativa das células, respectivamente), e as que guardam as propriedades físicas em cada célula (*MatrizDasCores(x, z, y)*). A *MatrizDasCores* guarda números que podem variar de 0 a 39, os quais têm uma relação biunívoca com as cores que podem ser usadas no modelo geofísico. Os dados dessa matriz são os mesmos usados nos arquivos gerados pela interface, subtraídos de uma unidade.

Das variáveis usadas como ferramenta na construção do modelo, as que merecem destaque são:

- *MatrizCD(x, z, y)* (CD significa “Corpos Distintos”): armazena números que diferenciam dois ou mais corpos do modelo geofísico, mesmo que tenham as mesmas propriedades elétricas [dois conjuntos de células de mesma cor, e que não têm contato direto um com o outro, têm o mesmo número na *MatrizDasCores* (pois são da mesma cor), mas têm números diferentes na *MatrizCD* (pois são de dois corpos diferentes)];
- *MatrizDesfazerCores*: guarda os dados da *MatrizDasCores* usados antes da última alteração;
- *MatrizDesfazerCD*: guarda os dados da *MatrizCD* usados antes da última alteração;
- *PosDosNos3D*: guarda as coordenadas dos nós calculados para a visualização do modelo em 3 dimensões.

4.2.3 - A estrutura gráfica das janelas

Esta interface é formada por três janelas que ajudam a modificar e visualizar o modelo geofísico desejado. A janela principal é a que permite modificar as propriedades elétricas das células que compõem o modelo geofísico, e tem *links* que permitem chamar as outras duas

janelas. Uma delas tem campos que permitem modificar o tamanho do modelo (número e tamanho das células), enquanto que a outra permite visualizar o modelo na forma de um paralelepípedo. A seguir temos a descrição de cada janela.

4.2.3.1 - Janela para geração de modelos geofísicos 2D ou 3D

A janela na qual o usuário desenha o modelo geofísico (Figura 4.1) é formada basicamente por:

- Uma *DrawGrid* (*DrawGridMalha*) onde as células do modelo são pintadas;
- Uma *DrawGrid* (*DrawGridCores*) onde se escolhe a cor de pintura;
- Uma *StringGrid* (*StringGridRoMi*) onde são declaradas as propriedades elétricas e eletromagnéticas (resistividade e permeabilidade magnética relativa) das células;
- Um *CheckBox* para ocultar ou mostrar os campos de edição (na *StringGridRoMi*) da permeabilidade magnética relativa (ou o PFE no caso do método IP);
- Um *menu* (do tipo *ActionToolBar*, o qual é modificado no formulário principal da eGs 3.0) com ferramentas para ajudar na criação do modelo.

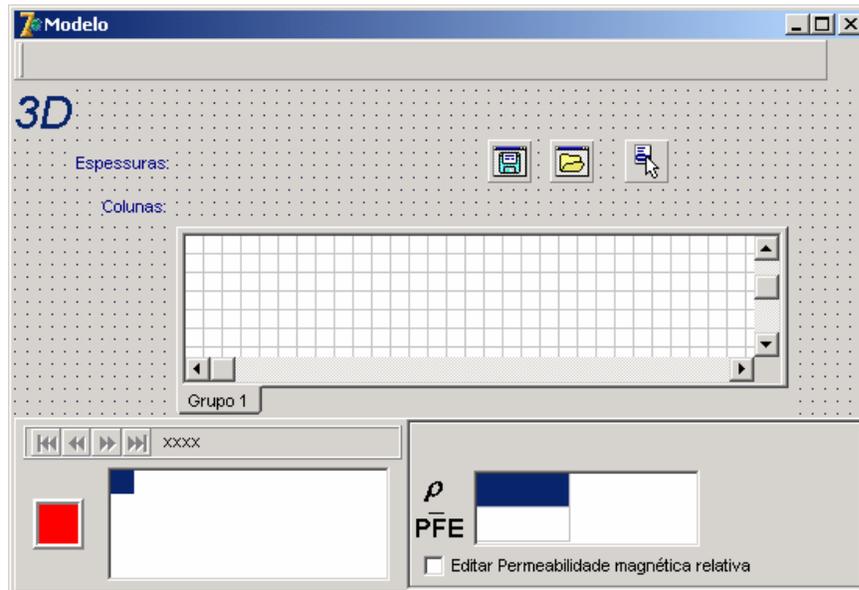


Figura 4.1: Janela usada para gerar modelos geofísicos 2D ou 3D.

4.2.3.2 - Janela para modificar o tamanho do modelo geofísico

Para usuários mais avançados, a eGs 3.0 tem a opção para modificar no modelo geofísico o tamanho e o número de suas células. Esta modificação é feita em uma janela (Figura 4.2) com a seguinte estrutura:

- *Edits* que permitem declarar o número de células do modelo;
- *StringGrids* para declarar os tamanhos das células.

Figura 4.2: Janela usada para modificar o tamanho do modelo.

4.2.3.3 - Janela para visualizar o modelo geofísico em três dimensões

A eGs 3.0 permite que o usuário verifique se o modelo geofísico (principalmente 3D) corresponde ao que foi idealizado. Esta verificação pode ser feita através da visualização do

modelo em três dimensões, podendo ver o seu interior através de cortes, ou visualizando somente as células com as cores seleccionadas pelo usuário. A janela que permite esta visualização é formada basicamente por:

- *Edits* para declarar os limites da visualização (em x , y , e z) e, no caso da visualização por cortes, para declarar os nós onde serão feitos os cortes;
- *CheckBoxs* para escolher as cores das células que vão aparecer (no caso da visualização por cores seleccionadas);
- *UpDowns* para escolher o melhor ângulo de visualização;
- *RadioButtons* para escolher a escala usada (linear ou logarítmica);
- *Image* para pintar a figura.

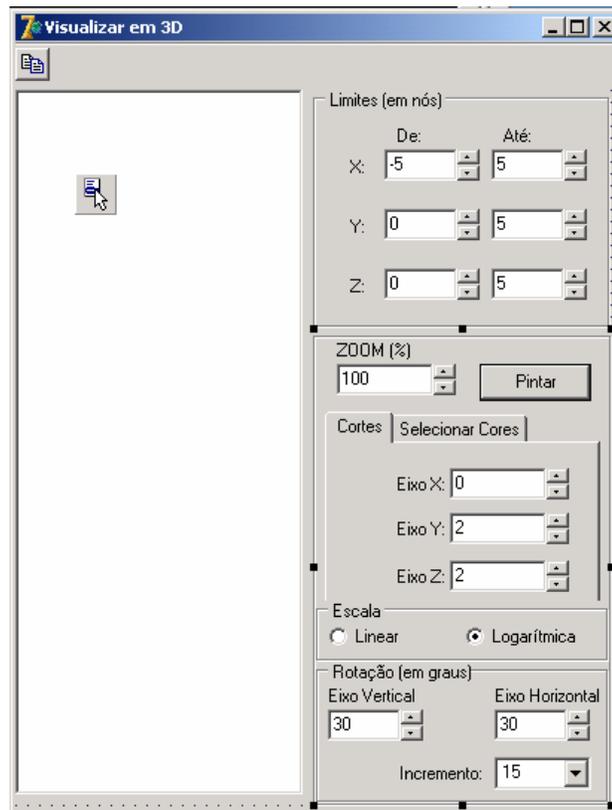


Figura 4.3: Janela usada para visualizar o modelo geofísico tridimensionalmente.

4.2.4 - A estrutura de modificação dos dados

Agora que já foi feito um resumo sobre como os dados de pré-processamento são usados e guardados, e a partir de onde eles são gerados, o leitor já deve ter os requisitos necessários para

compreender como esses dados são manipulados para passar de meros “clicks” para os arquivos de pré-processamento. Este tópico é destinado basicamente a dar dicas a respeito do funcionamento desta interface, principalmente com relação as suas ferramentas. Os assuntos discutidos neste tópico são:

- As opções para modificar as cores das células;
- Os itens *desfazer* e *refazer*;
- O procedimento para pintar o modelo geofísico em três dimensões.

4.2.4.1 - Modificação das cores das células

Na fase de geração do modelo geofísico, duas matrizes estão interligadas por se modificarem mutuamente: *MatrizDasCores* e *MatrizCD*. Elas são modificadas ao mesmo tempo no procedimento *PintaEVerificaNumeroCorpoAoRedor*, o qual tem como entrada as coordenadas do corpo retangular que vai ser inserido no modelo, e o número da cor deste corpo. Este procedimento verifica em todas as células que estão encostadas ao redor do corpo retangular se há alguma célula da mesma cor que ele, e em caso afirmativo, a numeração das células do corpo retangular na *MatrizCD* será a mesma desta célula para que elas sejam consideradas um único corpo, e caso haja mais de um corpo desta mesma cor ao redor deste que está sendo criado, o procedimento guarda os números destes corpos e troca todos pelo número do primeiro corpo encontrado, fazendo com que o conjunto destas células de mesma cor represente um único corpo (um único número) na *MatrizCD*. Caso não haja nenhuma célula da mesma cor ao redor, o procedimento acrescenta uma unidade na variável *ContadorCD* (contador que controla a numeração dos corpos na matriz) e usa este novo número nas células do corpo retangular, caracterizando assim um novo corpo. Após a determinação dos números na *MatrizCD*, o procedimento modifica a numeração das células do corpo retangular na *MatrizDasCores*, trocando os números destas células pelo da cor usada na pintura. Este procedimento é de extrema importância para o funcionamento das funções que modificam as cores das células: *criar corpo retangular*; *mudar a cor de um corpo*; *mudar a cor dos corpos de mesma propriedade*; e *colar região copiada*.

A opção *criar corpo retangular* é usada para modificar as cores de um conjunto de células, limitado por um retângulo definido pelo usuário através do mouse. A modificação das

cores destas células é feita com a ajuda dos eventos *OnMouseDown* (que pode, neste caso, ser substituído pelo evento *OnSelectCell*) e *OnMouseUp* da *DrawGridMalha*, e cujos passos são mostrados a seguir. Em primeiro lugar, são coletados no evento *OnMouseDown* as coordenadas (*CelulaX1*, *CelulaY1*) da célula esquerdo-superior do retângulo formado pelas células que vão ser modificadas. Em seguida, no evento *OnMouseUp*, são coletadas as coordenadas (*CelulaX2*, *CelulaY2*) da célula direito-inferior do mesmo retângulo, e o número da cor usada para a pintura. Finalmente, ainda no evento *OnMouseUp*, é chamado o procedimento *PintaEVerificaNumeroCorpoAoRedor*, o qual modifica a numeração das células.

Para modificar as cores das células de um único corpo (opção *mudar cor de um corpo*) ou modificar as cores de todas as células da mesma cor (opção *mudar cor das células de mesma propriedade*) no modelo, chama-se, no evento *OnMouseUp*, o procedimento *VerificaEPintaParaSbtn2eSbtn3* que procura dentre todas as células, as que pertencem ao mesmo corpo (*MatrizCD*) ou as que são da mesma cor (*MatrizDasCores*), respectivamente. Quando uma célula com a mesma característica (pertencer a um mesmo corpo ou ser da mesma cor, dependendo da opção marcada) da célula clicada é encontrada, o procedimento troca a cor desta célula através do procedimento *PintaEVerificaNumeroCorpoAoRedor*.

A opção *colar região copiada* troca as cores das células de uma determinada região retangular pelas cores das células da região que foi copiada anteriormente. A região copiada fica armazenada na *MatrizCopiaRegiao*, e estes dados são usados para modificar as cores das células do retângulo com o mesmo tamanho desta matriz, e cujo canto esquerdo-superior está na célula clicada. Para trocar as cores destas células, chama-se o procedimento *ColarRegiao*, o qual usa cada cor guardada na matriz para aplicar o procedimento *PintaEVerificaNumeroCorpoAoRedor* na respectiva célula que deve ter a sua cor modificada.

Após a modificação das cores das células com qualquer das quatro ferramentas discutidas acima, chama-se o procedimento *DrawGridMalha.Refresh* [que chama o procedimento *OnPaint*, o qual pinta as células da *DrawGridMalha* que estão aparecendo na tela neste instante (Cantù, 2001)] e o procedimento *VerificaCoresDiferentes2D* (o qual verifica quantas e quais cores estão sendo usadas no modelo 2D, ou na seção do modelo 3D, e disponibiliza um espaço para estas cores na *StringGridRoMi*, onde as propriedades físicas associadas a cada cor são declaradas pelo usuário).

4.2.4.2 – Os itens *Desfazer* e *Refazer*

Para possibilitar desfazer a última alteração feita com o auxílio de uma das ferramentas para modificação das células (*criar corpo retangular; mudar a cor de um corpo; mudar a cor dos corpos de mesma propriedade; colar região copiada.*), guarda-se na *MatrizDesfazerCores* os valores da *MatrizDasCores* que estavam nesta antes de se fazer a última alteração. De forma análoga guarda-se na *MatrizDesfazerCD* os dados da *MatrizCD*. Assim, quando for acionada a opção para desfazer, basta resgatar os dados que estavam armazenados.

Após desfazer uma ação, é possível refaze-la. Para isto chama-se o procedimento *ComandosParaRefazer*, o qual usa as informações coletadas na última alteração [opção usada para mudar as cores das células; célula clicada; cor usada (no caso da opção *colar região copiada*, guarda-se na *MatrizDesfazerColar*, as cores das células que foram coladas);] para colocar de volta a alteração feita pelo usuário. Abaixo temos dois trechos do código fonte. O primeiro é usado para guardar os dados para refazer (a variável *grupo* identifica a seção que está sendo alterada no modelo geofísico 3D). O segundo é usado para recolocar na malha a alteração feita e desfeita pelo usuário (uso da opção *Refazer*). Os caracteres “...” indicam que ali há trechos do código fonte que foram pulados:

- 1º trecho do código fonte

```
// guarda o nº da cor usada na pintura das células
NCor:=TranformaCorParaNumero(Panel2.Color);
// guarda os dados necessários para refazer
Desfazer[grupo-1].SBtn:=Form00_MenuPrincipal.AdicionarCorpo2D;
Desfazer[grupo-1].CoordEsqSup.x:=CelulaX1;
Desfazer[grupo-1].CoordEsqSup.y:=CelulaY1;
Desfazer[grupo-1].CoordDirInf.x:=CelulaX2;
Desfazer[grupo-1].CoordDirInf.y:=CelulaY2;
Desfazer[grupo-1].Cor:=NCor;
```

Trecho do código fonte responsável pelo armazenamento dos dados usados para refazer uma ação. Estes comandos foram retirados do procedimento *TForm4_Malha3D.DrawGridMalhaMouseUp*.

- 2º trecho do código fonte

```

procedure TForm4_Malha3D.ComandosParaRefazer(NoDaAcao: Integer);
begin
  // verifica se as coordenadas da célula estão dentro da malha
  if(Desfazer[grupo-1].CoordEsqSup.x <= DrawGridMalha.ColCount-1)and
    (Desfazer[grupo-1].CoordEsqSup.y <= DrawGridMalha.RowCount-1)then
    begin
      // aciona a função que foi usada para modificar as cores das células
      if Desfazer[grupo-1].SBtn=Form00_MenuPrincipal.AdicionarCorpo2D then
        begin
          with Form00_MenuPrincipal do
            begin
              AdicionarCorpo2D.Checked:=True;           TrocaCorProp2D.Checked:=False;
              TrocaCorCorpo2D.Checked:=False;   ActionCopiaCell.Checked:=False;
              ActionColaCell.Checked:=False;
            end;
          end
        else
          begin
            ...

            end

          // verifica quais células foram clicadas
          CelulaX1:=Desfazer[grupo-1].CoordEsqSup.x;
          CelulaY1:=Desfazer[grupo-1].CoordEsqSup.y;
          if(Desfazer[grupo-1].CoordDirInf.x <= DrawGridMalha.ColCount-1)then
            CelulaX2:=Desfazer[grupo-1].CoordDirInf.x
          else
            CelulaX2:=DrawGridMalha.ColCount-1;
          if(Desfazer[grupo-1].CoordDirInf.y <= DrawGridMalha.RowCount-1)then
            CelulaY2:=Desfazer[grupo-1].CoordDirInf.y
          else
            CelulaY2:=DrawGridMalha.RowCount-1;
          // verifica qual cor foi usada
          NCor:=Desfazer[grupo-1].Cor;
          CelulaX:=CelulaX1;   CelulaY:=CelulaY1;

          // chama o procedimento que realizou a alteração feita nas células
          if (Form00_MenuPrincipal.TrocaCorCorpo2D.Checked)or
            (Form00_MenuPrincipal.TrocaCorProp2D.Checked)then
            VerificaEPintaParaSBtn2ESBt3
          else
            if (Form00_MenuPrincipal.AdicionarCorpo2D.Checked) then
              PintaEVerificaNumeroCorpoAoRedor
            else

```

```

    if (Form00_MenuPrincipal.ActionColaCell.Checked) then
        RefazerColarRegiao(Desfazer[grupo-1].CoordDirInf);
    end;
end;

```

Trecho do código fonte responsável por refazer uma ação que foi desfeita. Neste trecho estão os principais comandos usados no procedimento TForm4_Malha3D.ComandosParaRefazer.

4.2.4.3 – Geração da figura para a visualização em 3D

4.2.4.3.1 – Determinação das coordenadas dos nós

Para desenhar o paralelepípedo referente ao trecho do modelo geofísico que vai ser visualizado em 3D, a primeira medida a ser tomada é determinar as coordenadas dos nós. Para isto, deve-se seguir alguns passos.

Primeiramente, deve-se calcular as coordenadas de quatro pontos fundamentais: o canto do paralelepípedo que ficará mais próximo do observador (vamos chamá-lo de P), e os pontos de fuga (Gomes & Velho, 2003) nas direções x , y , e z (vamos chamá-los de Pfx , Pfy , e Pfz , respectivamente). As coordenadas do ponto P são calculadas por aproximações baseadas nos tamanhos das células que vão aparecer na visualização, enquanto que as coordenadas dos pontos de fuga dependem basicamente do ângulo de visualização, que determina as rotações dos mesmos, e as suas distâncias em relação ao ponto P .

O próximo passo, é calcular as coordenadas dos nós que ficam sobre cada reta determinada pelo ponto P e por um dos pontos de fuga. O processo para a determinação desses pontos será discutido na secção 4.2.4.3.4. A Figura 4.4 ilustra as posições desses nós (os pontos de fuga aparecem mais próximos do que realmente são). Estes nós, que estão localizados em cada um dos seguimentos de reta pretos da Figura 4.4, são guardados em vetores, para posteriormente ajudarem na determinação das coordenadas de todos os nós do trecho do modelo que será visualizado.

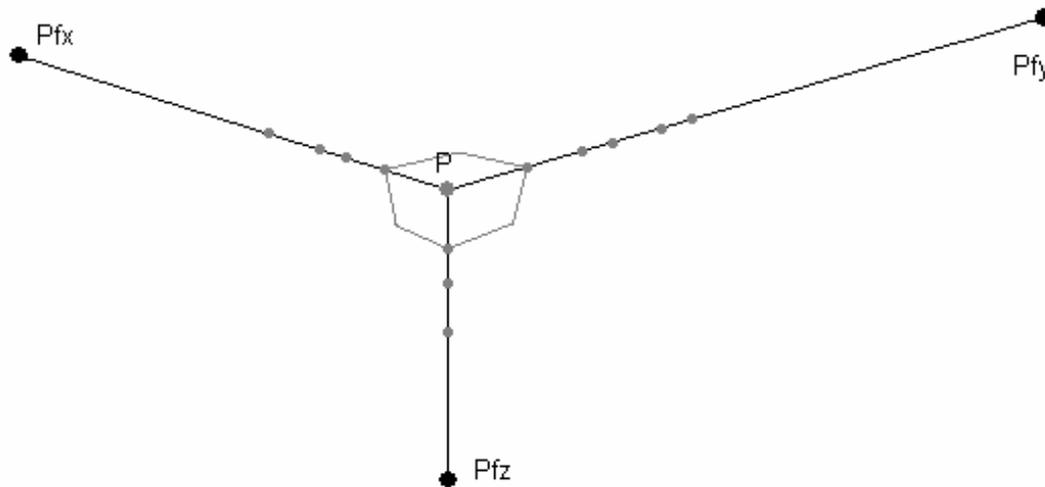


Figura 4.4: Ilustração dos pontos de fuga (pontos pretos) e dos nós que determinam todos os demais nós da figura (pontos cinza).

Finalmente, deve-se determinar as coordenadas de cada nó do trecho do modelo geofísico e guarda-las na matriz *PosDosNos3D*. Cada ponto encontrado corresponde a um nó, e cada um deles é encontrado por meio da resolução de um sistema de duas equações, onde cada equação representa uma reta. O ponto de intersecção dessas duas retas corresponde a um nó do modelo. Cada uma dessas duas retas é determinada por dois pontos (para facilitar a compreensão de como as posições dos nós são encontradas, vamos fixar os pontos de fuga, mas estas coordenadas podem ser determinadas com os pontos de fuga permutados em relação à ordem colocada aqui). A primeira reta é determinada pelo ponto *P fz* e por um dos nós sobre o seguimento de reta que passa por *P* e *P fy*, enquanto que a segunda reta é determinada pelo ponto *P fy* e por um dos nós sobre o seguimento de reta que passa por *P* e *P fz*. Fazendo todas as combinações para determinar estas retas, encontra-se as coordenadas dos nós da face *YZ* (Figura 4.5) do paralelepípedo. Um processo análogo é feito para determinar as coordenadas dos nós na face *XZ*. De posse das posições dos nós nessas duas faces, as coordenadas do restante dos nós é encontrada pela intersecção de duas retas. A primeira delas é determinada pelo ponto *P fx* e um dos pontos da face *YZ*, e a outra é determinada pelo ponto *P fy* e um dos pontos da face *XZ*. Fazendo todas as combinações desse tipo para encontrar as duas retas (e conseqüentemente a coordenada do nó), todos os pontos que representam os nós do trecho do modelo que está sendo visualizado em 3D são determinados (Figura 4.6).

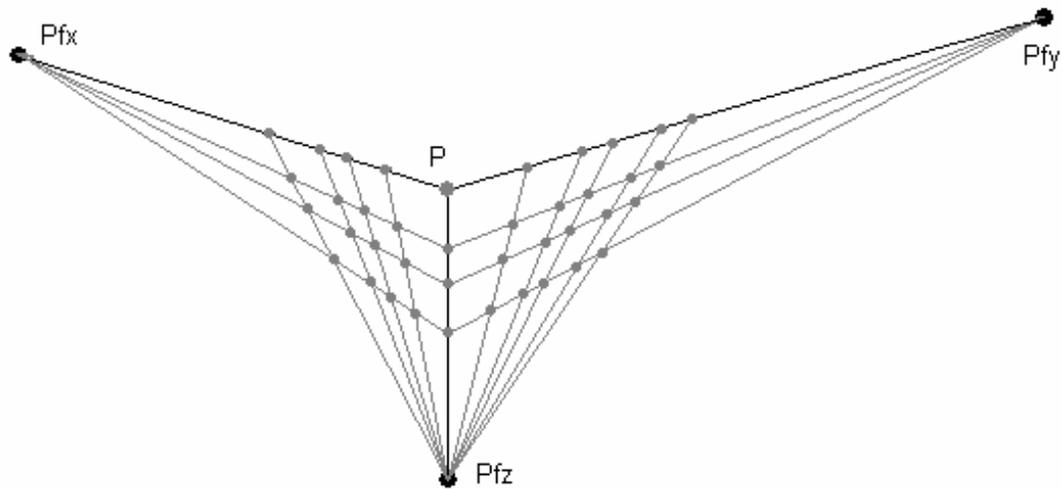


Figura 4.5: Ilustração dos nós nas faces XZ e YZ do paralelepípedo, e das retas usadas para determinar a posição desses nós.

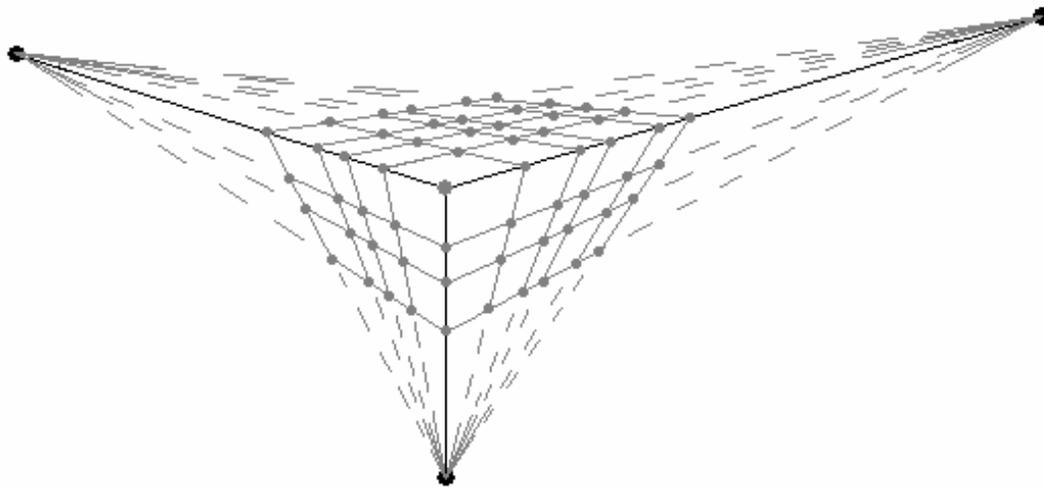


Figura 4.6: Ilustração dos nós do paralelepípedo através dos pontos de intersecção das retas usadas para determina-los. Os nós do interior do paralelepípedo também foram determinados, mas não foram colocados na figura.

4.2.4.3.2 – Visualização por cortes

Para desenhar a figura do modelo geofísico visualizado através de cortes nos eixos, basta pintar as faces visíveis das células que devem aparecer. Para isto, basta fazer um laço para pintar todas as células de uma face externa do modelo (a face YZ, por exemplo) e colocar uma condição

para que, caso uma célula esteja num intervalo onde não deve aparecer (à esquerda do corte em Y e acima do corte em Z, neste caso), sejam pintadas as faces das células internas ao invés das externas, ou seja, as faces paralelas a esta face externa e que passam pelo nó do corte na direção perpendicular a esta face externa (corte na direção X, neste caso). Um processo análogo é feito com as outras duas faces.

As cores usadas para pintar as faces das células são as mesmas usadas para pintar as células da *DrawGridMalha*, ou seja, os números guardados na *MatrizDasCores* transformados em cor pela função *TransformaNumeroParaCor*. Cada face é pintada usando-se os quatro pontos (nós da matriz *PosDosNos3D*) que a determinam e a cor da célula. Esta pintura é feita através do procedimento *Canvas.Polygon* do Delphi (Cantù, 2001).

4.2.4.3.3 – Visualização por cores escolhidas

Agora que as coordenadas de todos os nós já estão armazenadas, pintar somente as células de cores previamente escolhidas é uma tarefa fácil. Para isto, basta fazer um laço para percorrer todas as células na porção do modelo que vai aparecer na figura. Para cada célula percorrida, verificamos se a sua cor é igual a uma das cores escolhidas pelo usuário. Em caso afirmativo, pinta-se as três faces da célula que podem ser vistas, de acordo com o ângulo de visualização. Caso a cor da célula não seja igual a nenhuma das cores escolhidas, não é necessário pintar nada. Só deve-se tomar cuidado para pintar as células de trás antes das células da frente nas três direções, para que a figura tenha sentido.

4.2.4.3.4 – Rotação do paralelepípedo

A rotação feita no paralelepípedo depende de dois ângulos declarados pelo usuário: *TetaEixoV* (rotação com eixo vertical) e *TetaEixoH* (rotação com eixo horizontal). Com estes dois ângulos é possível girar o paralelepípedo, bastando para isso girar os três pontos de fuga, e os nós que ficam sobre cada reta determinada pelo ponto *P* e por um dos pontos de fuga.

Para determinar a posição do ponto de fuga, é necessário calcular a distância que ele deve ficar do ponto *P* e, em seguida, transformar as suas coordenadas de acordo com duas rotações sucessivas: as abscissas que dependem do ângulo *TetaEixoV*, e as ordenadas que dependem dos

ângulos $TetaEixoV$ e $TetaEixoH$. Abaixo temos um trecho do código fonte que é responsável por essa rotação (o ponto P aqui é chamado de “Centro”):

```
// determina a que distância o ponto de fuga deve ficar de Centro
mTg:=(zoom/100)*m*(Tan(Pi/2 - TetaEixoV)+Tan(Pi/2 - TetaEixoH)+
      (Sin(TetaEixoH)+Sin(TetaEixoV)));
if mTg > (zoom/100)*100000. then mTg:=(zoom/100)*100000.;

// determina as coordenadas do ponto de fuga na direção Y
PntDeFugaY.X:=Centro.X+ mTg*Cos(TetaEixoV);
PntDeFugaY.Y:=Centro.Y- mTg*Sin(TetaEixoV);
PntDeFugaY.Y:=Centro.Y-(Centro.Y-PntDeFugaY.Y)*Cos(TetaEixoH);

//*****

// determina a que distância o ponto de fuga deve ficar de Centro
mTg:=(zoom/100)*m*(Tan(TetaEixoV)+Tan(Pi/2 - TetaEixoH)+
      (Sin(TetaEixoH)+Sin(Pi/2 - TetaEixoV)));
if mTg > (zoom/100)*100000. then mTg:=(zoom/100)*100000.;

// determina as coordenadas do ponto de fuga na direção X
PntDeFugaX.X:=Centro.X+ mTg*Cos(TetaEixoV+Pi/2);
PntDeFugaX.Y:=Centro.Y- mTg*Sin(TetaEixoV+Pi/2);
PntDeFugaX.Y:=Centro.Y-(Centro.Y-PntDeFugaX.Y)*Cos(TetaEixoH);

//*****

// determina a que distância o ponto de fuga deve ficar de Centro
mTg:=(zoom/100)*m*(Tan(TetaEixoH)+ (Sin(Pi/2 - TetaEixoH)));
if mTg > (zoom/100)*100000. then mTg:=(zoom/100)*100000.;

// determina as coordenadas do ponto de fuga na direção Z
PntDeFugaZ.Y:=Centro.Y+mTg*Sin(TetaEixoH);
PntDeFugaZ.X:=Centro.X;
```

O próximo passo é determinar as posições dos nós (pontos da cor cinza na Figura 4.4) que ficam sobre as retas determinadas pelo ponto P e pelos pontos de fuga. A distância entre estes pontos (tamanho das células) fica menor a medida que eles ficam mais afastados do observador. Para determinar o quanto estas distâncias devem ser menores, usa-se a razão de semelhança de dois triângulos. O maior deles tem um dos lados ($LadoMaiorTr$) dado pela distância entre o ponto P e o ponto de fuga. O triângulo menor tem o lado correspondente ($LadoMenorTr$) dado pela distância entre o ponto sobre a reta determinada por P e o ponto de fuga, que está imediatamente

mais afastado do observador, e o ponto de fuga. A distância entre os nós é determinada multiplicando-se o valor *razaoTr* (em que $razaoTr = LadoMenorTr / LadoMaiorTr$) pela medida da distância calculada anteriormente sem levar em conta este efeito. Abaixo temos o trecho do código fonte responsável pela determinação dos pontos sobre a reta determinada pelos pontos *P* e *Pfy* (*PtsFrontalEixoY[i]* é o vetor que guarda os pontos sobre a reta, em que o índice *i* do vetor vai aumentando à medida que o ponto se afasta do observador):

```
// #####      EixoY      #####
LadoEixo:=0;      razaoTr:=1;
LadoMaiorTr:=Sqrt(Sqr(PntDeFugaY.x-Centro.x)+
                  Sqr(PntDeFugaY.y-Centro.y));
PtsFrontalEixoY[0].X:=Centro.X;      PtsFrontalEixoY[0].Y:=Centro.Y;

// determinação dos pontos sobre o eixo Y-Abs
for i:=1 to NCelyAbs do
  begin
    LadoEixo:=LadoEixo+LadoCelyAbs[i-1]*razaoTr;
    PtsFrontalEixoY[i].X:=Centro.X+LadoEixo*Cos(TetaEixoV);
    PtsFrontalEixoY[i].Y:=Centro.Y-LadoEixo*Sin(TetaEixoV);
    PtsFrontalEixoY[i].Y:=Centro.Y-Abs(Centro.Y-
                                        PtsFrontalEixoY[i].Y)*Cos(TetaEixoH);
    LadoMenorTr:=Sqrt(Sqr(PntDeFugaY.x-PtsFrontalEixoY[i].X)+
                     Sqr(PntDeFugaY.y-PtsFrontalEixoY[i].Y));
    razaoTr:=LadoMenorTr/LadoMaiorTr;
  end;
```

4.3 – INTERFACE PARA VISUALIZAÇÃO DE GRÁFICOS DE CURVAS

O funcionamento desta interface será mostrado em cinco etapas. Em primeiro lugar, vamos destacar as principais características dos arquivos usados nesta interface. Logo depois, vamos mostrar como os dados usados para fazer os gráficos são armazenados. Em seguida, vamos destacar as principais características das janelas que compõem a interface. Posteriormente, vamos mostrar, de uma maneira geral, o processo que transforma os dados do arquivo em gráficos. Finalmente, vamos mostrar o funcionamento das principais ferramentas da interface.

4.3.1 – Os arquivos usados na interface

Os gráficos de curvas gerados nesta interface são provenientes de dois tipos de arquivos: os de dados observados e os de resposta do modelo (arquivo de saída). Os arquivos de dados observados são os obtidos pelas medidas feitas em campo, enquanto que os arquivos de saída são gerados pela eGs com base no modelo e no método (Dipolo-Dipolo, Magnetotelúrico, VLF, etc) escolhidos pelo usuário. Os arquivos de dados observados são identificados pela extensão “*.OBS”. Os arquivos de saída dependem também dos arquivos de dados observados, pois os pontos sobre o eixo das abscissas (que são a primeira coluna da matriz contida no arquivo) nas duas curvas têm que ser os mesmos, para que a comparação entre essas curvas tenha sentido. A extensão desses arquivos depende do método ao qual eles pertencem. Abaixo temos exemplos da estrutura de dados dos dois tipos de arquivo. O primeiro é de um arquivo de dados observados, enquanto que o segundo é de um arquivo de saída (os arquivos são de métodos diferentes).

- arquivo de *dados observados*

```

VLF2D          ! Chave que indica a qual método geofísico estes dados pertencem
29 17000       ! Número de observações, Frequência
! Matriz que contém os dados para plotar
-300 -0.17351E+00 -0.65984E+01
-250 -0.18681E+00 -0.94631E+01
-200 -0.19139E+00 -0.13320E+02
-150 -0.17845E+00 -0.18280E+02
-100 -0.13666E+00 -0.24081E+02
-90  -0.12382E+00 -0.25226E+02
-80  -0.10929E+00 -0.26314E+02
-70  -0.93018E-01 -0.27301E+02
-60  -0.74909E-01 -0.28122E+02
-50  -0.54880E-01 -0.28676E+02
-40  -0.32870E-01 -0.28781E+02
-30  -0.91329E-02 -0.28021E+02
-20  0.13745E-01 -0.25049E+02
-10  0.16096E-01 -0.12187E+02
 0    0.10880E-06 -0.48006E-04
 10  -0.16095E-01 0.12187E+02
 20  -0.13745E-01 0.25049E+02
 30  0.91328E-02 0.28021E+02
 40  0.32870E-01 0.28781E+02
 50  0.54880E-01 0.28676E+02
 60  0.74909E-01 0.28122E+02
 70  0.93018E-01 0.27301E+02
 80  0.10929E+00 0.26314E+02
 90  0.12382E+00 0.25226E+02
100  0.13666E+00 0.24081E+02
150  0.17845E+00 0.18280E+02
200  0.19139E+00 0.13320E+02
250  0.18681E+00 0.94631E+01
300  0.17351E+00 0.65984E+01

```

- arquivo de *saída*

```
! Matriz que contém os dados para plotar
.0100  36.0  41.5
.0158  37.7  37.5
.0251  43.1  36.5
.0398  54.0  37.0
.0631  63.0  39.5
.1000  68.1  45.0
.1584  69.8  48.4
.2512  62.7  53.5
.3981  53.2  57.9
.6309  43.8  60.7
1.0000  35.4  63.3
1.5850  26.4  61.7
2.5120  21.6  58.3
3.9810  17.4  55.2
6.3100  14.5  48.7
10.0000 14.4  43.3
15.8500 15.3  33.6
25.1200 20.8  28.6
39.8100 28.7  25.1
63.0900 40.9  28.4
100.0000 50.4  32.0
158.5000 59.0  38.6
251.2000 56.8  48.0
398.1000 52.4  54.4
630.9000 45.1  54.7
1000.0000 38.5  60.6
```

4.3.2 – Armazenamento dos dados usados nos gráficos

Para traçar as curvas, esta interface precisa armazenar os dados do arquivo, cujas colunas contêm os dados que serão mostrados nos gráficos, e as informações dadas pelo usuário tais como a quantidade de gráficos, o número de curvas em cada gráfico, quais colunas do arquivo terão seus dados mostrados nos gráficos, a escala que vai ser usada e os intervalos de visualização dos gráficos.

Ao abrir um arquivo, as colunas do mesmo são organizadas em um vetor chamado *ColPPlot* de modo que estas colunas fiquem organizadas uma abaixo da outra, na ordem em que elas estavam no arquivo. Quando se abre um novo arquivo, os dados do arquivo anterior são apagados deste vetor para dar lugar aos dados deste novo arquivo. Assim, para que seja possível

comparar os dados de dois arquivos diferentes (porém ambos devem ter os mesmos valores na primeira coluna) guarda-se, na matriz *ColDoPlotAnterior*, os dados de todos os gráficos que estão na tela (sendo estes dados do mesmo arquivo ou não) e, no momento em que se plota os dados, junta-se estes dados aos do novo arquivo, permitindo assim a comparação entre eles.

Além dos dados do arquivo, é necessário armazenar as informações dadas pelo usuário a respeito dos gráficos. A primeira delas é com relação ao número de gráficos. Esta interface permite fazer um, dois ou quatro gráficos de uma só vez. Esta informação é armazenada na variável *NdePlots*. Deve-se armazenar também o número de curvas que vão aparecer em cada gráfico. Estes dados são guardados no vetor *NcolPlotAcumulativo[d]* (em que *d*, que varia de 1 a 4, indica o gráfico do qual a variável se refere) que indica o total de curvas que vai aparecer no gráfico, enquanto que o vetor *NcolPlot[d]* indica o número de curvas que vão ser acrescentadas neste gráfico. Outra informação sobre o gráfico de curvas é a que indica quais colunas do arquivo vão ser pintadas em cada gráfico. Estas informações ficam armazenadas na matriz *PosDaColuna* de dimensão 4 x 6 (4 é o número máximo de gráficos feitos de uma só vez, e 6 é o número máximo de curvas que podem ser colocadas em cada gráfico). A escala usada no gráfico também é uma informação dada pelo usuário. Esta escala pode ser de três tipos: linear tanto em X quanto em Y (*Linear-Linear*), logarítmica em X e linear em Y (*Log-Linear*), ou logarítmica em X e em Y (*Log-Log*). Esta informação é guardada na variável *TipoDePlot[d]*, em que *d* é o mesmo citado acima. A última informação importante dada pelo usuário é com relação aos intervalos de visualização dos gráficos. Estas informações são o valor inicial do intervalo de visualização (*ValInicialXLinLin[d]* que neste caso indica o valor inicial do eixo X na escala *Linear-Linear*), o número de subintervalos neste intervalo de visualização (*NIntXLogLin[d]* que neste caso indica o número de subintervalos no eixo X na escala *Log-Linear*) e o valor de cada subintervalo (*ValorDoIntervaloYLogLin[d]* que neste caso indica o número de subintervalos no eixo Y na escala *Log-Linear*).

O armazenamento dos dados do arquivo e das escolhas do usuário a respeito do gráfico é essencial para se fazer um gráfico. Outras informações tais como qual figura que vai representar os pontos ou qual grade que vai cobrir os gráficos são de menor importância e será discutido posteriormente com menor detalhe.

4.3.3 – As janelas que compõem a interface

Esta interface é composta por seis janelas. A maioria delas não tem componentes muito diferentes das que foram mostradas até aqui. Portanto, aqui serão mostradas somente três dessas janelas para evitar repetições.

A primeira janela (Figura 4.7) é a que mostra o(s) gráfico(s). Ela é composta por *Panels* e *Images* usados para identificar as curvas, uma *ToolBar*, um *PrintDialog* e um *PopUpMenu*.

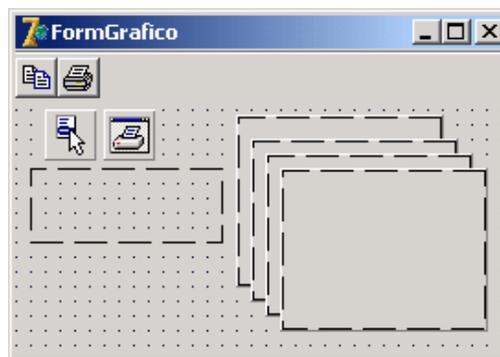


Figura 4.7: Janela usada para mostrar os gráficos.

A próxima janela (Figura 4.8) é usada para declarar quais colunas do arquivo serão usadas para gerar os gráficos, e em que gráfico elas irão aparecer. Ela é composta basicamente por *Edits* que permitem declarar quantas e quais colunas do arquivo serão usadas para gerar os gráficos e saber quais já foram usadas, e *Labels*, sendo que um deles mostra quantas colunas o arquivo tem.



Figura 4.8: Janela usada para declarar quais colunas do arquivo serão usadas para gerar os gráficos.

A última janela mostrada aqui (Figura 4.9) é a que permite declarar os intervalos de visualização dos gráficos. Esta janela é a mesma usada para alterar os limites de visualização dos gráficos nas três escalas usadas nesta interface (*Linear-Linear*, *Log-Linear* e *Log-Log*). Ela é composta basicamente por *Edits* e *UpDowns*, que aparecem ou não de acordo com uma das três funções designadas para a janela.

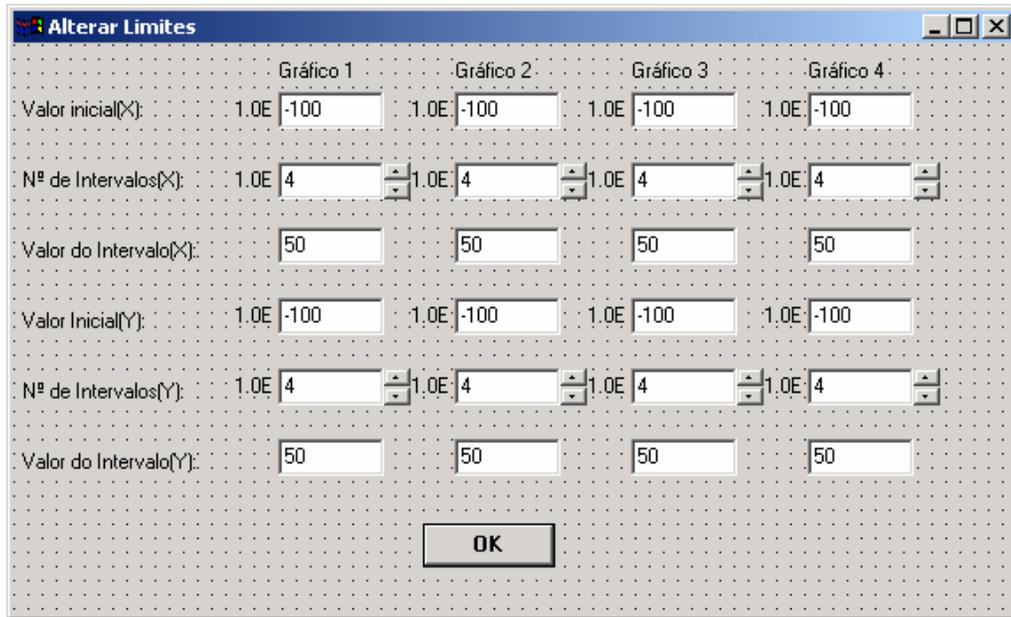


Figura 4.9: Janela usada para declarar os intervalos de visualização dos gráficos nas escalas *Linear-Linear*, *Log-Linear* e *Log-Log*.

4.3.4 – O processo de geração de gráficos

Para gerar os gráficos nesta interface deve-se seguir uma série de passos que estão em função do tipo de grade, da escala, do número de curvas em cada gráfico e do número de gráficos. Os passos básicos para a geração de gráficos são mostrados a seguir.

O usuário pode chamar um gráfico de três formas (de acordo com o tipo de grade que vai cobrir os gráficos): *SemMalha*, *MalhaParcial* ou *MalhaCompleta*. Para efeito de ilustração, vamos analisar a opção *MalhaParcial*, mas as outras duas opções são análogas. Em primeiro lugar, ao acionar a opção *MalhaParcial*, chama-se o procedimento *PlotMalhaParcialExecute*, que identifica através da variável *malha* o tipo de grade que foi escolhido. Se o grupo de curvas mostrado for o primeiro (o primeiro após ter aberto o programa ou, o primeiro após ter limpa-

o(s) gráfico(s) anterior(es) através da opção *LimparPlot*) chama-se o procedimento *TipoMalha*. Caso contrário, chama-se o procedimento *PlotarComAlteracoes*. O procedimento *TipoMalha* limpa os gráficos, verifica se os limites de visualização dos gráficos são automáticos (deve estar marcada a opção *Limites Automáticos*) ou não, pinta as grades em cada gráfico (por meio do procedimento *UmPlotMalha*) e em seguida chama o procedimento *PlotarSemMalha*. Já o procedimento *PlotarComAlteracoes* (que é usado para fazer um gráfico a partir de outro que já está feito) chama o procedimento *LiberaFormEPlota*, o qual chama os procedimentos *LiberaForm* e *TipoMalha*. Nos dois casos chama-se o procedimento *PlotarSemMalha*, o qual chama o procedimento *UmPlotPlotarSemMalha* uma vez para cada gráfico.

O procedimento *UmPlotPlotarSemMalha* chama os procedimentos que calculam, para um determinado gráfico, as coordenadas dos pontos das curvas do gráfico e as posições dos textos que mostram a escala e identificam o gráfico. Em primeiro lugar, o procedimento calcula o valor da variável *NcolPlotAcumulativo[d]*. Em seguida chama o procedimento *LabelsIdentificaEixos*, que coloca as identificações dos eixos dos gráficos. Finalmente, segue uma série de comandos que dependem da escala escolhida (*TipoDePlot[d]*). Vamos descrever o código que faz o gráfico do tipo *Linear-Linear*. Primeiramente, chama-se o procedimento *LabelsXLinLin* e o *LabelsYLinLin*, os quais criam e posicionam os *labels* que mostram as coordenadas do gráfico. Em seguida temos um laço que repete os comandos uma vez para cada linha do arquivo. Dentro deste laço chama-se o procedimento *EscolherColunas*, que procura no vetor *ColPPlot* os dados que devem ser plotados. Por último, e ainda dentro deste laço, é feito um outro laço (cujo contador varia de um até *NcolPlotAcumulativo*) dentro do qual chama-se o procedimento *PintarGrafico*, o qual encontra a melhor posição do *Image* para pintar o ponto da curva de acordo com o tipo de plot, e em seguida, chama o procedimento *TipoDeGrafico*, que literalmente pinta os pontos, de acordo com a figura (círculo, ponto, asterisco ou linha) escolhida pelo usuário. Para os plots em escala *Log-Linear* ou *Log-Log* o processo é totalmente análogo.

4.3.5 – Discussão a respeito das principais opções

Até aqui o funcionamento desta interface foi discutido de uma maneira geral e superficial. Nesta seção serão discutidos trechos do código fonte que merecem ser analisados com mais cuidado. Será analisada aqui a leitura dos arquivos de dados observados, o armazenamento dos

dados (colunas do arquivo) que serão plotados e o funcionamento da opção que calcula os limites (intervalos de visualização) dos gráficos automaticamente.

Ao abrir um arquivo de dados observados (identificados pela extensão “*.OBS”) não há nenhuma indicação a respeito do método geofísico ao qual este arquivo pertence. Como cada um desses métodos tem uma informação diferente a respeito dos seus dados observados [informações que vêm antes da matriz (seção 4.3.1)] não é possível obter corretamente os dados da matriz, a não ser que se tenha alguma indicação sobre isto. Para obter esta indicação, implementamos um método que usa uma *chave* (um código) que identifica o método elétrico ou eletromagnético ao qual o arquivo pertence. Assim é possível identificar quantos dados estão antes da matriz, e ler esta de maneira correta. Abaixo temos o código fonte do procedimento que faz esta identificação:

```

procedure TForm00_MenuPrincipal.LeDadosInicioArquivo(Extensao:String);
var Dados,Dados2:Real;
    i:integer;
begin
// LÊ OS DADOS QUE NÃO INTERESSAM PARA O PLOT, CASO HAJA.
if (extensao='obs')or(extensao='Obs')or(extensao='OBS') then
begin
read(ArquivoAbrirPlot,chaveOBS);

//PARA O CASO DO MTI OU SEV
if (chaveOBS='MT1D') or (chaveOBS='SSEV1D') or (chaveOBS='WSEV1D') or (chaveOBS='DSEV1D') then
readln(ArquivoAbrirPlot,Dados)//nº de obs
// PARA O CASO DO LinhaInfDomFreq EY
else if (chaveOBS='LIDFEY1D') then
begin
read(ArquivoAbrirPlot,Dados,Dados);
for i:= 1 to Trunc(Dados)-1 do read(ArquivoAbrirPlot,Dados2);
readln(ArquivoAbrirPlot,Dados2);
end
//DipoloMagnético EFi, LinhaInfDomFreq HX ou HZ, BobinaCirc EFi
else if (chaveOBS='DMVDFE1D') or (chaveOBS='LIDFHX1D') or (chaveOBS='LIDFHZ1D')
or (chaveOBS='BCDFE1D') then
begin
read(ArquivoAbrirPlot,Dados,Dados);
for i:= 1 to Trunc(Dados) do read(ArquivoAbrirPlot,Dados2);
readln(ArquivoAbrirPlot,Dados2);
end
//PARA O CASO DO EletrojatoLI, BobinaCirc HR ou HZ,DipoloMagnético ou HR ou HZ
else if (chaveOBS='MTEEJLI1D') or (chaveOBS='BCDFHR1D') or (chaveOBS='BCDFHZ1D')
or (chaveOBS='DMVDFHR1D') or (chaveOBS='DMVDFHZ1D') then
begin

```

```

    read(ArquivoAbrirPlot,Dados,Dados);
    for i:= 1 to Trunc(Dados)+1 do read(ArquivoAbrirPlot,Dados2);
    readln(ArquivoAbrirPlot,Dados2);
end
//PARA O CASO DO Eletrojato 1 Gaussiana
else if (chaveOBS='MTEEJ1G1D') then
begin
    read(ArquivoAbrirPlot,Dados,Dados);
    for i:= 1 to Trunc(Dados)+2 do read(ArquivoAbrirPlot,Dados2);
    readln(ArquivoAbrirPlot,Dados2);
end
//PARA O CASO DO Eletrojato 3 Gaussiana
else if (chaveOBS='MTEEJ3G1D') then
begin
    read(ArquivoAbrirPlot,Dados,Dados);
    for i:= 1 to Trunc(Dados)+4 do read(ArquivoAbrirPlot,Dados2);
    readln(ArquivoAbrirPlot,Dados2);
end
// PARA O CASO DO VLF, MagnetoTelurico TE ou TM
else if (chaveOBS='VLF2D') or (chaveOBS='TEMT2D') or (chaveOBS='TMMT2D') then
begin
    read(ArquivoAbrirPlot,Dados); // n° de estações e freq
    readln(ArquivoAbrirPlot,Dados);
end
else ShowMessage('A chave do arquivo não está escrita corretamente.');
```

end;

end;

Quando o usuário chama a janela que permite declarar as colunas do arquivo que vão ser plotadas, ele deve clicar no botão “OK” para confirmar os dados digitados. Neste momento, os dados declarados nos campos de edição (*edits*) da janela são colocados em variáveis locais (*NosDasColunas[,]* é uma delas), e em seguida transferidos para a matriz global *PosDaColuna*. No entanto, as posições onde os dados serão armazenados dependem dos gráficos que já estão feitos. O trecho de código fonte a seguir mostra como esta escolha é feita:

```

//coloca os valores das colunas nas variáveis usadas para plotar
for i:=1 to Form00_MenuPrincipal.NdePlots do
begin
    if Form00_MenuPrincipal.NcolPlotAcumulativo[i]<6 then
    begin
        case i of
            1:Form00_MenuPrincipal.UpDown_Max[1]:=UpDown1.Max;
            2:Form00_MenuPrincipal.UpDown_Max[2]:=UpDown2.Max;
```

```

3:Form00_MenuPrincipal.UpDown_Max[3]:=UpDown3.Max;
4:Form00_MenuPrincipal.UpDown_Max[4]:=UpDown4.Max;
end;

if Form00_MenuPrincipal.TelaLimpa=False then //verifica se já tem plot na tela
begin
  for e:=1 to Form00_MenuPrincipal.NcolPlot[i] do
    {Passa o valor da coluna que está em NosDasColunas para PosDaColuna.
    A diferença deste trecho para o de baixo, é que aqui se considera
    que já existem colunas plotadas, então se coloca
    este valor no espaço que corresponde a uma coluna
    que vai ser plotada além das que já estão. Para saber quantas
    colunas já foram plotadas, usa-se "6-UpDown_Max[i]" em vês de
    NcolPlot[i], pois esta última está com o valor somente do
    novo plot e não do anterior e dele juntos("i" é o contador do
    NdePlots que varia de 1 a 4).}
    begin
      Form00_MenuPrincipal.
      PosdaColuna[i,6-Form00_MenuPrincipal.
      UpDown_Max[i]+e]:=NosDasColunas[i,e];
    end;
  end
else
begin
  for e:=1 to Form00_MenuPrincipal.NcolPlot[i] do
    begin
      Form00_MenuPrincipal.PosdaColuna[i,e]:=
      NosDasColunas[i,e];
    end;
  end;
end;
end;
end;

```

Quando um usuário faz gráficos na eGs 3.0, ele tem a opção de visualiza-los nos intervalos que lhe forem mais convenientes. O usuário pode visualizar o gráfico em sua totalidade ou até mesmo em um pequeno subintervalo, como se fizesse um zoom. Em escala linear, o usuário precisa declarar o limite inferior da visualização (*Valor inicial*), o número de subintervalos (*número de intervalos*) e o tamanho de cada subintervalo (*Valor do intervalo*). Em escala logarítmica são declarados o limite inferior (*Valor inicial*, que é um múltiplo de 10) e o número de subintervalos (*número de décadas*). Para o usuário que não queira se preocupar com o ajuste desses intervalos, esta interface oferece a opção *Limites Automáticos*, a qual calcula os valores descritos acima de forma que as curvas que vão aparecer no gráfico sejam visualizados

por completo. Para calcular estes valores é necessário, em primeiro lugar, obter (no procedimento *AbrirArquivoPlot*) os valores mínimo e máximo de cada coluna da matriz do arquivo [*MinColuna[k]* e *MaxColuna[k]*, em que *k* varia de 1 até o número de colunas do arquivo (*colunas*)], assim como identificar as colunas que têm valores negativos ou nulos (*TemNumNegativo[k]*). Em seguida, já no procedimento *AlterarLimitesAutomaticamente*, o máximo (*Max[d]*) e o mínimo (*Min[d]*) de cada gráfico são calculados para em seguida determinar os limites destes gráficos. Logo depois, verifica-se qual a melhor escala para visualizar o gráfico. Valores negativos ou nulos só podem ser visualizados em escala linear, por isso, se a variável *TemNumNegativo* da coluna que vai ser plotada for verdadeira (tem valor *True*), não se usa escala logarítmica no eixo do gráfico que a representa. Finalmente, se a escala for linear, calcula-se os valores dos subintervalos $((Max - Min) / 4)$ e usa-se um número fixo de subintervalos (quatro para o eixo X e seis para Y). Já se a escala for logarítmica, escolhe-se o limite inferior e o número de subintervalos que permitam visualizar a(s) curva(s) por completo. O procedimento *AlterarLimitesAutomaticamente* é chamado no procedimento *TipoMalha*.

4.4 – INTERFACE PARA VISUALIZAÇÃO DE MAPAS DE CONTORNO

Neste item será discutida a interface para visualização dos dados de resposta do modelo e de campo através de mapas de contorno. Para mostrar o funcionamento desta interface, este item será subdividido em três partes. Os temas a serem discutidos são:

- A janela usada na interface;
- O armazenamento dos dados;
- O funcionamento da interface.

4.4.1 – Janela usada para visualização de mapas de contorno

A janela que permite que usuário visualize a resposta de um modelo ou os dados de campo por mapas de contorno de isovalores está ilustrada a Figura 4.10. Ela é formada basicamente por:

- Uma *ToolBar* onde estão localizados os botões para aumentar ou diminuir a figura (zoom), copiar a figura e optar pela escala linear ou logarítmica;

- Um *Image* onde a figura é pintada.

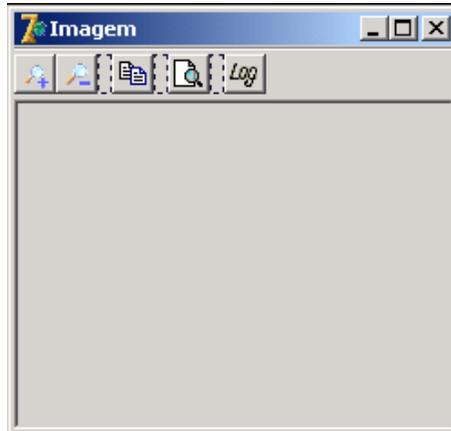


Figura 4.10: Janela usada para exibir mapas de contorno.

4.4.2 – O armazenamento dos dados exibidos

Os dados mostrados nesta interface através de mapas de contorno são armazenados numa matriz chamada *MatrizImagem*. Estes dados são obtidos [ao acionar o evento *OnCreate* (Cantù, 2001)] a partir da interpolação dos dados de uma outra matriz, a qual contém os dados originais. A matriz que contém os dados originais pode armazenar tanto os dados obtidos diretamente de um arquivo quanto uma transformação desses dados (os arquivos de resposta de modelos do método MCSEM, por exemplo, contém as partes real e imaginária de valores complexos, e os valores que são armazenados nessa matriz são os valores absolutos desses números complexos, ou seja, números reais). A escolha da matriz que contém estes dados originais é feita em função do método elétrico ou eletromagnético de onde os dados serão usados para pintar a figura, e é feita de acordo com o valor da variável *MetodoImagem* (variável definida na Unit *UnitMenuPrincipal*), a qual recebe o nome do método elétrico ou eletromagnético que vai fazer uso da interface para apresentar os respectivos dados de saída. Para cada método que faz uso da interface há um trecho de código fonte responsável pelo armazenamento dos dados da matriz que contém os originais na *MatrizImagem*.

4.4.3 – O funcionamento da interface

Diferentemente da interface usada para mostrar gráficos de curvas, esta interface permite criar inúmeras janelas para visualizar os dados dos arquivos de saída. Cada janela criada pode exibir uma matriz diferente, já que esta é definida no momento da criação da nova janela. Antes de criar a janela, é necessário definir o valor da variável *MetodoImagem*. Após criar a janela, é necessário chamar o procedimento *PintaFiguraInterpolada* para pintar a figura.

O procedimento *PintaFiguraInterpolada* identifica o valor máximo (*maxMat*) e o mínimo (*minMat*) da matriz *MatrizImagem*, para depois associar estes valores às cores vermelha e azul, respectivamente. Esta associação é feita via regra de três. A seguir temos as etapas do processo de determinação da cor associada ao valor da matriz. Primeiramente, deve-se associar as *n* cores usadas na pintura aos *n* primeiros números naturais (0 a *n* - 1). Esta associação é feita na função *Cores(ordem):TColor*, que tem como entrada um número inteiro entre 0 e *n* - 1 (*ordem*) e saída uma cor. Esta interface usa 50 cores (*n* = 50), mas pode-se acrescentar ou retirar cores. A seguir, usa-se uma regra de três para associar qualquer número entre *minMat* e *maxMat*, a um número compreendido entre 0 e *n* - 1. Finalmente, é só arredondar este número encontrado e chamar a função *Cores*, que retornará a respectiva cor. A verificação da cor usada para pintar cada célula que compõe a figura é feita dentro de um laço usado para pintar cada célula. Abaixo temos o trecho do procedimento *PintaFiguraInterpolada* responsável pela pintura da figura:

```
ImageImagem.Canvas.Pen.Color:=CorFundo;
ImageImagem.Canvas.Brush.Color:=CorFundo;
ImageImagem.Canvas.Rectangle(0, 0, ImageImagem.Width, ImageImagem.Height);
for i:=0 to LinInterpol-1 do
  begin
    for j:=0 to ColInterpol-1 do
      begin
        if maxMat=minMat then //corrige para não dar divisão por zero
          ordem:=0
        else
          if EscalaLog then
            begin
              ordem:=Round(
                (NCores*(Log10(MatrizImagem[i,j])-Log10(minMat)))/
                (Log10(maxMat)-Log10(minMat)));
            end
          else
```

```

begin
  ordem:=Round(
    (NCores*(MatrizImagem[i,j]-minMat))/(maxMat-minMat));
end;

{ o valor de "ordem" foi obtido com uma regra de três:

(maxMat - minMat) ----- NCores (NCores é o número de cores - 1)
(valor - minMat) ----- ordem
}

if lado>1 then
begin
  ImageImagem.Canvas.Pen.Color:=Cores(ordem);
  ImageImagem.Canvas.Brush.Color:=Cores(ordem);
  ImageImagem.Canvas.Rectangle(lado*j+20, lado*i+20,
                                lado*(j+1)+20, lado*(i+1)+20);
end
else //aqui se usa "pixels" e não "rectangle"
begin
  ImageImagem.Canvas.Pixels[j+20,i+20]:=Cores(ordem);
end;
end;
end;

```

4.5 – INTERFACE PARA VISUALIZAÇÃO DE PSEUDO-SEÇÕES

Neste item será discutida a interface que permite visualizar os dados observados e de saída na forma de pseudo-seções, como é freqüentemente usado no método da polarização induzida (IP). Este método geofísico tem seus dados de saída fora dos padrões em relação aos demais métodos geofísicos usados na eGs, por isso a interface discutida no item 4.4 não poder ser usada para visualizar estes dados. Portanto, foi necessário construir uma interface para a entrada e análise destes dados em particular.

Esta interface é composta de três partes. A primeira, que não foi construída por mim, é a que permite a entrada e o armazenamento dos dados. Os detalhes sobre ela podem ser obtidos em Pereira (2005). A segunda e a terceira são as que melhor permitem visualizar e interpretar estes dados. Estas foram construídas por mim e, portanto, serão discutidas neste Capítulo.

Como nos itens anteriores este também será subdividido. Os temas dos subitens são:

- Arquivos usados na interface;

- A janela usada;
- O armazenamento dos dados;
- O funcionamento da interface.

4.5.1 – Os arquivos de dados do tipo IP

Os arquivos de saída de modelos do método IP são os que mais estão fora dos padrões dos arquivos usados na eGs. Abaixo temos um exemplo desse tipo de arquivo. Perceba a diferença entre este tipo de arquivo e os apresentados no item 4.3.1 e posteriormente no item 4.6.1.

```

13 5 !n° de eletrodos, n° de níveis
! resistividades aparentes
1007.0 1021.0 946.0 636.0 287.0 287.0 640.0 957.0 1021.0 1006.0
1098.0 919.0 424.0 101.0 58.0 102.0 440.0 943.0 1097.0
896.0 332.0 66.0 63.0 62.0 68.0 353.0 927.0
290.0 56.0 92.0 138.0 88.0 59.0 313.0
54.0 114.0 218.0 214.0 107.0 55.0

```

4.5.2 – A janela usada na interface

Nesta interface, uma única janela é usada tanto para editar os dados, quanto para visualiza-los (Figura 4.11). Para separar estas duas funções, usou-se a componente *PageControl*, a qual permite a utilização de um mesmo espaço por diversas páginas, uma de cada vez (Cantù, 2001). A janela é composta por:

- *Edits* para declarar e visualizar o número de níveis e eletrodos;
- *RadioButtons* e *CheckBoxes* para escolher o tipo de dado que será visualizado e o grau de suavização (interpolação) usada;
- Um *PageControl* que se subdivide em três *TabSheets* (*TabSheet1* comporta os *edits* que permitem editar os dados; *TabSheet2* contém a *DrawGrid* onde são pintados os quadrados com cores correspondentes aos dados dos *edits*; *TabSheet3* tem apenas um *ScrollBar* que comporta um *Image* que é criado dinamicamente.).

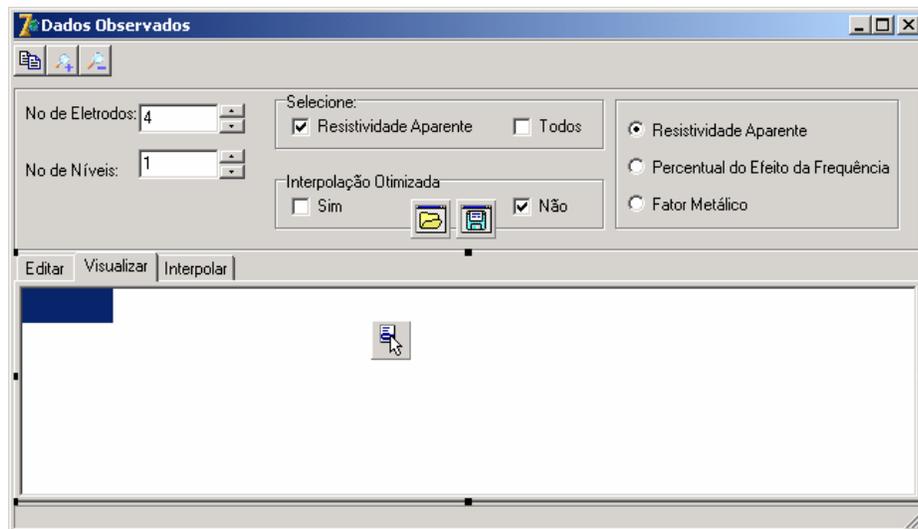


Figura 4.11: Janela usada no método IP.

4.5.3 – O armazenamento dos dados

A interface para os dados do tipo IP precisa armazenar dados de dois tipos: numéricos e cores.

Dentre os dados numéricos que precisam ser armazenados podemos destacar, em primeiro lugar, os dados do arquivo. Estes dados são o número de eletrodos (*NEletrodos*), o número de níveis (*Niveis*), o número de matrizes (*NMalhas*) e as próprias matrizes que podem ser a resistividade aparente, o PFE e o fator metálico, ou somente a resistividade aparente. Estas matrizes ficam armazenadas em *ResAp_PFE_MF[malha]*, em que *malha* varia de 0 a 2 no caso em que existem as três medidas, e vale 0 no caso em que só se tem a medida da resistividade aparente. Além disso, é preciso armazenar os valores que estão no intervalo dos que aparecem na matriz, e que correspondem às cores usadas no espectro de cores. Estes dados ficam armazenados no vetor *ValorEspectro[malha]*, e cujos valores variam do menor ao maior valor da respectiva matriz, a qual é armazenada em *ResAp_PFE_MF[malha]*, e com intervalos fixos entre os valores consecutivos.

Os dados do tipo cor ficam armazenados em um único vetor chamado *Espectro[malha]*, cujos valores vão do vermelho ao azul, passando pelo alaranjado, amarelo e verde. Estas cores têm uma correspondência biunívoca com os respectivos valores do vetor *ValorEspectro[malha]*.

4.5.4 – O funcionamento da interface

4.5.4.1 – Os *Registros* criados para a interface

Em object pascal um registro (*record*) é um tipo composto de variável que pode ser definido pelo programador (Cantù, 2001). As variáveis *Espectro[]*, *ValorEspectro[]* e outras usadas na interpolação dos dados são de tipos definidos na eGs 3.0 através de registros: *TValorDist*, *TEspectro* e *TLinhasIP*. Estas variáveis podem ser consideradas como vetores de outros vetores. O trecho de código abaixo mostra como elas são definidas (os caracteres “...” indicam que ali há trechos do código fonte que foram pulados):

```
Type
  //definição dos registros
  TValorDist=record //Valores distintos
    Valor:array of real;
  end;

  TEspectro=record
    Cor:array of TColor;
  end;

  TLinhasIP=record
    Coluna:array of real;
  end;

  ...

  //declaração das variáveis (vetores) dos tipos definidos acima
  ValorEspectro:Array[0..2] of TValorDist;
  Espectro:Array[0..2] of TEspectro;

  ...

  //define que tamanho terá o vetor Espectro[malha].Cor
  SetLength(Espectro[malha].Cor,4*n[malha]+1);
  //define que tamanho terá o vetor ValorEspectro[malha].Valor
  SetLength(ValorEspectro[malha].Valor,4*n[malha]+1);
  //atribui valores às componentes do vetor Espectro[malha].Cor
  Espectro[malha].Cor[0]:=rgb(0,0,255);
  for i:=1 to n[malha] do
    Espectro[malha].Cor[i]:=rgb(0,incNCor*i,255);
```

4.5.4.2 – Os passos para pintar a pseudo-seção (sem interpolação)

A descrição apresentada aqui é usada para pintar a pseudo-seção sem interpolação apresentada na figura 3.17(b). Primeiramente deve-se chamar o procedimento *OrdenarCores*, o qual calcula os vetores *Espectro[malha]* e *ValorEspectro[malha]*. Finalmente chama-se o procedimento *PintarMalha* para pintar a figura da pseudo-seção na *DrawGrid*.

O procedimento *PintarMalha* é quem de fato pinta a figura da pseudo-seção e, portanto, é chamado em todos eventos que modificam esta figura. A organização deste procedimento é apresentada a seguir. Em primeiro lugar, pinta-se as retas e os textos que identificam os eletrodos e os níveis, de acordo com o tamanho do zoom e da posição da figura definida na barra de rolagem. A seguir, defini-se qual cor vai ser usada para pintar cada quadrado. Para isso verifica-se qual é o valor do vetor *VetorEspectro[malha]* que mais se aproxima do de *ResAp_PFE_MF[malha]* (onde estão guardados os valores do arquivo), e toma-se a cor correspondente no vetor *Espectro[malha]*. Finalmente, pinta-se o quadrado com a cor selecionada e coloca-se o texto em cima do mesmo.

4.5.4.3 – Os passos para obter a pseudo-seção suavizada (interpolada)

No processo de pintura da pseudo-seção analisado no item 4.5.4.2, foram usados somente os dados mostrados nas células de edição (*edits*), e que eram obtidos através da leitura de um arquivo ou do preenchimento manual dos mesmos. Já a pseudo-seção suavizada (Figura 3.18) usa os dados desta pseudo-seção, e outros obtidos através de interpolação linear. O método usado para criar estes dados interpolados é descrito a seguir.

Todo o processo de interpolação dos dados e pintura da figura é feito em um único procedimento chamado *Interpolar*. Neste procedimento calcula-se, em primeiro lugar, o valor da variável *LadoQuadInterp*, que guarda o comprimento dos lados dos quadrados (em pixels) que melhor ajusta o tamanho da figura ao tamanho do formulário (janela). A seguir, transfere-se os dados de *ResAp_PFE_MF* para o vetor *LinhasInterpola*. Este vetor é do tipo *TlinhasIP* definido em 4.5.4.1. Em seguida, tem início o laço que calcula e guarda os valores interpolados. Este laço dobra o número de linhas e colunas do vetor *LinhasInterpola* e diminui pela metade o tamanho dos lados dos quadrados (*LadoQuadInterp*) a cada iteração. O laço termina quando o vetor

LadoQuadInterp fica igual ao de *MaxLadoQInterpol*, o qual assume o valor 2 ou 8 (em pixels) de acordo com o refinamento escolhido pelo usuário para a interpolação (interpolação otimizada). A redefinição das dimensões de *LinhasInterpola* é feita da seguinte maneira:

```
NLinhasInterp:= 2*NLinhasInterp;
NColunasInterp:= 2*NColunasInterp;
SetLength(LinhasInterpola, NLinhasInterp);
For i:=0 to NlinhasInterp - 1 do
    SetLength(LinhasInterpola[i].Coluna, NColunasInterp - i);
```

Cada iteração do laço calcula os dados interpolados de acordo com a ilustração da figura 4.12.

Nesta figura estão representados os seguintes valores:

```
2  6  10  8  6
5  9  10  8
8  9  10
```

Estes valores são organizados em *LinhasInterpola* respectivamente da seguinte maneira (aqui a palavra “*LinhasInterpola*” vai ser substituída por “*LI*”, e a palavra “*Coluna*” vai ser substituída por “*C*”):

```
LI[0].C[0]   LI[0].C[1]   LI[0].C[2]   LI[0].C[3]   LI[0].C[4]
LI[1].C[0]   LI[1].C[1]   LI[1].C[2]   LI[1].C[3]
LI[2].C[0]   LI[2].C[1]   LI[2].C[2]
```

Finalmente, após a determinação de todos os dados interpolados, repete-se os passos usados na obtenção da figura sem interpolação (item 4.5.4.2) com comandos análogos aos dos procedimentos *OrdenarCores* e *PintarMalha*.

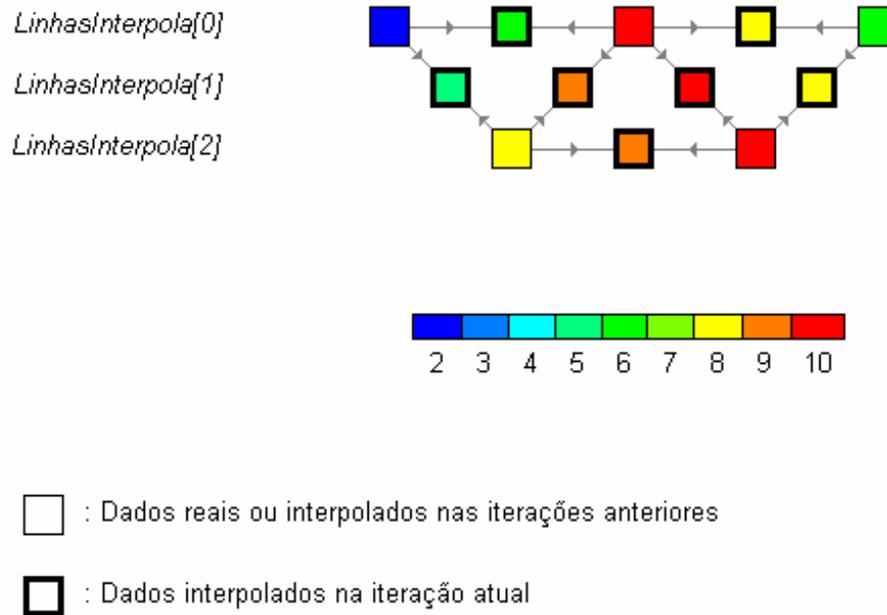


Figura 4.12 : Ilustração do processo de interpolação linear dos dados do tipo IP. Os valores interpolados são obtidos pela média aritmética dos dois valores mais próximos. As setas que apontam para os valores interpolados indicam de onde vêm os dados usados para encontrar estes valores. Os vetores *LinhasInterpola[]* estão indicados ao lado do conjunto de valores que ficam guardados nos mesmos.

4.6 – INTERFACE PARA GERENCIAMENTO DE ARQUIVOS DE SAÍDA DO MÉTODO MCSEM

Para finalizar este Capítulo, será discutida a interface que permite gerenciar arquivos de saída do método MCSEM. Os dados observados do método MCSEM em termos das componentes do campo elétrico são, em geral, normalizados pelas componentes do campo elétrico de um meio sabidamente sem reservatório mas que tenha a mesma seqüência estratigráfica. As informações geoeletricas desse meio de referência são obtidas através de perfis de poço ou por meio do método MMT (Método Magnetotelúrico Marinho). Assim, podemos apresentar os dados de saída de três maneiras: a componente do campo elétrico do modelo com hidrocarbonetos, a componente do campo elétrico do modelo sem hidrocarbonetos, ou a razão

entre as componentes do campo elétrico desses dois modelos. Portanto foi necessário construir uma interface em particular que permita manipular os arquivos que contêm esses dados, que dê as opções de escolha das pastas onde estes arquivos estão localizados e do modo como os dados desses arquivos serão apresentados. A seguir temos os assuntos abordados neste item do Capítulo:

- Arquivos usados na interface;
- A janela usada;
- O armazenamento dos dados;
- O funcionamento da interface.

4.6.1 – Os arquivos de resposta de modelos (arquivos de saída)

A resposta de modelos do método MCSEM é armazenada em seis arquivos. Três desses arquivos contêm as componentes (X, Y e Z) do campo elétrico gerado na ausência de hidrocarbonetos na subsuperfície, enquanto que os outros três arquivos são gerados na presença de hidrocarbonetos. Estes arquivos são armazenados em pastas cujo nome é escolhido pelo usuário. A eGs 3.0 salva nessas pastas três arquivos com o mesmo nome do diretório que à contém, e com as extensões “.Ex”, “.Ey” e “.Ez”. Abaixo temos as seis primeiras colunas (de um total de quarenta) de um arquivo desse tipo (as colunas representam, aos pares, as componentes real e imaginária de um valor complexo):

0.13087E+00	-0.20180E+00	-0.13411E+00	-0.11765E-01	-0.61448E-01	0.24067E-01
0.10089E+00	-0.98360E-01	-0.21599E-01	-0.18459E-01	-0.24928E-01	0.11143E-01
0.24209E-01	-0.40351E-01	-0.22655E-02	-0.12380E-01	-0.73437E-02	0.34808E-02
0.40370E-02	-0.16356E-01	-0.82517E-03	-0.66220E-02	-0.19123E-02	0.44607E-03
-0.49505E-03	-0.67898E-02	-0.96062E-03	-0.33524E-02	-0.65894E-03	-0.44689E-03
-0.12373E-02	-0.28844E-02	-0.96908E-03	-0.16667E-02	-0.48187E-03	-0.53102E-03
-0.11208E-02	-0.12053E-02	-0.84603E-03	-0.78467E-03	-0.47436E-03	-0.36719E-03
-0.85110E-03	-0.44929E-03	-0.67024E-03	-0.31561E-03	-0.43418E-03	-0.17988E-03
-0.59810E-03	-0.10828E-03	-0.49311E-03	-0.75306E-04	-0.35464E-03	-0.44492E-04
-0.40049E-03	0.33710E-04	-0.34281E-03	0.34201E-04	-0.26507E-03	0.30119E-04
-0.26026E-03	0.79532E-04	-0.22947E-03	0.71872E-04	-0.18692E-03	0.59407E-04
-0.16758E-03	0.81921E-04	-0.15133E-03	0.74090E-04	-0.12831E-03	0.62271E-04
-0.10979E-03	0.67984E-04	-0.10116E-03	0.62023E-04	-0.88613E-04	0.53137E-04
-0.75419E-04	0.50757E-04	-0.70683E-04	0.46727E-04	-0.63629E-04	0.40708E-04
-0.55645E-04	0.35648E-04	-0.52881E-04	0.33089E-04	-0.48660E-04	0.29243E-04
-0.44399E-04	0.24208E-04	-0.42633E-04	0.22640E-04	-0.39877E-04	0.20265E-04
-0.37834E-04	0.16280E-04	-0.36588E-04	0.15336E-04	-0.34612E-04	0.13891E-04
-0.33685E-04	0.11119E-04	-0.32730E-04	0.10549E-04	-0.31198E-04	0.96676E-05
-0.30713E-04	0.79119E-05	-0.29937E-04	0.75599E-05	-0.28687E-04	0.70100E-05
-0.28289E-04	0.59851E-05	-0.27639E-04	0.57578E-05	-0.26589E-04	0.53990E-05

4.6.2 – A estrutura da janela usada

A janela na qual o usuário pode gerenciar tanto os arquivos que contêm os dados de resposta de modelos do método MCSEM quanto o modo de apresentação desses dados (Figura 4.13) é formada basicamente por:

- Uma *ToolBar* onde estão localizados os botões para abrir os arquivos e mostrar os dados na interface para visualização de mapas de contorno;
- Dois *DirectoryListBoxs* que permitem escolher os diretórios onde estão localizados os arquivos com os dados;
- Dois *FileListBoxs* que permitem visualizar os arquivos dos diretórios selecionados;
- Dois *ComboBox*, que permitem escolher o drive onde será feita a busca pelos arquivos.

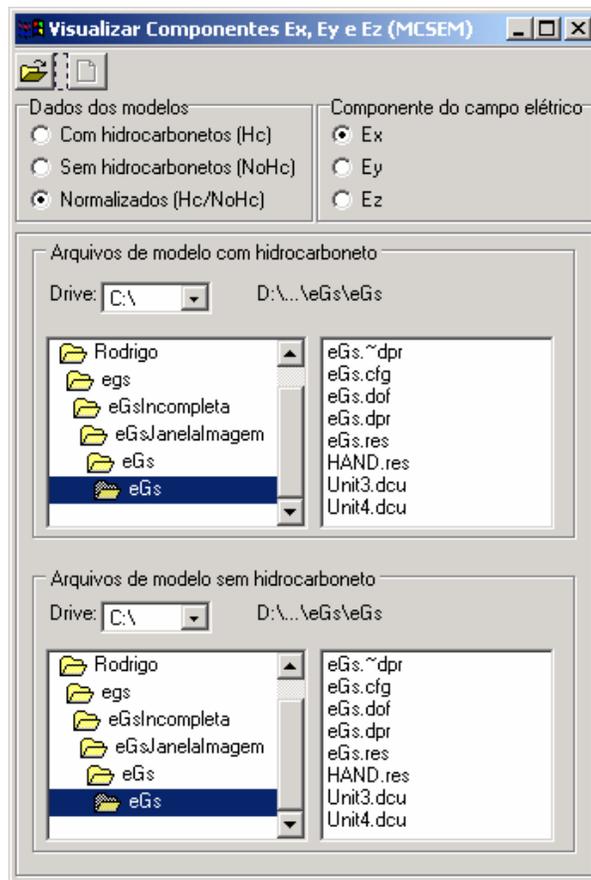


Figura 4.13: Janela usada para gerenciar os arquivos de resposta de modelos do método MCSEM.

4.6.3 – Armazenamento dos dados

Os dados contidos nos arquivos correspondem a apenas um quarto da matriz usada para armazená-los (quadrante direito-inferior), por isso, após a leitura dessa parcela dos dados, os mesmos devem ser “refletidos” na matriz a fim de preenchê-la por completo. Isto pode ser feito graças à simetria desses dados.

Esta interface armazena em matrizes três tipos de dados dos arquivos selecionados. O primeiro deles corresponde ao campo elétrico gerado por um modelo na presença de hidrocarbonetos no subsolo marinho. As componentes (X, Y e Z) deste campo são armazenadas nas matrizes $MatrizMCSEM[0].MatrizExEyEz[i, j, k]$. O primeiro índice (0 (zero) neste caso) corresponde ao modo de apresentação dos dados [campo elétrico do modelo com hidrocarbonetos (índice 0); campo elétrico do modelo sem hidrocarbonetos (índice 1); razão entre estes dois campos elétricos (índice 2)]. Os índices “i” e “j” correspondem às coordenadas dos valores do

campo elétrico na matriz enquanto que o índice “ k ” corresponde à componente do campo elétrico (os valores de k iguais a 0, 1 e 2 correspondem, respectivamente, às componentes X, Y e Z do campo elétrico). Os outros dois tipos de dados correspondem aos índices 1 e 2 de *MatrizMCSEM*, como já foi citado. É oportuno ressaltar que *MatrizMCSEM[p].MatrizExEyEz[i, j, k]* contém nove matrizes, pois p e k assumem os valores 0, 1 ou 2. Estas matrizes recebem seus valores no procedimento *TFormVisMCSEM.ToolBtnAbrirMCSEMClick*.

4.6.4 – O funcionamento da interface

Antes de qualquer coisa, o usuário deve escolher os dois diretórios que contêm os arquivos que serão usados (um diretório contém os arquivos de saída gerados a partir de um modelo *com* hidrocarbonetos e o outro a partir de um modelo *sem* hidrocarbonetos). Cada diretório deve ter o mesmo nome dos arquivos contidos nele, com a ressalva de que os arquivos devem ter as extensões “.Ex”, “.Ey” e “.Ez”. No procedimento que abre os arquivos e guarda os dados (*TFormVisMCSEM.ToolBtnAbrirMCSEMClick*), é feita uma verificação para saber se os arquivos existem nas condições descritas acima. O trecho de código fonte abaixo mostra como isto é feito. A função *FileExists* do Delphi verifica se o arquivo existe ou não.

```

Diretorio1:=(DirectoryListBox1.Items[DirectoryListBox1.ItemIndex]);{mostra
o nome do último diretório (que tem o mesmo nome dos arquivos contidos nele,
mas com extensões .Ex, .Ey e .Ez)}
Diretorio2:=(DirectoryListBox2.Items[DirectoryListBox2.ItemIndex]);

//verifica se os arquivos existem
FaltaArquivo:=False;

Caminho1:=DirectoryListBox1.Directory+'\'+Diretorio1+'.Ex';
if not(FileExists(Caminho1)) then FaltaArquivo:=True;
Caminho1:=DirectoryListBox2.Directory+'\'+Diretorio2+'.Ex';
if not(FileExists(Caminho1)) then FaltaArquivo:=True;

Caminho1:=DirectoryListBox1.Directory+'\'+Diretorio1+'.Ey';
if not(FileExists(Caminho1)) then FaltaArquivo:=True;
Caminho1:=DirectoryListBox2.Directory+'\'+Diretorio2+'.Ey';
if not(FileExists(Caminho1)) then FaltaArquivo:=True;

Caminho1:=DirectoryListBox1.Directory+'\'+Diretorio1+'.Ez';
if not(FileExists(Caminho1)) then FaltaArquivo:=True;

```

```
Caminho1:=DirectoryListBox2.Directory+'\'+Diretorio2+'.Ez';
if not(FileExists(Caminho1)) then FaltaArquivo:=True;
```

Após selecionar os diretórios que contém os arquivos que serão usados, o usuário deve clicar no botão *abrir* para que os dados dos arquivos sejam armazenados nas respectivas matrizes. Para visualizar uma das componentes do campo elétrico (E_x , E_y ou E_z) em um dos modos de apresentação (Hc , $NoHc$ ou $Hc/NoHc$) o usuário deve selecionar a opção desejada. Estas escolhas ficam armazenadas, respectivamente, nas variáveis *Nmatriz* e *HcNoHcRazao*, que podem receber os valores 0, 1 ou 2. A visualização desses dados deve ser feita na interface para visualização por mapas de contorno. Abaixo temos o código fonte do procedimento acionado quando o usuário clica no botão que chama uma nova figura:

```
procedure TFormVisMCSEM.ToolBtnFiguraMCSEMClick(Sender: TObject);
var
FormularioImagem: TFormImagem;
texto1, texto2:string[10];
begin

HcNoHcRazao:=RdGrpArquivos.ItemIndex;
NMatriz:=RdGrpCampoEletrico.ItemIndex;
Form00_MenuPrincipal.MetodoImagem:='MCSEM';
Application.CreateForm(TFormImagem, FormularioImagem);

case HcNoHcRazao of
0:texto1:='Hc';
1:texto1:='NoHc';
2:texto1:='Hc/NoHc';
else
texto1:='';
end;

case NMatriz of
0:texto2:='Ex';
1:texto2:='Ey';
2:texto2:='Ez';
else
texto2:='';
end;

FormularioImagem.Caption:='Imagem'+ ' - '+texto1+ ' - '+texto2;
FormularioImagem.Show;
FormularioImagem.PintaFiguraInterpolada(clBtnFace);
```

end;

4.7 – SUGESTÕES PARA IMPLEMENTAÇÃO

Dentre as características das interfaces da eGs 3.0 mostradas neste Capítulo, algumas poderiam ser modificadas a fim de alcançar um melhor desempenho com relação à facilidade de uso. Outras ferramentas sequer foram implementadas. O principal motivo disso foi a falta de tempo durante o desenvolvimento do programa aliado ao atraso no aparecimento dessas idéias. Na tentativa de remediar parcialmente este mal, ficam aqui sugestões para futuras alterações no programa.

4.7.1 – Personalização das cores

A interface para criação de modelos dispõe de 40 cores fixas para a identificação dos materiais que compõem o modelo. Foram selecionadas cores que vão do azul ao vermelho, passando pelo verde, pelo amarelo e pelo alaranjado.

Para que o usuário tenha mais liberdade para escolher as cores que quer usar nos modelos, basta deixar reservados nas funções *TransformaNumeroParaCor* e *TransformaCorParaNumero* espaços para serem preenchidos com as cores escolhidas pelo usuário. A escolha da cor pode ser feita com o auxílio da componente *ColorDialog* do Delphi (Cantù, 2001).

4.7.2 – Visualização dos resultados de inversão em 3D

Uma das grandes vantagens de se possuir uma interface para visualizar modelos 3D é que ela também pode ser ajustada para visualização dos resultados de inversão em 3D.

Na versão atual da eGs, quando se usa a ferramenta *visualizar em 3D*, os dados do modelo 3D que são pintados na figura do paralelepípedo são provenientes de uma matriz (*MatrizDasCores*) que guarda o modelo na forma de seções bidimensionais. Porém, os dados gerados por uma inversão em 3D são vagamente diferentes. As matrizes obtidas neste caso contêm os valores das resistividades em cada célula, ao invés de um número inteiro associado a esta resistividade. Além do que as seções bidimensionais representam apenas uma e não várias fatias do paralelepípedo. Para mostrar dados de inversão em 3D deve-se fazer ajustes na interface

para a ler os dados de uma outra matriz (que não vai ser a *MatrizDasCores*), que guarde o resultado da inversão. As resistividades devem ser mostradas usando-se uma lista de cores associadas às mesmas, como foi feito no método IP.

4.7.3 – Armazenamento dos arquivos plotados

A interface usada para comparar gráficos permite confrontar dados de arquivos diferentes, porém, só permite armazenar, para livre escolha das colunas, um arquivo de cada vez. Para facilitar o acesso aos dados dos arquivos, seria útil guardar as matrizes dos arquivos que foram abertos, identificando cada matriz pelo nome do arquivo ao qual ela pertence, o que permitiria o acesso desses dados mais rapidamente.

4.7.4 – Opção para plotar os dados em ordem decrescente

Em alguns métodos geofísicos, como o Magnetotelúrico, costuma-se plotar os dados tanto em ordem crescente quanto em ordem decrescente (neste último caso o sentido usado é o da maior para a menor frequência). Para atender a estes casos deve-se colocar uma opção para que o usuário possa escolher a ordem dos valores no plot. De acordo com a opção escolhida deve-se inverter ou não a ordem dos dados do vetor *ColPPlot*, que guarda os valores da matriz.

5 - CONCLUSÃO

Esta dissertação trata do desenvolvimento de parte de um programa, o qual é usado como interface gráfica para uma suíte de aplicativos para modelagem numérica de dados geofísicos elétricos e eletromagnéticos: a eGs. A construção do programa foi dividida em duas parcelas. A primeira parcela (que não é abordada nesta dissertação) trata basicamente da *reformulação e ampliação da eGs*, enquanto que esta parcela do programa trata das *interfaces para entrada de dados (geração, modificação e visualização de modelos bi e tridimensionais), e saída de dados (apresentação dos dados de resposta do modelo e dos dados observados)*.

As interfaces apresentadas neste trabalho foram construídas com o objetivo de facilitar a construção e avaliação de modelos geofísicos elétricos e eletromagnéticos. O Capítulo 3 apresenta estas interfaces e suas opções. A interface para entrada de dados apresenta diversas opções que facilitam a alteração das cores das células do modelo (que representam propriedades elétricas) e a visualização de modelos em uma figura tridimensional, permitindo controlar com maior precisão a geometria do modelo geofísico proposto. A interface para saída de dados proporciona um variado leque de opções para comparar os dados de saída (resposta do modelo) com os dados observados através de gráficos de curvas, visualização de dados através de mapas de contorno, além de garantir um programa independente de outros programas por não depender de pacotes como o MATLAB. Para o método IP (Induced Polarization), construiu-se uma interface para visualizar os dados através de um espectro de cores. Para o método MCSEM (Marine Controlled Source Electromagnetic), construiu-se uma interface para o gerenciamento dos arquivos e do modo de apresentação dos dados de saída. As informações ao programador apresentadas no Capítulo 4 foram escritas com o intuito de impedir que seja necessário escrever, repetidamente, os mesmos códigos de compilação pelas diferentes gerações de alunos que venham a participar do projeto de construção do programa. Elas mostram informações a respeito do código fonte do programa e das estratégias usadas na construção deste programa. O Apêndice A desta dissertação consiste de um manual com exemplos para aqueles que desejam usar os recursos da eGs na modelagem e interpretação de dados elétricos e eletromagnéticos.

Durante a construção do software procurou-se, sempre que possível, optar por alternativas que levassem a um programa que apresentasse poucas dificuldades tanto para ser alterado pelo seu programador, quanto para ser utilizado pelo seu usuário. Como resultado, a interface gráfica

de entrada de dados é suficientemente flexível para atender a grande maioria dos métodos elétricos e eletromagnéticos, enquanto que a interface para saída de dados apresenta restrições apenas a grupos de métodos, mas ambas as interfaces podem ser facilmente estendidas para atender a novos métodos. Com a construção de um manual para o usuário, procurou-se facilitar o acesso ao programa, tentando evitar que o mesmo fique inutilizado. Espera-se que a eGs venha facilitar tanto o ensino dos métodos elétricos e eletromagnéticos nos cursos de graduação em geofísica e geologia, quanto o desenvolvimento de TCCs e dissertações de mestrado.

REFERÊNCIAS BIBLIOGRÁFICAS

- CANTÙ, M. 2001. **Dominando o Delphi 6 – A Bíblia**. Editora Makron Books, 962p.
- GOMES, J. & VELHO, L. 2003. **Fundamentos da computação gráfica**. IMPA, 603p. (Série de Computação e Matemática).
- MONTEIRO, H. D. M. 2001. **Interface gráfica para pré e pós-processamento de dados elétricos e eletromagnéticos em Geofísica**. Belém. Universidade Federal do Pará, Centro de Geociências, 56p (Dissertação de Mestrado).
- PEREIRA, G. L. 2005. **Estruturação e ampliação da eGs (electrical GEOPHYSICS suite)**. Belém. Universidade Federal do Pará, Centro de Geociências, 174p (Dissertação de Mestrado).
- RIJO, L. 1977. **Modeling of electric and electromagnetic data**. Salt Lake City, University of Utah. Department of geology and geophysics. 242p. (PhD Thesis).
- RIJO, L. 1992. **Teoria dos métodos elétricos e eletromagnéticos I e II**. Departamento de Geofísica, Universidade Federal do Pará (Notas de aula). www.rijo.pro.br
- SILVA, J. B. C., MEDEIROS, W. E., BARBOSA, V. C. F. 2001. Pitfalls in Nonlinear Inversion. **Pure appl. geophys.**, **158**: 945-964.
- VOGADO, G. E. R. 2005. **Estudo de efeitos de manto no VLF através de modelagem numérica**. Belém. Universidade Federal do Pará, Centro de Geociências (Dissertação de Mestrado em preparação).

APÊNDICE A – MANUAL DO USUÁRIO

A melhor maneira de se familiarizar com um novo programa é através de exemplos. Por isso serão dados dois exemplos explorando algumas opções da eGs. Será iniciado com um exemplo de modelagem 1D e depois de modelagem 2D.

A.1 – ONDE OBTER A eGs 3.0

A eGs 3.0 e a 1ª versão do manual serão distribuídos através do site de professor Luiz Rijo cujo endereço é www.rijo.pro.br.

Este manual, assim como a eGs, serão periodicamente atualizados com sugestões dos usuários e com implementação de novas opções de métodos elétricos e eletromagnéticos.

É importante informar que a eGs é um software livre, ou seja sem fins lucrativos, portanto é de suma importância a colaboração dos usuários para melhorar o desempenho do aplicativo.

A.2 – INSTALAÇÃO DA eGs 3.0 (Windows 2000 ou XP)

Para iniciar o processo de instalação do aplicativo, o usuário deve seguir os seguintes passos:

- 1º) Verifique se existe alguma versão anterior da eGs;
- 2º) Verifique se existe espaço suficiente em disco. A instalação completa da eGs ocupa entre 4 MB a 6MB de espaço no disco, isto vai depender da versão do Windows;
- 3º) Escolha o local de pasta para instalação c:\eGs. Este diretório irá receber todos os arquivos da distribuição;
- 4º) Inicie eGs setup wizard (setup_eGs.exe, mostrado na Figura A.1). Será apresentada a janela de boas-vindas (Figura A.2);



Figura A.1: Ícone de instalação da eGs 3.0.



Figura A.2: Janela de boas-vindas para instalação do programa

5º) Clique em “Next” para ir a próxima janela. A Figura A.3 mostra informação do local onde a eGs deve ser instalada;

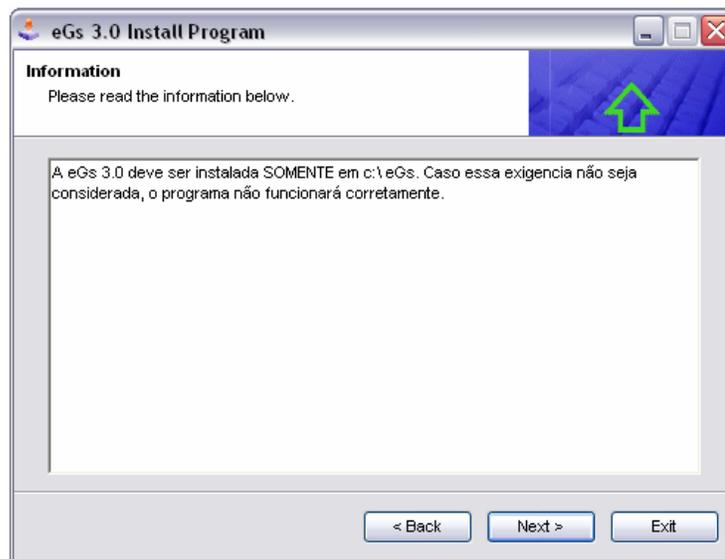


Figura A.3: Informação para instalação.

6º) Clique em “Next” para ir a próxima janela. A Figura A.4 mostra informações sobre o termo de licença para ser lido pelo usuário. Após lido, deve-se selecionar “I agree with the above terms and conditions” para concordar e “I do not agree” para não concordar;

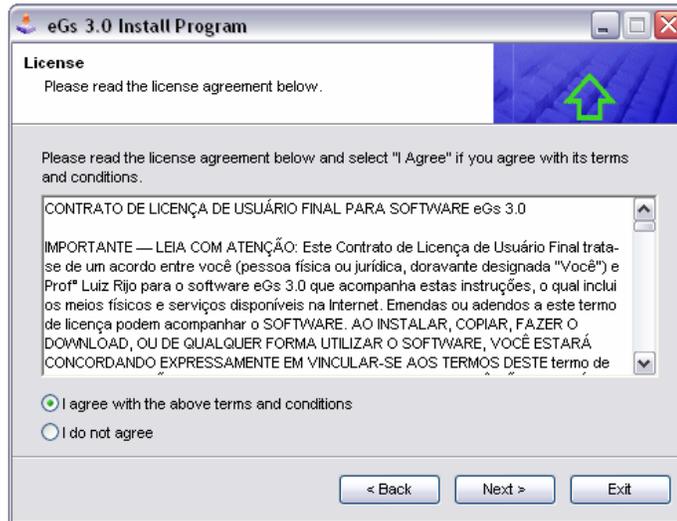


Figura A.4: Termo de licença.

7º) Se o termo de licença estiver sido concordado, clique em “Next” para ir a próxima janela. A Figura A.5 mostra a informação sobre o local onde a eGs será instalada, que deve ser em c:\eGs, pois em outra pasta o aplicativo não funcionará. Abaixo do diretório de instalação, estão as informações sobre o espaço necessário e o espaço disponível em disco;

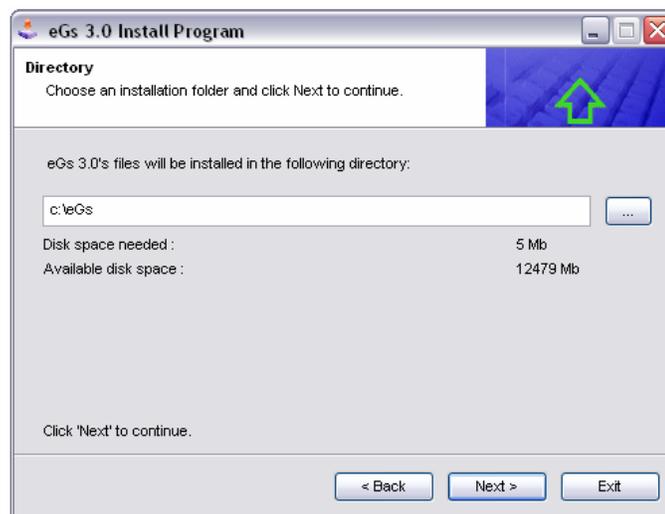


Figura A.5: Informações do diretório, do espaço necessário e livre no disco rígido.

8º) Clique em “*Start*” para instalar o programa. A Figura A.6 mostra a janela de confirmação de instalação no diretório c:\eGs.

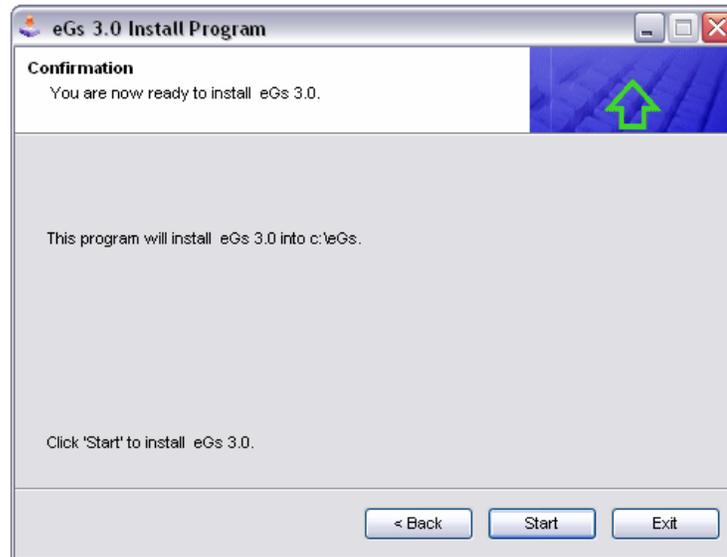


Figura A.6: Janela de confirmação da instalação.

9º) Clique em “*Next*” para finalizar a instalação. A Figura A.7 mostra a janela informativa de sucesso da instalação do programa.

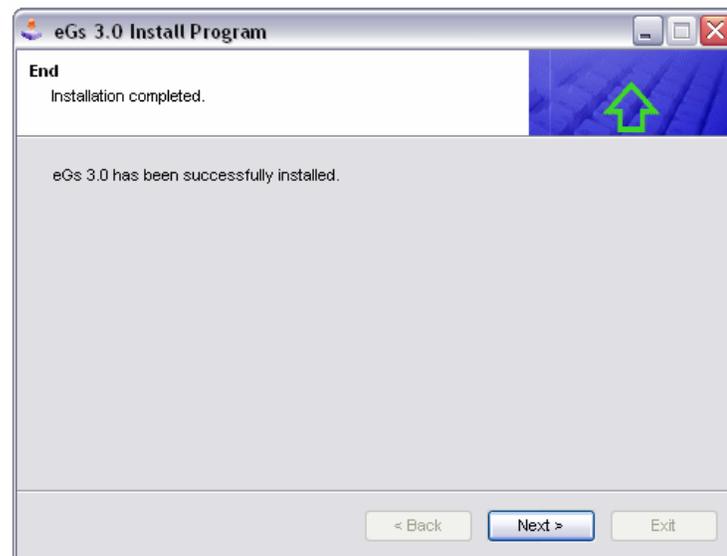


Figura A.7: Janela que informa a finalização da instalação do programa.

A.3 – COMO USAR A eGs 3.0

A.3.1 – 1D (uma dimensão)

A seguir, vamos mostrar um exemplo de modelagem 1D usando o método magnetotelúrico.

A.3.1.1 – Magnetotelúrico (Problema Direto)

Para utilizar a eGs em uma aplicação geofísica que utiliza o método magnetotelúrico, faz-se os seguintes passos:

1º) Escolher o tipo de problema.

Para construir um novo ou abrir um arquivo ao modelo, primeiramente é necessário escolher, na janela “Métodos” (Figura A.8), a dimensão, em seguida dê dois clique em Sondagem Magnetotelúrica e o tipo de problema (direto ou inverso) que deseja usar.

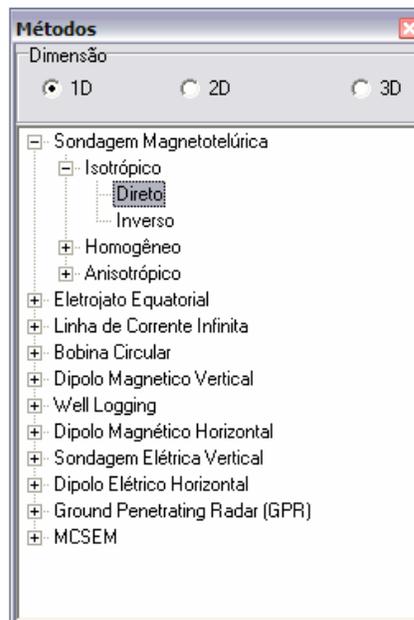


Figura A.8: Janela Métodos

2º) Construir um *modelo geométrico*.

Depois de feita a escolha do tipo de problema, há duas maneiras de editar um modelo geométrico:

- i. A partir da edição de um *modelo geométrico* novo;
- ii. A partir de um *modelo geométrico* salvo anteriormente.

i) A partir da edição de um modelo geométrico novo:

Para abrir uma janela “Modelo” para editar um modelo estratificado em camadas, primeiramente deve ser escolhido o tipo de problema na janela “Métodos”, depois basta seguir um dos dois caminhos seguintes:

- Janela Seleção → Modelo → Novo (Figura A.9);
- Menu → Arquivo → Novo → Modelo (Figura A.10).



Figura A.9: Janela “Seleção” para abrir um *modelo geométrico* novo.

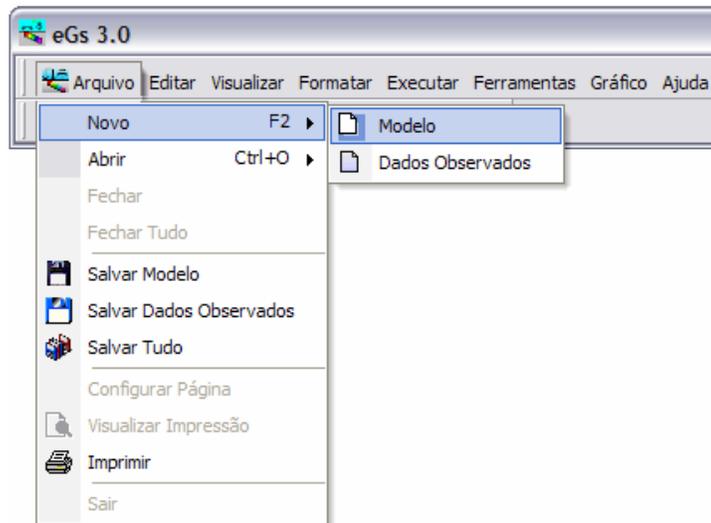


Figura A.10: Abrir um *modelo geológico* novo através do Menu.

Surgirá a janela “Modelo” mostrada na Figura A.11.



Figura A.11: Janela “Modelo” para início da edição de um *modelo geológico*.

Como exemplo, usou-se um meio estratificado com cinco camadas, como mostra a Figura A.12.

Resistividade	Permeabilidade magnética relativa	Espessura
↓	↓	↓
100 ohm-m	1	1 km
200 ohm-m	1	2 km
2 ohm-m	1	1 km
1000 ohm-m	1	30 km
10 ohm-m	1	

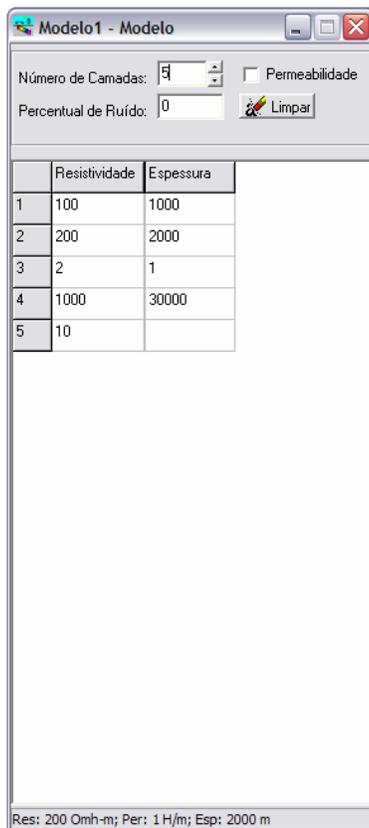
Figura A.12: *Modelo geoeletrico* 1D usado como exemplo.

Observe na Figura A.12 que a ultima camada é o substrato, não sendo necessário ser editada, portanto a eGs não permite editar a espessura do substrato.

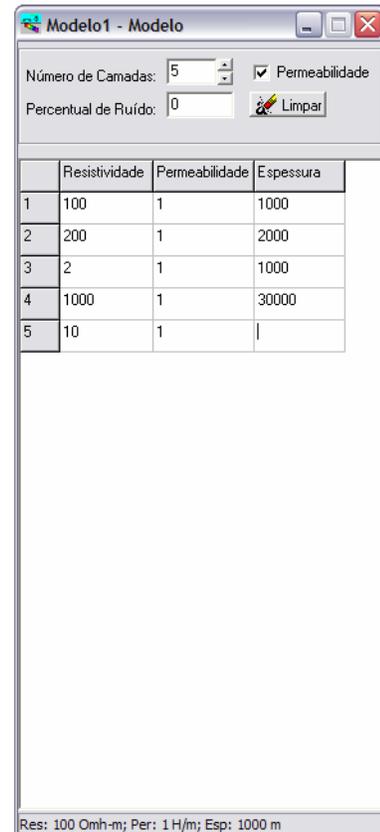
As Figuras A.13.(a) e A.13.(b) são as janelas “Modelo”, as quais servem para editar o *modelo geoeletrico*, onde:

⇒ O *Número de camadas* altera o número de linhas da tabela da janela “Modelo”, podendo ser através do teclado digitando valores entre 1 e 100 inclusive ou pelo mouse através de clique nas setas. Se o clique for na seta apontada para cima, os valores são aumentados de um em um, e se para baixo os valores são diminuídos;

⇒ A primeira e segunda colunas da grade, na Figuras A.13.(a), representam, respectivamente, resistividade (ohm-m) e espessura (m) das camadas.



(a)



(b)

Figura A.13: (a) Janela “Modelo” com permeabilidade magnética relativa igual a 1 em todas camadas. (b) Janela “Modelo” para alteração da permeabilidade magnética relativa.

A eGs usa como padrão a permeabilidade relativa igual a 1. Caso o usuário deseje alterar a permeabilidade relativa, basta marcar o item *permeabilidade*, e uma coluna chamada permeabilidade será mostrada entre as colunas resistividade e espessura, como mostra a Figura A.13.(b).

Para salvar o *modelo geológico* pode ser de duas maneiras: através do item salvar modelo do menu (Figura A.14) ou quando o usuário desejar fechar a janela “Modelo” e não salvou as alterações (Figura A.15);

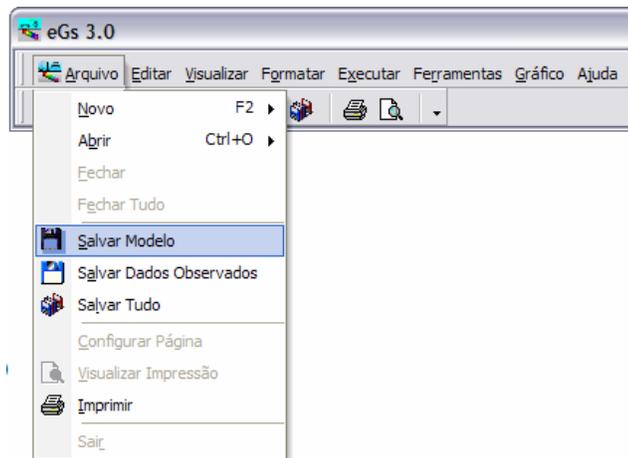


Figura A.14: Opção *Salvar Modelo* do Menu.

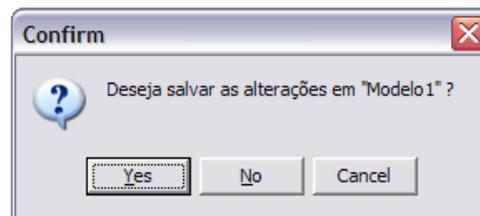


Figura A.15: Janela de aviso quando um *modelo geoeétrico* está sendo fechado, mas as alterações não foram salvas.

ii) A partir de um modelo geoeétrico salvo anteriormente

Para abrir uma janela “Modelo” com informações editadas anteriormente, primeiramente deve ser escolhido o tipo de problema na janela “Métodos”, depois basta seguir um dos dois caminhos seguintes:

- Janela Seleção → Modelo → Abrir (Figura A.16);
- Menu → Arquivo → Abrir → Modelo (Figura A.17).



Figura A.16: Janela “Seleção” para abrir um *modelo geológico* salvo.

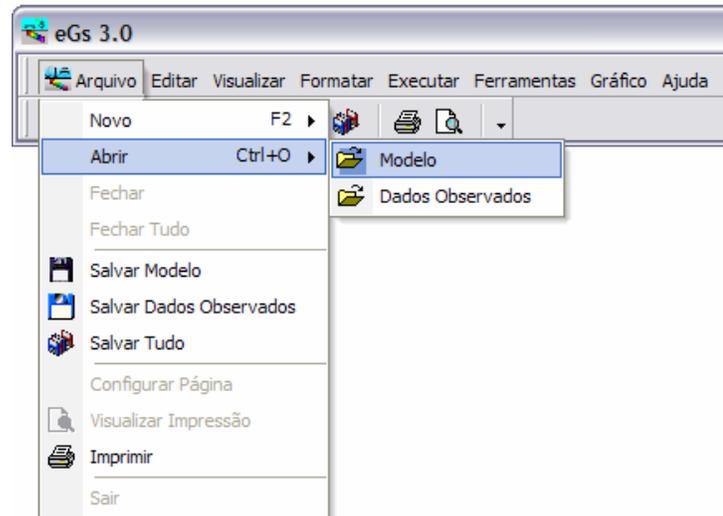


Figura A.17: Abrir um *modelo geológico* salvo através do Menu.

Surgirá a janela “Abrir”, mostrada na Figura A.18. Deve-se procurar arquivos com a extensão *.MGE. Em seguida a janela da Figura A.19 para reedição.

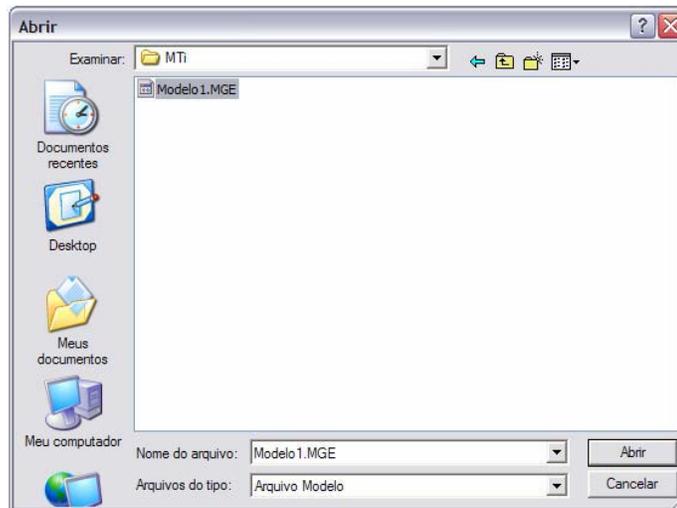


Figura A.18: Janela amigável para abertura de arquivos.

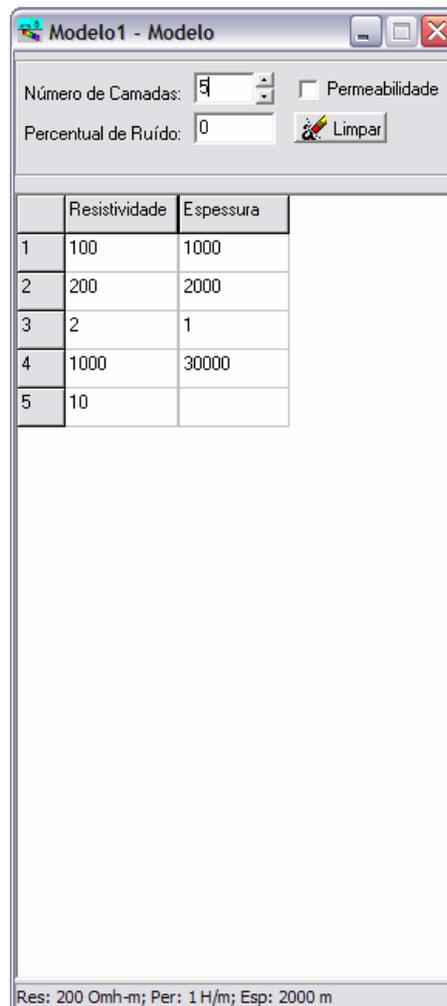


Figura A.19: Janela “Modelo” aberta para reedição de um arquivo salvo anteriormente.

3º) Construir os *dados observados*.

Depois de feita a escolha do tipo de problema, há duas maneiras de editar os dados observados:

- i. A partir da edição de dados *observados* novos;
- ii. A partir de *dados observados* salvo anteriormente.

i) A partir da edição de dados observados novos:

Para abrir uma janela “Dados Observados” para editar informações novas, primeiramente deve ser escolhido o tipo de problema na janela “Métodos”, depois basta seguir um dos dois caminhos seguintes:

- Janela Seleção → Dados Observados → Novo (Figura A.20);
- Menu → Arquivo → Novo → Dados Observados (Figura A.21).



Figura A.20: Janela “Seleção” para abrir a janela “Dados Observados” nova.

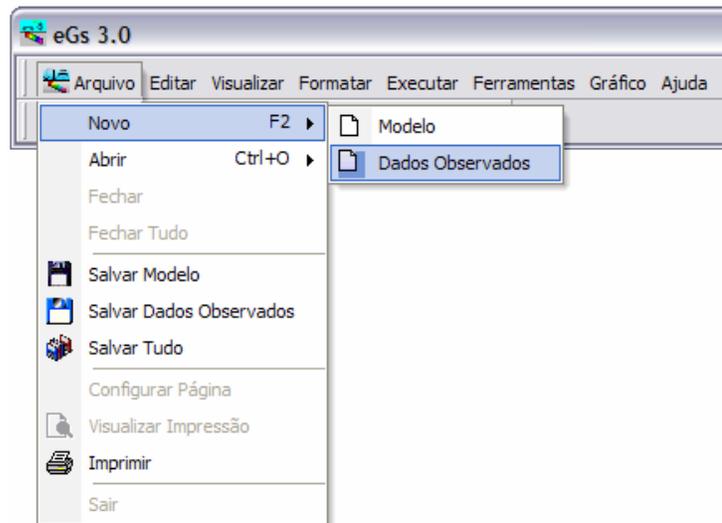


Figura A.21: Abrir uma janela “Dados Observados” nova através do Menu.

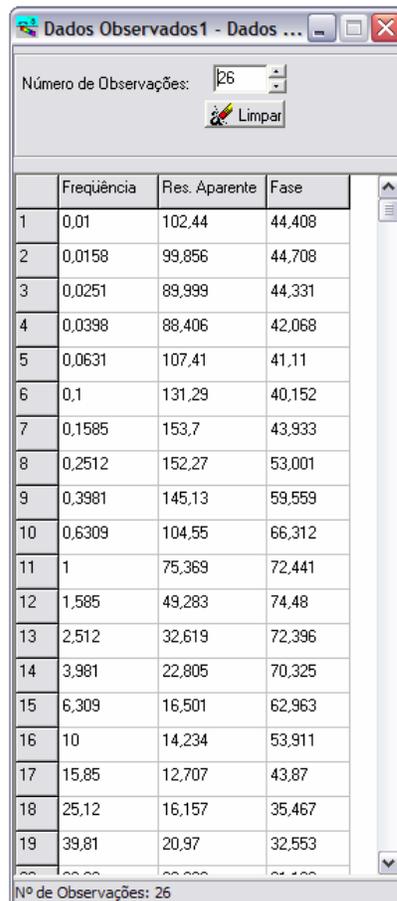
Surgirá a janela “Dados Observados” mostrada na Figura A.22.



Figura A.22: Nova janela “Dados Observados” para início da edição.

Como exemplo, usou-se 26 observações de campo com as seguintes frequências: 0.01, 0.01585, 0.02512, 0.03981, 0.06309, 0.1, 0.1585, 0.2512, 0.3981, 0.6309, 1, 1.585, 2.512, 3.981, 6.309, 10, 15.85, 25.12, 39.81, 63.09, 100, 158.5, 251.2, 398.1, 630.9 e 10000 Hz (Figura A.23).

No problema direto a resistividade aparente e fase não são exigidas somente se o objetivo for a interpretação dos dados por tentativa e erro usando a modelagem numérica, mas não há restrição caso o usuário não deseje editar essas duas propriedades físicas nesta etapa.



	Frequência	Res. Aparente	Fase
1	0,01	102,44	44,408
2	0,0158	99,856	44,708
3	0,0251	89,999	44,331
4	0,0398	88,406	42,068
5	0,0631	107,41	41,11
6	0,1	131,29	40,152
7	0,1585	153,7	43,933
8	0,2512	152,27	53,001
9	0,3981	145,13	59,559
10	0,6309	104,55	66,312
11	1	75,369	72,441
12	1,585	49,283	74,48
13	2,512	32,619	72,396
14	3,981	22,805	70,325
15	6,309	16,501	62,963
16	10	14,234	53,911
17	15,85	12,707	43,87
18	25,12	16,157	35,467
19	39,81	20,97	32,553
20	63,09	22,805	32,553
21	100	22,805	32,553
22	158,5	22,805	32,553
23	251,2	22,805	32,553
24	398,1	22,805	32,553
25	630,9	22,805	32,553
26	10000	22,805	32,553

Figura A.23: Nova janela “Dados Observados” editada.

O item *Número de Observações* da janela “Dados observados” tem a função de alterar as linhas da tabela, as quais representam cada observação, que podem ser modificadas através do teclado digitando valores da frequência, da resistividade aparente e fase, respectivamente, para a primeira, segunda e terceira colunas. O botão limpar, tem a finalidade de deixar em branco todos os campos da grade.

Para salvar os *dados observados* pode ser de duas maneiras: através do item salvar Dados Observados do menu (Figura A.24) ou quando o usuário desejar fechar a janela “Dados Observados” e não salvou as alterações (Figura A.25);

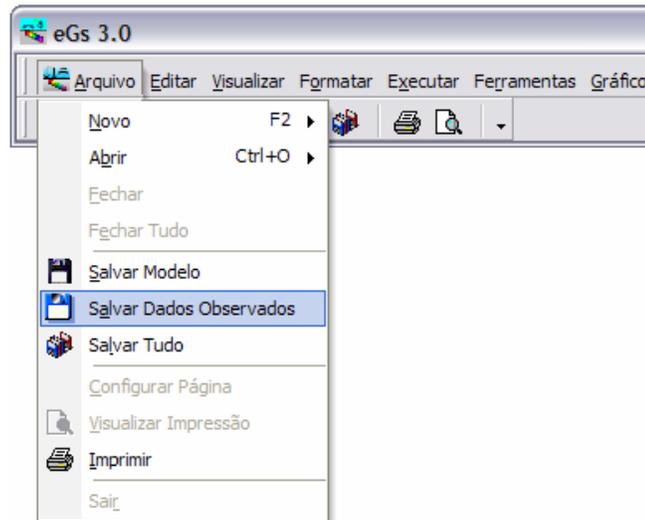


Figura A.24: Opção *Salvar Dados Observados* do Menu

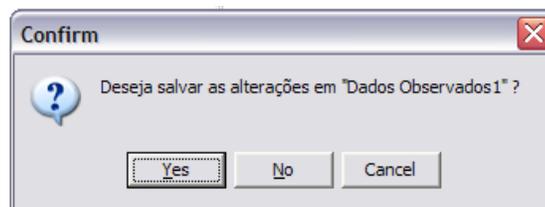


Figura A.25: Janela de aviso quando os *dados observados* estão sendo fechados, mas as alterações não foram salvas.

ii) A partir de dados observados salvos anteriormente:

Para abrir uma janela “Dados Observados” com informações editadas anteriormente, primeiramente deve ser escolhido o tipo de problema na janela “Métodos”, depois basta seguir um dos dois caminhos:

- Janela Seleção → Dados Observados → Abrir (Figura A.26);
- Menu → Arquivo → Abrir → Dados Observados (Figura A.27).



Figura A.26: Janela “Seleção” para abrir *dados observados* salvos.

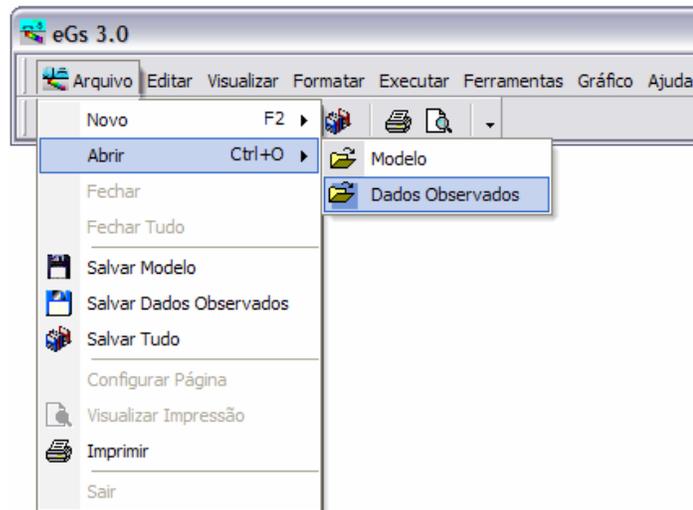


Figura A.27: Abrir os *dados observados* salvos através do Menu.

Surgirá a janela “Abrir”, mostrada na Figura A.28. Deve-se procurar arquivos com a extensão *.OBS. Em seguida a janela da Figura A.29 para reedição.

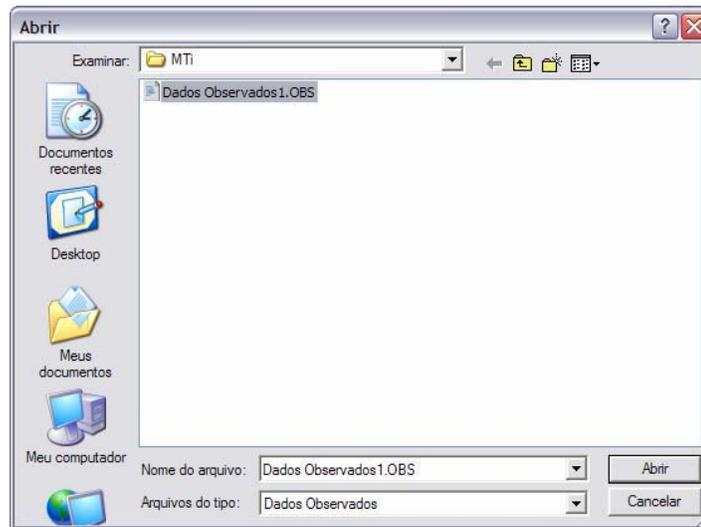


Figura A.28: Janela amigável para abertura de arquivos.

	Frequência	Res. Aparente	Fase
1	0,01	102,44	44,408
2	0,0158	99,856	44,708
3	0,0251	89,999	44,331
4	0,0398	88,406	42,068
5	0,0631	107,41	41,11
6	0,1	131,29	40,152
7	0,1585	153,7	43,933
8	0,2512	152,27	53,001
9	0,3981	145,13	59,559
10	0,6309	104,55	66,312
11	1	75,369	72,441
12	1,585	49,283	74,48
13	2,512	32,619	72,396
14	3,981	22,805	70,325
15	6,309	16,501	62,963
16	10	14,234	53,911
17	15,85	12,707	43,87
18	25,12	16,157	35,467
19	39,81	20,97	32,553
20	59,559	26,108	29,108
21	89,999	32,108	26,108
22	131,29	39,108	23,108
23	199,856	47,108	20,108
24	299,856	56,108	17,108
25	449,856	66,108	14,108
26	699,856	77,108	11,108

Nº de Observações: 26

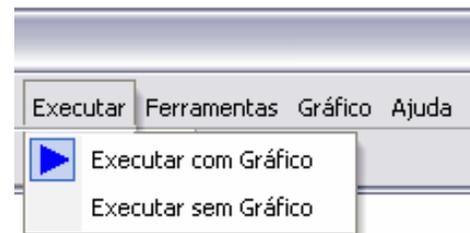
Figura A.29: Janela “Dados Observados” aberta para reedição de um arquivo salvo anteriormente.

4º) Executar a modelagem numérica:

Na eGs 3.0 a modelagem numérica só pode ser praticada após o *modelo geométrico* e *dados observados* estarem editados. Feito isso, o botão “Executar”, que pode ser encontrado na janela “Seleção” (Figura A.30(a)) ou no item *Executar* do Menu (Figura A.30(b)), é ativado e com apenas um clique a modelagem é obtida, que pode ser com a geração de figuras para interpretação (Figura A.31) ou apenas armazena os dados de saída, isso vai depender da escolha feita pelo usuário nos subitens do Menu *Executar* (Figura A.30(b)).



(a)



(b)

Figura A.30:(a) Janela “Seleção” com o botão *Executar* ativado (b) Itens *Executar Com Gráfico* e *Executar Sem Gráfico*.

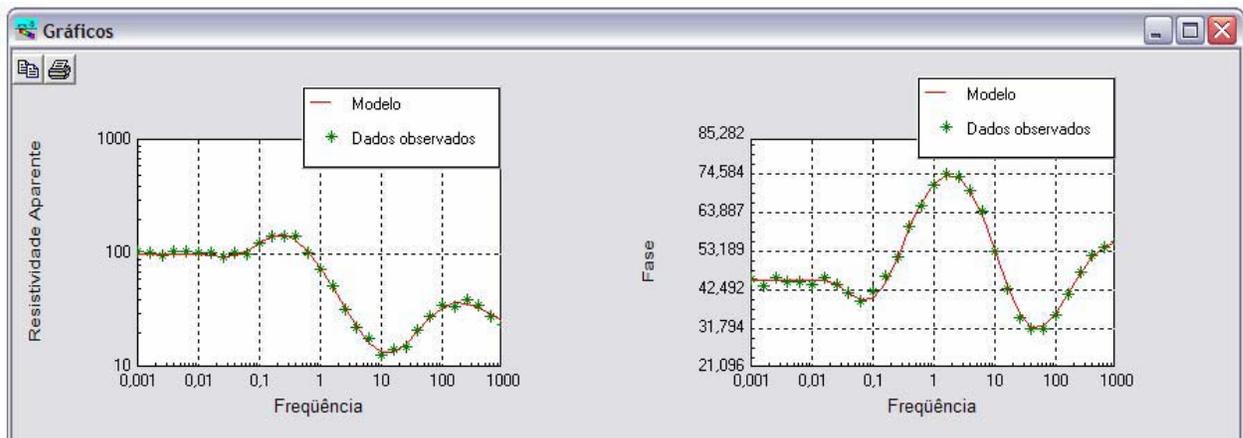


Figura A.31: Resultado da modelagem numérica com o item *Executar Com Gráfico* ativado.

Os arquivos do resultado do modelo *geométrico* e *dados observados* são salvos em uma pasta com o mesmo nome dado ao arquivo com extensão *.MGE. Por exemplo, se o *modelo geométrico* foi salvo em *C:\eGs\Exemplos\1D\MT\MTi\Mt1D.MGE*, então os dados de saída serão armazenados em uma pasta com o nome *Mt1D*. Veja Figura A.32.

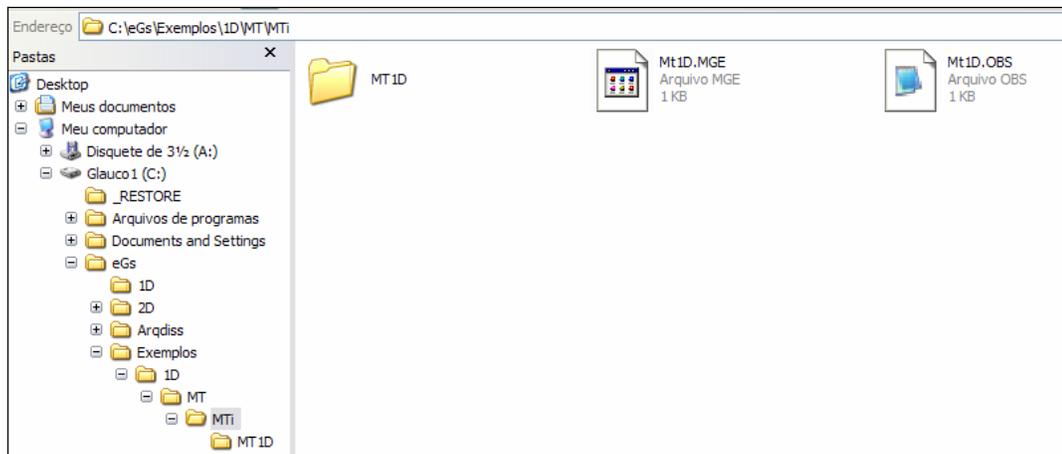


Figura A.32: Ilustração de uma modelagem usando o arquivo *Mt1D.MGE* como *modelo geométrico*.

Caso o usuário não queira salvar os arquivos *.MGE e *.OBS, os quais representam, respectivamente, o *modelo geométrico* e os *dados observados*, mas deseja gerar o gráfico à interpretação, o aplicativo armazena temporariamente os arquivos de saída, constrói o gráfico e ao fechar a eGs 3.0 esses arquivos são apagados do disco rígido.

5º) Interpretação dos dados (gráficos de curvas)

Aacionando-se a função *Executar com Gráfico*, os gráficos já aparecem automaticamente após a execução do programa. Mas caso o usuário prefira optar por executar a modelagem sem gráfico (botão *Executar sem Gráfico*) para interpretar os dados mais tarde, será necessário abrir os arquivos que contêm estes dados. Para abrir os arquivos que contêm os dados de resposta do modelo ou os dados observados, o usuário deve clicar em Menu→Gráfico→Curvas→ Abrir Arquivo (Figura A.33). A caixa de diálogo padrão do *Windows* aparece para que o usuário possa procurar o arquivo. No caso do método magnetotelúrico, o arquivo escolhido deve ser do tipo *.RAF ou *.OBS (com dados do método).

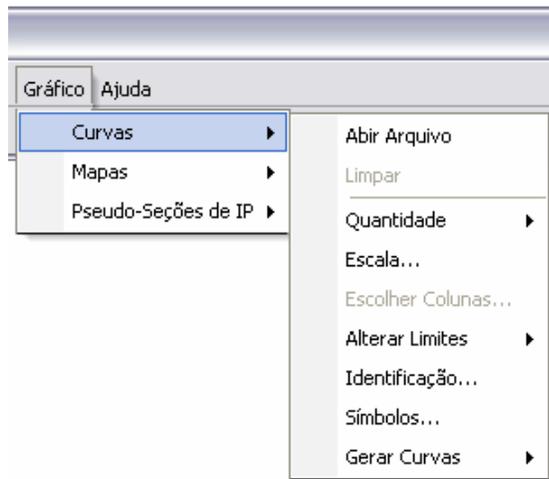
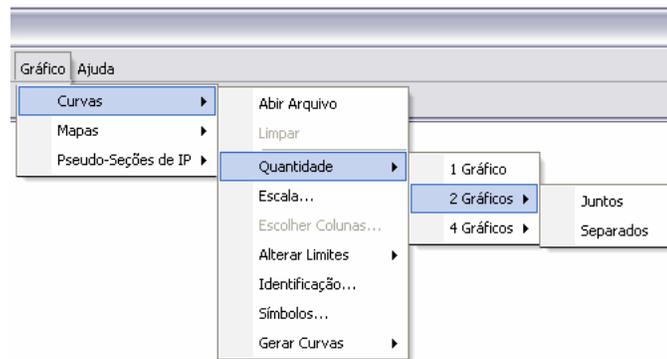
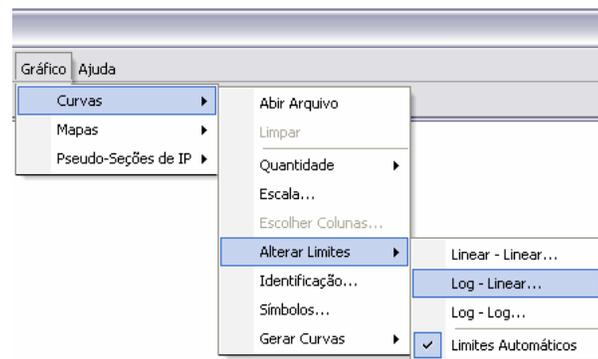


Figura A.33: Item *Abrir Arquivo*

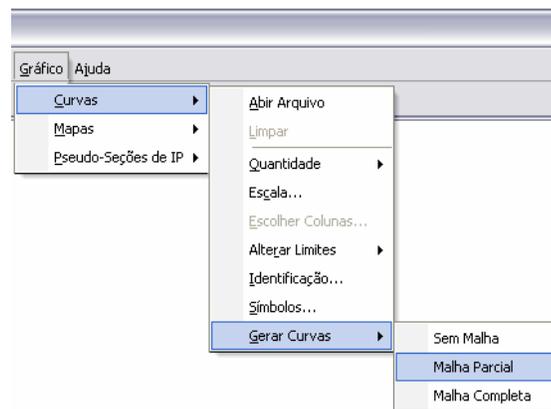
Caso o arquivo não tenha nenhum problema com os seus dados, já é possível fazer um gráfico. Basta escolher uma das opções no subitem *Gerar Curvas* (*Sem malha*, *Malha parcial*, *Malha completa*). A Figura A.34.(c) mostra as opções do subitem *Gerar Curvas*.



(a)



(b)



(c)

Figura A.34: Sub-menus do menu *Gráfico*. A figura (a) mostra os subitens do item *Quantidade* (um, dois ou quatro gráficos, juntos ou separados). A figura (b) exhibe os subitens do item *Alterar Limites* que permitem alterar os intervalos de visualização dos gráficos nas escalas *Linear-Linear*, *Log-Linear* ou *Log-Log*, ou escolher a opção *Limites Automáticos*. A figura (c) mostra os subitens do item *Gerar Curvas*, os quais permitem escolher a malha (grade) que cobre os gráficos (*Sem malha*, *Malha parcial* ou *Malha completa*).

A eGs3.0 possui diversas opções para fazer os seus gráficos. Dentre estas opções temos:

- *Quantidade*: Permite escolher quantos gráficos vão aparecer de uma só vez (Figura A.34.(a)).
- *Escala* (Tipo de gráfico): Permite escolher a escala que será usada em cada um dos gráficos. Dentre as escalas usadas temos *linear-linear* (linear nos eixos X e Y), *log-linear* (linear no eixo X e logarítmica no eixo Y) e *log-log* (logarítmica nos eixos X e Y). A janela que permite fazer este tipo de alteração está ilustrada na Figura A.35.



Figura A.35: Janela que permite escolher as escalas de cada gráfico.

- *Escolher colunas*: Permite escolher quantas e quais colunas do arquivo serão usadas para cada gráfico (janela na Figura A.36).



Figura A.36: Janela que permite escolher quais colunas do arquivo serão usadas para gerar o gráfico.

- *Alterar limites* (Intervalos de visualização): Permite escolher os intervalos de visualização de cada gráfico (Figura A.37) de acordo com a escolha de um dos subitens de *Alterar limites: linear-linear, log-linear, log-log* (Figura A.34.(b)).

	Gráfico 1	Gráfico 2	Gráfico 3	Gráfico 4
Valor inicial(X):	1.0E 0	1.0E 0	1.0E 0	1.0E 0
Valor Final(X):	1.0E 3	1.0E 3	1.0E 3	1.0E 3
Valor Inicial(Y):	-100	-100	-100	-100
Nº de Intervalos(Y):	4	4	4	4
Valor do Intervalo(Y):	50	50	50	50

OK

Figura A.37: Janela que permite escolher os intervalos de visualização dos gráficos. Esta janela em particular permite alterar os intervalos de visualização de gráficos em escala Log-Linear.

- *Identificação*: Permite escolher os textos que identificam a janela, os eixos e as curvas (Figura A.38).

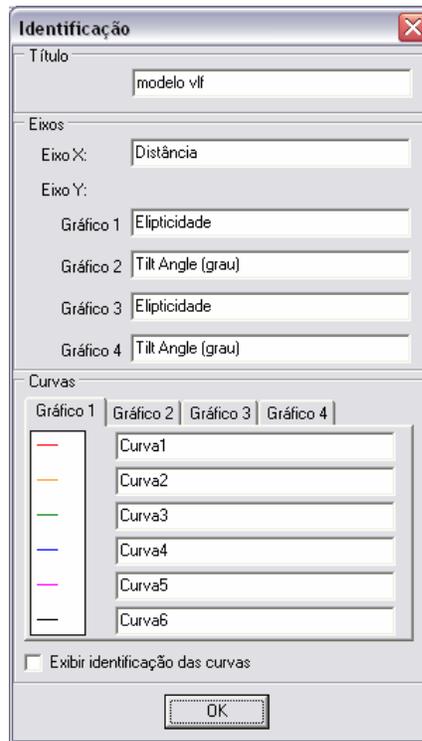


Figura A.38: Janela que permite escolher os textos que identificam a janela, os eixos e as curvas.

- *Símbolos*: Permite escolher quais símbolos vão representar os pontos no gráfico (Figura A.39).

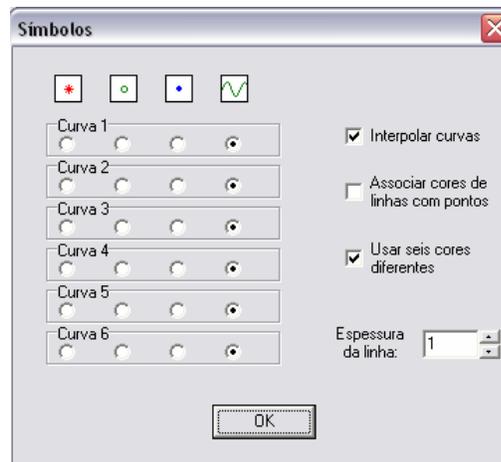


Figura A.39: Janela que permite escolher os símbolos que representarão os pontos nos gráficos.

- *Gerar Curvas* (Grade): Permite escolher que tipo de malha (grade) vai cobrir os gráficos. Este item está subdividido em *Sem malha*, *Malha parcial* e *Malha completa* (Figura A.34.(c)).

Para comparar os dados armazenados em mais de um arquivo, o usuário deve primeiro gerar as colunas que deseja do arquivo que está aberto, usando a caixa de diálogo *Escolher colunas* (Figura A.36). Em seguida, sem limpar o gráfico que já está na tela, o usuário deve abrir o próximo arquivo, sendo que este novo arquivo deve ter a primeira coluna da sua matriz igual à do arquivo anterior, caso contrário não é possível fazer o gráfico. É importante que não se limpe o gráfico que está na tela antes de abrir o novo arquivo, caso contrário os dados do arquivo anterior são apagados da memória. Finalmente, caso não haja nenhum problema com o novo arquivo que foi aberto, a caixa de diálogo *Escolher colunas* aparece automaticamente para o usuário escolher quais colunas do novo arquivo serão usadas para gerar as curvas. Após clicar no botão *OK* os dados do novo arquivo gerarão os gráficos junto com os dados do(s) arquivo(s) anterior(es).

A.3.2 – 2D (duas dimensões)

A seguir, vamos mostrar um exemplo de modelagem 2D usando o método IP e Resistividade.

A.3.2.1 – IP e Resistividade 2D – Problema Direto

Para utilizar a eGs em uma aplicação geofísica que utiliza o método IP num modelo 2D, deve-se seguir estes passos:

1º) Escolher o tipo de problema

Para construir um modelo novo ou abrir um modelo do arquivo, primeiramente é necessário escolher, na janela “Métodos” (Figura A.40), a dimensão 2D, em seguida dar dois clicks em + *IP e Resistividade* e escolher o tipo de problema (neste caso problema direto).

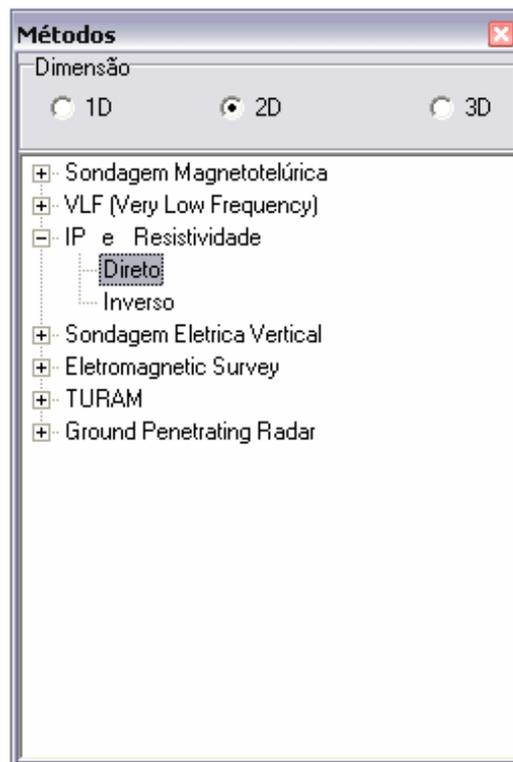


Figura A.40: Janela Métodos

2º) Chamar a janela usada para editar o modelo

Depois de feita a escolha do tipo de problema, há duas maneiras de editar um *modelo geométrico*:

- A partir de um *modelo geométrico* novo;
- A partir de um *modelo geométrico* salvo anteriormente.

i) A partir de um modelo geométrico novo:

Para abrir uma janela “Modelo” nova, primeiramente deve ser escolhido o tipo de problema na janela “Métodos”, depois basta seguir um dos dois caminhos:

- Janela Seleção → Modelo → Novo (Figura A.41);
- Menu → Arquivo → Novo → Modelo (Figura A.42).



Figura A.41: Janela “Seleção” para abrir um *modelo geométrico* novo.

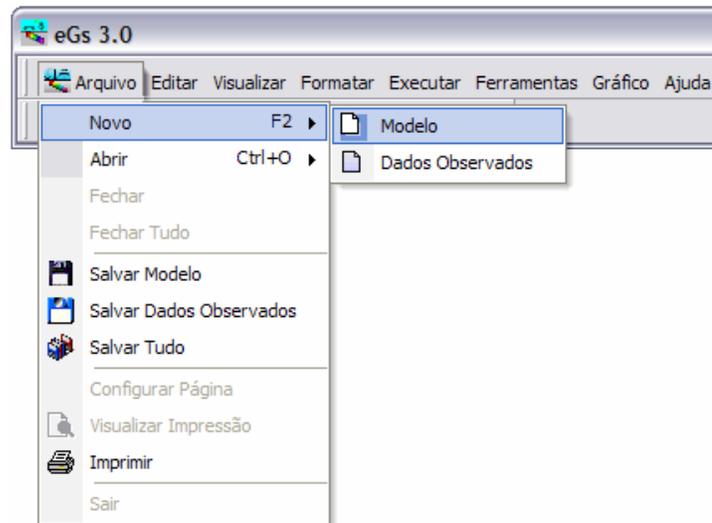


Figura A.42: Abrir um *modelo geológico* novo através do Menu.

ii) A partir de um modelo geológico salvo anteriormente

Para abrir uma janela “Modelo” com informações editadas anteriormente, primeiramente deve ser escolhido o tipo de problema na janela “Métodos”, depois basta seguir um dos dois caminhos:

- Janela Seleção → Modelo → Abrir (Figura A.43);
- Menu → Arquivo → Abrir → Modelo (Figura A.44).



Figura A.43: Janela “Seleção” para abrir um *modelo geológico* salvo.

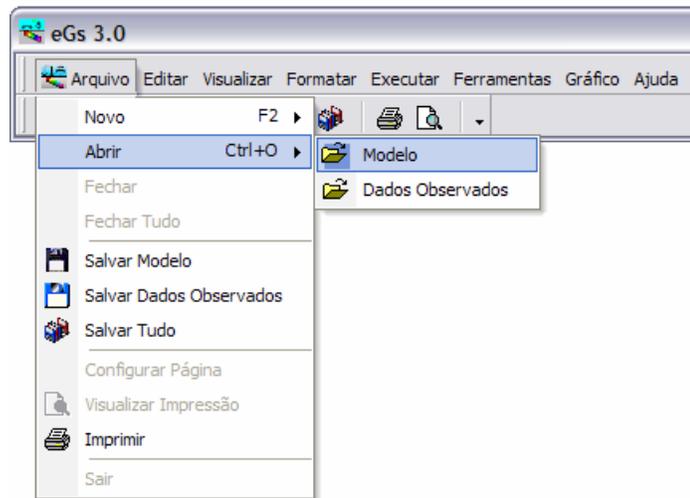


Figura A.44: Abrir um *modelo geológico* salvo através do Menu.

Após chamar um modelo novo ou abrir um do arquivo, aparecerá a janela usada para edição do modelo. A Figura A.45 mostra esta janela.

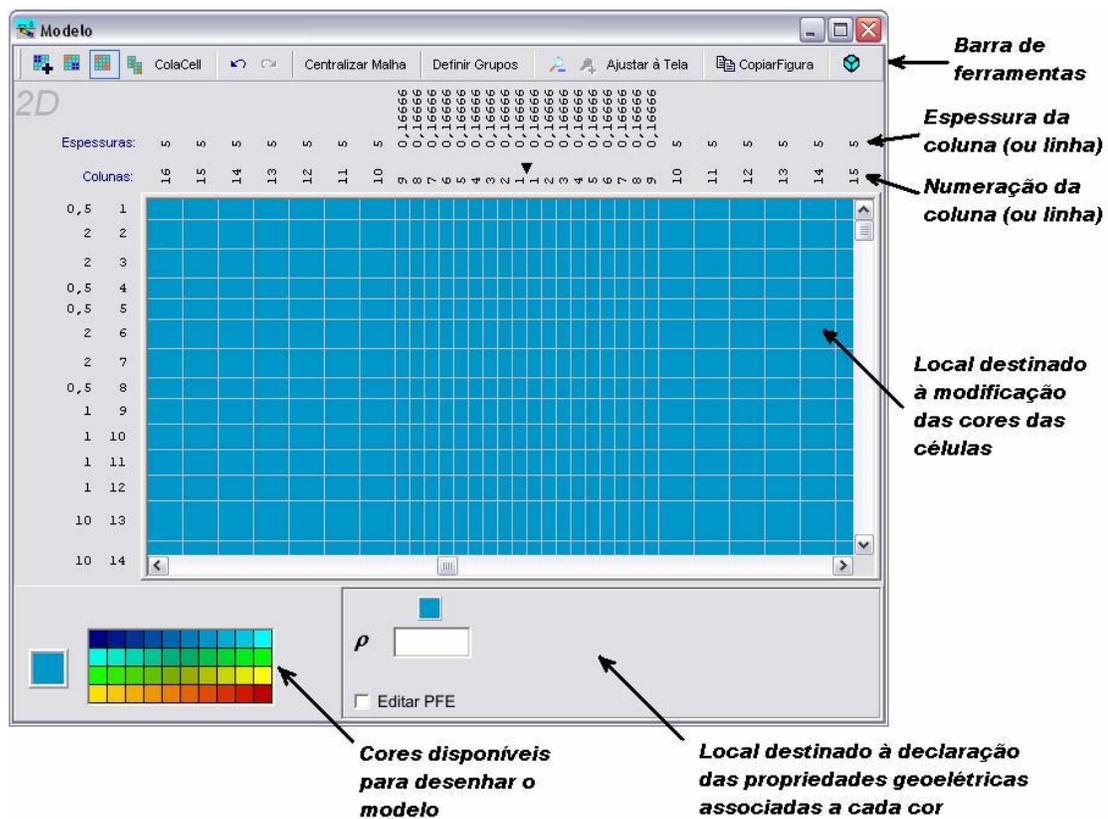


Figura A.45: Janela usada para editar o modelo 2D.

Na janela usada para desenhar o *modelo geométrico* (Figura A.45) estão todas as opções usadas para alterar a forma e as propriedades elétricas do modelo. A barra de ferramentas da janela usada para desenhar o *modelo geométrico* está ilustrada na Figura A.46. Abaixo temos uma breve explicação da função de cada item da barra.

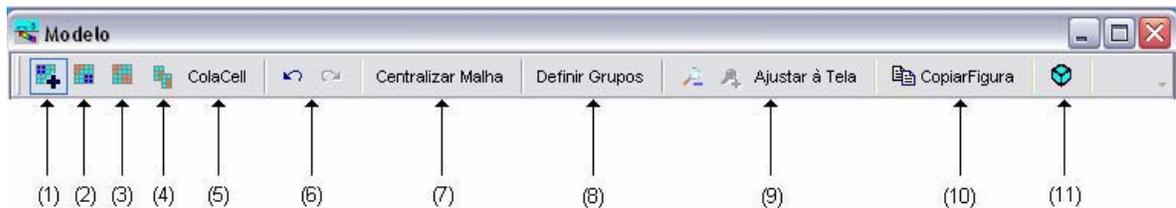


Figura A.46: Barra de ferramentas da janela usada para desenhar o *modelo geométrico*.

- (1): Permite criar um corpo retangular (arrastando-se o cursor do mouse na região onde as células terão suas cores modificadas);
- (2): Permite mudar a cor de um corpo (clicando-se o cursor do mouse em uma das células do corpo);
- (3): Permite mudar a cor das células de mesma cor (clicando-se o cursor do mouse em uma das células da cor que deve ser modificada);
- (4): Permite copiar as cores das células de uma região retangular (arrastando-se o cursor do mouse na região onde as células terão suas cores copiadas);
- (5): Permite colar as células da região copiada (clicando-se o cursor do mouse no local onde a região copiada será colada);
- (6): Desfaz ou refaz a última ação;
- (7): Move a barra de rolagem para centralizar a malha;
- (8): Chama a janela que permite definir as dimensões da malha;
- (9): Ferramentas de zoom;
- (10): Copia para a área de transferência o trecho da malha que está visível;
- (11): Chama a janela que permite visualizar o modelo em 3D.

3º) Editar o modelo

Na fase de criação ou de modificação de um modelo, o usuário pode alterar as características do mesmo de três formas. A seqüência para se fazer esta modificação deve ser, preferencialmente, a apresentada a seguir. As características do modelo que podem ser alteradas são:

- i. Dimensões (células da malha);
- ii. Forma;
- iii. Propriedades físicas.

i) Definir as dimensões do modelo

Para definir as dimensões da malha, o usuário deve clicar no botão *Definir Grupos* (item (8) da Figura A.46) na barra de ferramentas da janela usada para editar o modelo. A janela ilustrada na Figura A.47 aparece para que o usuário possa declarar as dimensões da malha. Nesta janela o usuário declara o número de nós da malha e as espessuras das células da mesma. As espessuras das células são declaradas em grupos. Em cada direção, defini-se o número de grupos de células, o número de células em cada grupo e a espessura das células do grupo.

Modelo1 - Definir Grupo

Número de Nós

Na Horizontal(X): 181 Na Vertical(AR): 7 Ruído (%): 0

Na Horizontal(Y): 51 Na Vertical(TERRA): 18

Posições dos eletrodos

Nº de Níveis: 6

Nº de Colunas: 13 Nº de eletrodos: 25

Resistividade Aparente
 Todos

Esquerda	-48	-44	-40	-36	-32	-28	-24
Direita	0	4	8	12	16	20	24

Grupos Na Horizontal (X)

Número de Grupos: 11

Grupos	1	2	3	4	5	6	7
Nº de Células	1	3	1	1	4	160	4
Espessura	10	5	2	1	0,5	0,25	0,5

Grupos Na Vertical (TERRA)

Número de Grupos: 10

Grupos	Nº de Células	Espessura
1	2	0,25
2	2	0,25
3	4	0,25
4	2	0,25
5	2	0,25
6	1	0,5
7	1	1
8	4	0,25
9	2	0,25
10	2	0,25

Colunas: 13.

Figura A.47: Janela usada para definir as dimensões da malha usada no *modelo geológico* do método IP e Resistividade 2D.

ii) Definir a forma do modelo

Para definir a forma do modelo, o usuário deve associar cada tipo de material (rocha) que compõe o modelo a uma cor. Para desenhar os corpos e camadas no modelo, o usuário pode simplesmente selecionar uma nova cor e clicar nas devidas células, ou usar as ferramentas para a edição de modelos (itens (1), (2), (3), (4) e (5) da Figura A.46). A Figura A.48 mostra o modelo construído.

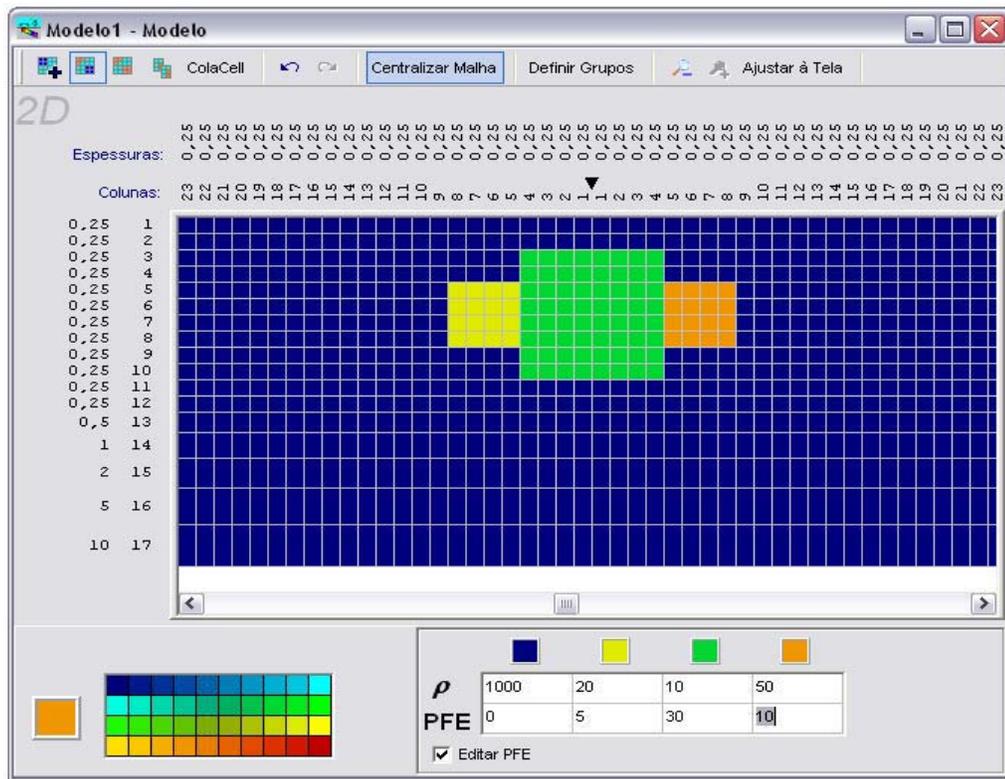


Figura A.48: Ilustração da forma do *modelo geoeletrico* usado na modelagem usando o método IP e Resistividade 2D.

iii) Declarar as propriedades físicas do modelo

Para associar os valores de propriedades físicas (que para o caso do IP são a resistividade e o PFE) a cada cor, o usuário deve preencher os campos que aparecem automaticamente no canto direito-inferior da janela (veja a Figura A.45). A Figura A.49 mostra com maior destaque estes campos.

				
ρ	1000	20	10	50
PFE	0	5	30	10
	<input checked="" type="checkbox"/> Editar PFE			

Figura A.49: Campos destinados à declaração das propriedades físicas do *modelo geoeletrico*.

4º) Salvar o modelo

O *modelo geoeétrico* pode ser salvo de duas maneiras: através do item *salvar modelo* do menu (Figura A.50) ou quando o usuário desejar fechar a janela “Modelo” e não salvou as alterações (Figura A.51);

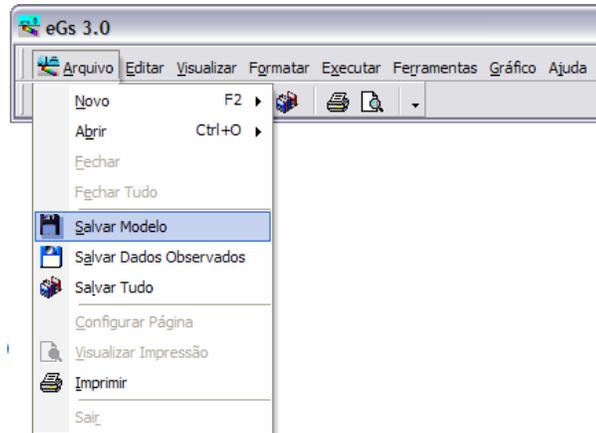


Figura A.50: Opção *Salvar Modelo* do Menu

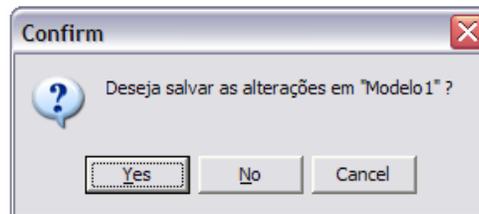


Figura A.51: Janela de aviso quando um *modelo geoeétrico* está sendo fechado, mas as alterações não foram salvas.

5º) Construir os *dados observados*.

Depois de feita a escolha do tipo de problema, há duas maneiras de editar os dados observados:

- i. A partir de dados observados novos;
- ii. A partir de dados observados salvo anteriormente.

i) A partir de dados observados novos:

Para abrir uma janela “Dados Observados” nova, primeiramente deve ser escolhido o tipo de problema na janela “Métodos”, depois basta seguir um dos dois caminhos:

- Janela Seleção→ Dados Observados → Novo (Figura A.52);
- Menu→Arquivo→Novo→ Dados Observados (Figura A.53).

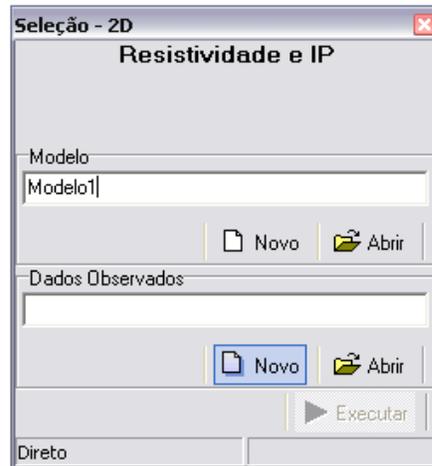


Figura A.52: Janela “Seleção” para abrir a janela “Dados Observados” nova.

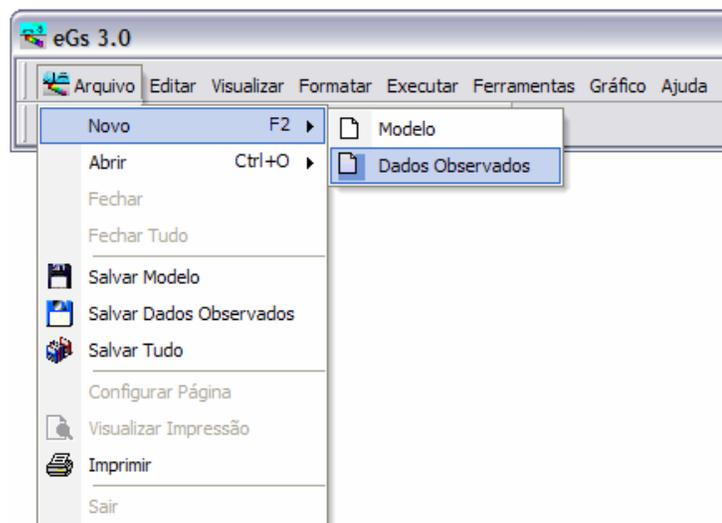


Figura A.53: Abrir uma janela “Dados Observados” nova através do Menu.

Surgirá a janela “Dados Observados” mostrada na Figura A.54.

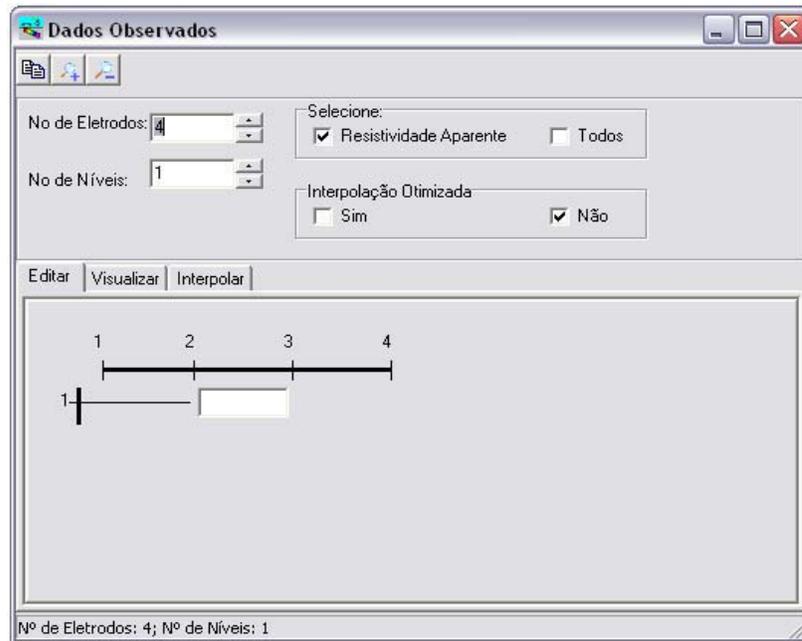


Figura A.54: Nova janela “Dados Observados” para início da edição.

A janela da Figura A.55 mostra a edição das observações de campo com 5 níveis e 16 eletrodos.

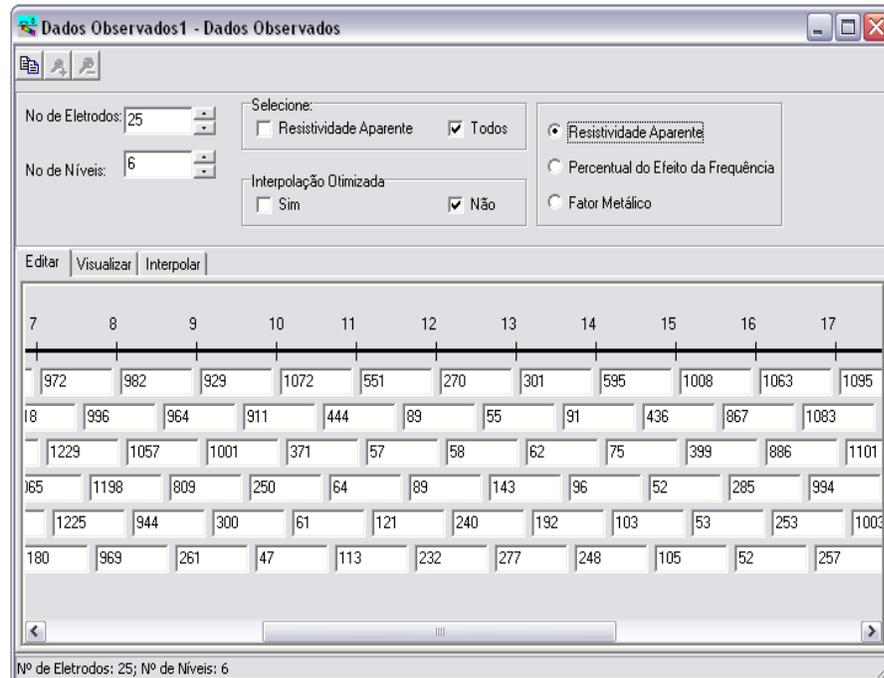


Figura A.55: Nova janela “Dados Observados” editada.

Os números de eletrodos e de níveis são alterados nas caixas de texto *No de Eletrodos* e *No de Níveis*, os quais devem obedecer à seguinte condição: $\text{No de Eletrodos} \geq \text{No de Níveis} + 3$. Caso o usuário não respeite a condição, o programa corrige acrescentando (ou diminuindo) eletrodos (ou níveis), de modo que sempre a condição seja satisfeita.

Caso a opção *Selecione* esteja marcada em *todos* (Figura A.56), indica que o usuário deseja editar ou visualizar resistividade aparente, PFE e fator metálico, aparecendo os itens mostrados na Figura A.57 para escolher qual a propriedade o usuário quer editar.

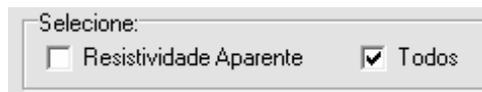


Figura A.56: Escolha para edição da resistividade aparente, percentual do efeito da frequência e fator metálico ou apenas da resistividade aparente.

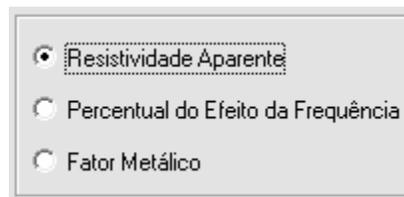


Figura A.57: Escolha do tipo de propriedade que o usuário deseja editar.

Os *dados observados* podem ser salvos de duas maneiras: através do item *Salvar Dados Observados* do menu (Figura A.58) ou quando o usuário desejar fechar a janela “Dados Observados” e não salvou as alterações (Figura A.59);

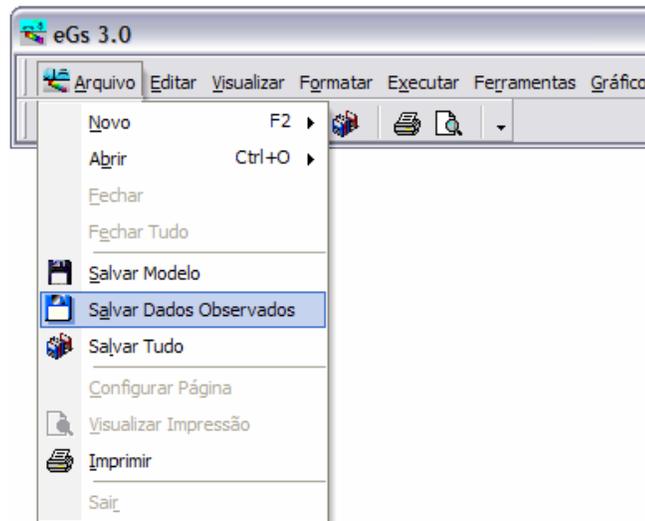


Figura A.58: Opção *Salvar Dados Observados* do Menu

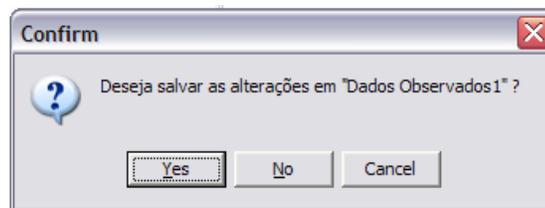


Figura A.59: Janela de aviso quando os *dados observados* estão sendo fechados, mas as alterações não foram salvas.

ii) A partir de dados observados salvos anteriormente

Para abrir uma janela “Dados Observados” com informações editadas anteriormente, primeiramente deve ser escolhido o tipo de problema na janela “Métodos”, depois basta seguir um dos dois caminhos:

- Janela Seleção → Dados Observados → Abrir (Figura A.60);
- Menu → Arquivo → Abrir → Dados Observados (Figura A.61).

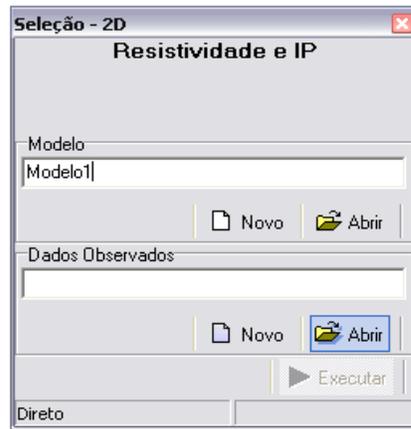


Figura A.60: Janela “Seleção” para abrir *dados observados* salvos.

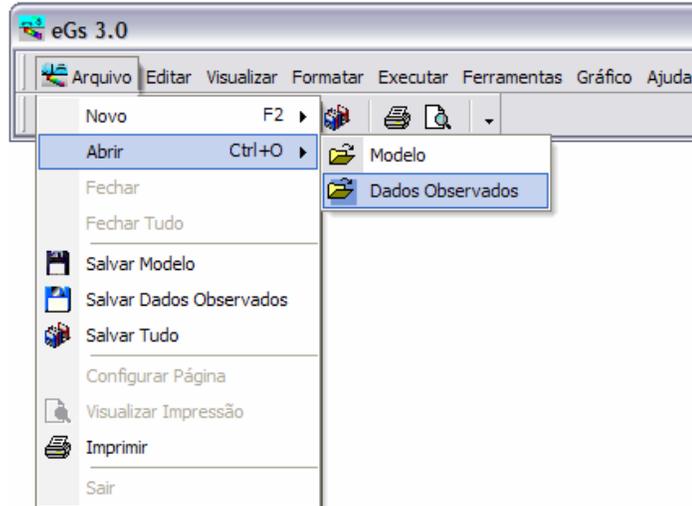


Figura A.61: Abrir os *dados observados* salvos através do Menu.

Surgirá a janela “Abrir Dados Observados”, mostrada na Figura A.62. Deve-se procurar arquivos com a extensão *.OBS. Em seguida aparece a janela para reedição, mostrada na Figura A.63.

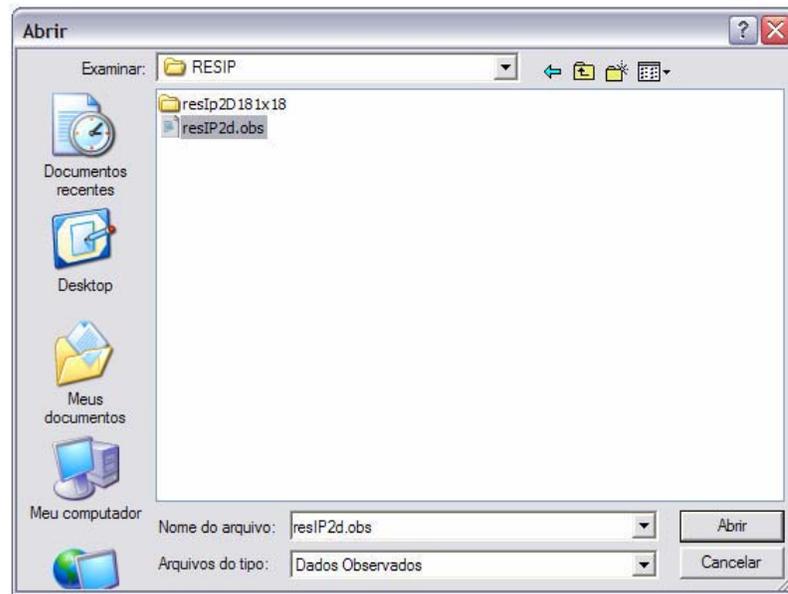


Figura A.62: Janela para abertura de arquivos.

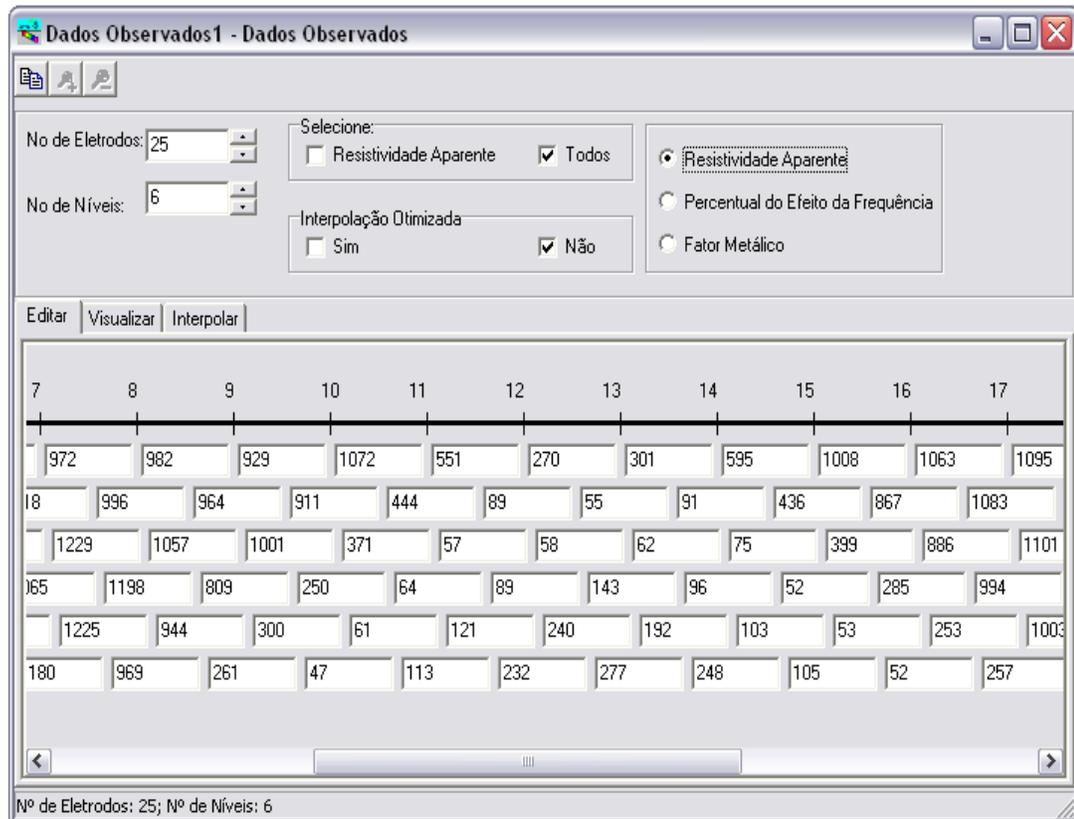


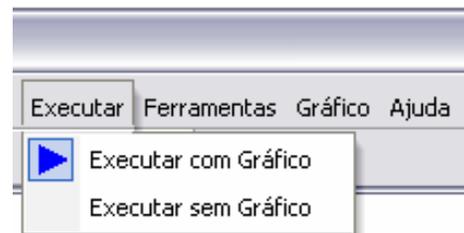
Figura A.63: Janela “Dados Observados” aberta para reedição de um arquivo salvo anteriormente.

6º) Executar a modelagem numérica:

Na eGs 3.0 a modelagem numérica só pode ser executada após o *modelo geológico* e os *dados observados* estarem editados. Feito isso, o botão “Executar”, que pode ser encontrado na janela “Seleção” (Figura A.64(a)) ou no item *Executar* do Menu (Figura A.64(b)), pode ser ativado e com apenas um clique a modelagem é obtida, que pode ser com a geração de figuras para interpretação (Figura A.65) ou apenas com o armazenamento dos dados de saída, dependendo da escolha feita pelo usuário nos subitens do Menu *Executar* (Figura A.64(b)).



(a)



(b)

Figura A.64:(a) Janela “Seleção” com o botão *Executar* ativado. (b) Itens *Executar Com Gráfico* e *Executar Sem Gráfico*.

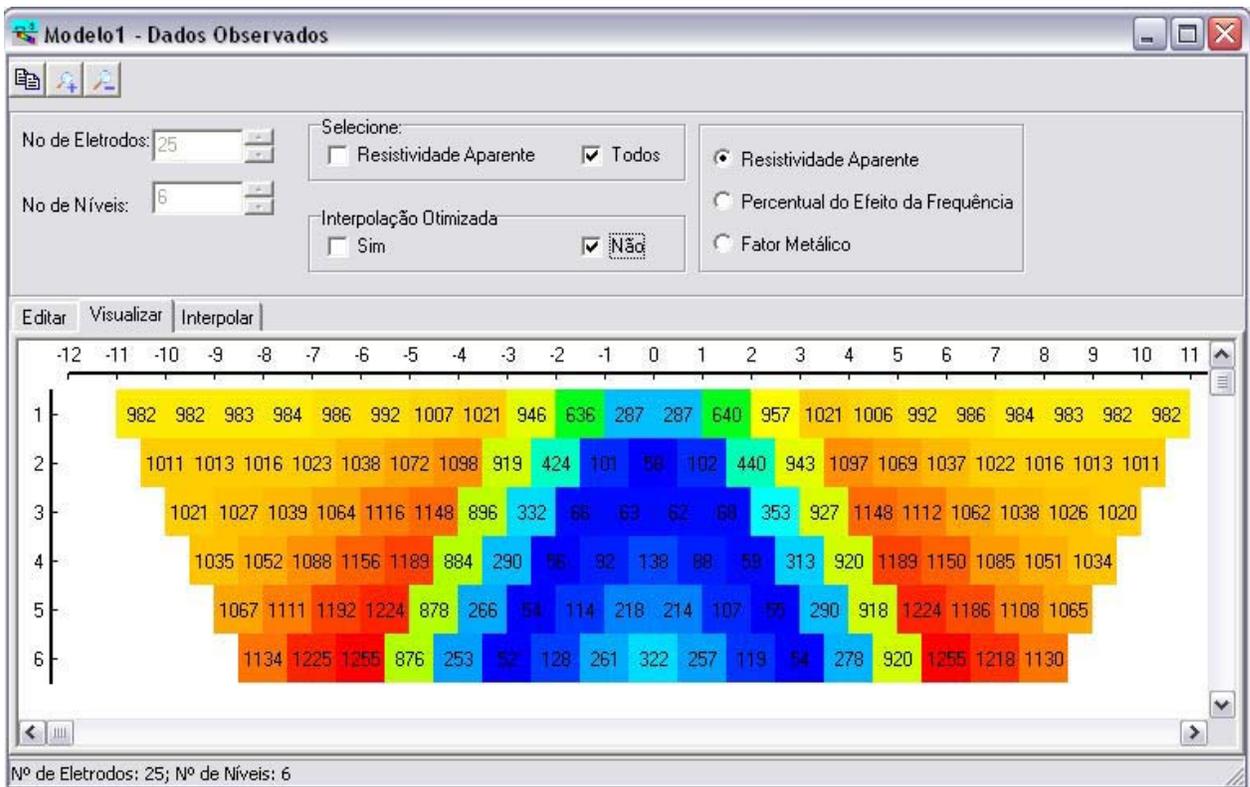


Figura A.65: Resultado da modelagem numérica com o item *Executar Com Gráfico* ativado.

Os arquivos do resultado do modelo *geolétrico* e *dados observados* são salvos em uma pasta com o mesmo nome dado ao arquivo com extensão *.MGE. Por exemplo, se o *modelo geolétrico* foi salvo em c:\eGs\Modelos\2D\ResIP\Modelo1.MGE, então os dados de saída serão armazenados em uma pasta com o nome Modelo1. Caso o usuário não queira salvar os arquivos *.MGE e *.OBS, os quais representam, respectivamente, o *modelo geolétrico* e os *dados observados*, o aplicativo armazena temporariamente os arquivos de saída e ao fechar a eGs 3.0 esses arquivos são apagados do sistema.

7º) Interpretação dos dados (pseudo-seções de IP e resistividade)

Os dados de resposta do modelo são visualizados na mesma janela usada para edição e visualização dos dados observados. A Figura A.66 mostra estes dados.

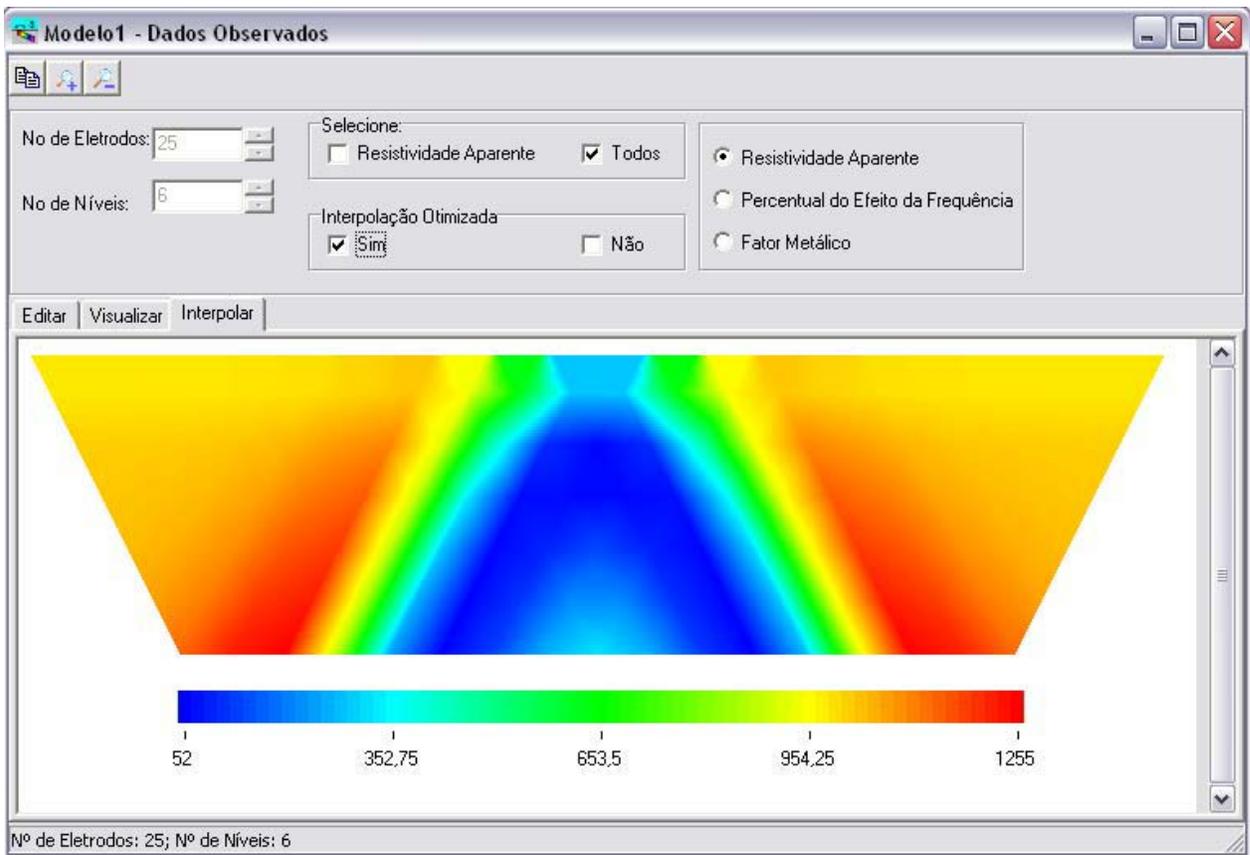


Figura A.66: Resposta do *modelo geoeletrico* do método IP e resistividade.

Em *Visualizar* e *Interpolar*, o usuário tem as seguintes opções de manipulação das figuras que representam as pseudo-seções:



→ Diminui as pseudo-seções nos tópicos *Visualizar* da janela;



→ Aumenta as pseudo-seções nos tópicos *Visualizar* da janela;



→ Copia as pseudo-seções nos tópicos *Visualizar* ou *Interpolar* da janela para a área de transferência.

A seção *Interpolar* tem a finalidade de mostrar ao usuário uma visualização mais suave do contraste das propriedades, como mostra a Figura A.66. Caso o usuário queira ver uma figura com melhor resolução das propriedades, basta escolher “sim” na opção *Interpolação Otimizada* (Figura A.67).

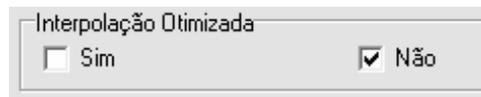


Figura A.67: Opção para interpolação das pseudo-seções de Resistividade e IP (deve estar marcada a opção “sim”)