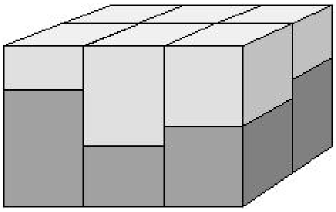
GRABLOX

Gravity interpretation and modeling software based on a 3-D block model

Version 1.5

User's guide

Markku Pirttijärvi



Report: Q16.2/2004/2

Geological Survey of Finland, Espoo unit
PO Box 96, 00250 Espoo

1. Introduction

The measurements of the gravity field provide a powerful method to study the internal structure of the Earth. In geophysical studies the petrophysical parameter that affects the gravity field is the density, ρ, which is equal to the mass (weight) per volume [kg/m^3]. Absolute gravity measurements aim to measure Earth's (vertical) gravitational acceleration, $g \approx 9.80665$ m/s^2 directly. Relative measurements that are usually made in geophysics aim to measure the difference in g with respect to some reference point. The accuracy of geophysical gravity measurements is about 10^{-6} m/s^2 = 0.1 mGal.

The distribution of the density variations inside the earth, or densities, positions, orientations and dimensions of isolated targets or geological formations define the magnitude and the spatial characteristics of the gravity anomaly. Interpretation of the gravity measurements made on Earth's surface is an inverse problem. The objective is to construct such a density distribution that would explain the measurements. Unfortunately, gravity data cannot be interpreted uniquely. Therefore, special care must be taken when assessing the validity of the final interpretation model. At the end, the interpreted density model can be used in geological and structural interpretation.

The GRABLOX program computes the gravity field of a 3-D block model. Figure 1 shows the model - a large super block, which is divided into smaller brick-like elements the size of which can vary. Each minor block is assigned an individual density value. For more information about 3-D block models, please, see the documentation of the BLOXER program, which is used to visualize and to maintain the 3-D block models.

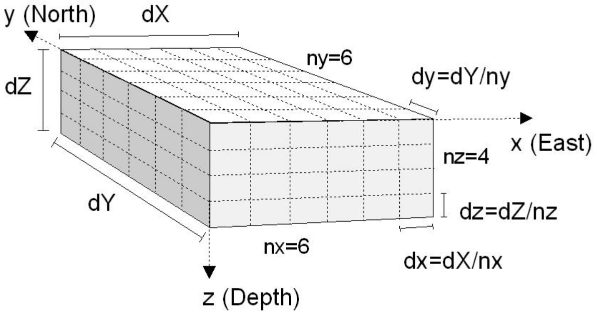


Figure 1. Block model of size $dX \cdot dY \cdot dZ$ divided into $nx \cdot ny \cdot nz$ minor-blocks of size $dx \cdot dy \cdot dz$. The model is aligned with the (geographical) coordinate system.

The GRABLOX program can be used both for forward and inverse modeling (inversion). The inversion method optimizes the density values of the individual blocks so that the difference between the measured and the computed gravity data get minimized. Another inversion method utilizes Occam's principle so that the roughness of the resulting model is minimized as well. The distribution of the density values inside the resulting block model can then be used in geological interpretation. In addition, the height of the blocks can be optimized both with and without Occam's method. Therefore the program can be used to interpret the thickness variations of the overburden layer (depth to the basement). Moreover, GRABLOX suits both for (large-scale) regional studies and (small-scale) local studies.

2. Table of contents

Keywords: Geophysics; Gravity, 3-D models; Interpretation; Modeling.

3. Installing the program

The distributed GRABLOX program requires a PC with Microsoft Windows 95-XP operating system and a graphics display capable for at least 1024×768 resolution. In forward modeling memory requirements and processor speed are not critical factors, since the program uses dynamic memory allocation and the forward computation is quite fast. However, the inversion of large block models requires lots of computer resources.

The program has a simple graphical user interface (GUI) that can be used to modify the parameter values, to handle file input and output, and to visualize the gravity data and the model. The user interface is based on the DISLIN graphics library. GRABLOX can be compiled and run also on certain Unix and Linux workstations, provided that appropriate Motif (window manager) and DISLIN libraries are installed and a supported Fortran90 compiler is available. The GRABLOC program is separate non-GUI version, which can be used to compute just the forward solution of a 3-D block model. Since the two programs are very similar this user's manual is applicable to the GRABLOC program as well.

The program requires two files:

GRABLOX.EXE	the executable file.
DISLIN.DLL	dynamic link library for the DISLIN graphics.

The distribution file (GRABLOX.ZIP) contains also a short description file (_README.TXT), this user's manual (GRABLOX_MANU.PDF), and an example data file (EXAMPLE.INP). To install the program, create a new directory and copy or unzip (with Pkzip/Winzip) the distribution files there. To be able to start the program from a shortcut that locates in a different directory one should move or copy the DISLIN.DLL file into the WINDOWS\SYSTEM folder or modify the PATH environment variable. When running the program over MS Network one needs to map the network drive (e.g., Explorer/Tools/Map network drive…) and provide it a logical drive letter.

4. Getting started

On startup the program will show a standard file selection dialog, which is used to provide the name of the input model file (*.INP). If the user cancels the file selection operation the GRABLOX.INP file will be used. If this file does not exist default parameters are used and the GRABLOX.INP file will be created automatically. The parameters of individual blocks are read from (and stored into) a separate *.BLX file. The block file is normally stored in a text format, but binary format can also be used (see BLOXER documentation). Before the user interface is built up, the program reads graph parameters from the GRABLOX.DIS file. Again, if this file does not exist default parameters are used and the file is created automatically.

The GRABLOX program runs in two windows: the console window (command shell) and the graphical user interface (GUI) window. The console window is used to provide input/output (I/O) functionality in certain tasks that have not been implemented in the GUI (for example, to provide the density values for each layer without BLOXER).

After the initial model and the system parameters have been defined the program builds up the user interface (see the Appendices). At this point the graph area will be blank because forward computation has not been made and measured data have not been read in. However, the model can be visualized using the *Layers* and *Sections* push buttons (Appendices A and B). Repetitive use of *Layers* and *Sections* push buttons will show successive layers or sections. The *Crossing dir* button swaps between (West-East directed) X-section and (South-North directed) Y-section views. The <-/-> button is used to reverse the direction of layer and section swapping.

The *Compute* push button on top of the rightmost control pane of the application window is used to perform a forward computation. The *Contours* button is used to plot the data as a contour map along with a 3-D view of the model and a short description on some auxiliary parameters (Appendix C). The *Profiles* push button can be used to display a single profile across the model area and the *Crossing dir* button swaps between SN and WE directed profiles (Appendix D).

Model and system parameters can be changed using the text fields in the left control pane. The *Update* button must be pressed to validate the changes. The computational options can be changed using the items in the *Gravity* menu. Before measured data have not been read in most of the controls in the right control pane are inactive (grayed). To learn more about data interpretation, please, see chapter 6.

The rest of this chapter defines the purpose of the various program components.

4.1 File menu

As shown in Appendices, the application window has four menus. The *File* menu contains following items:

Open model	open an existing model file.
Read data	read in data for interpretation or comparison.
Read regional	read in regional data (stationary field).
Save model	save the model into a file.
Save data	save the data (computed & measured) into a file.
Save results	save the results (description & data) into a file.
Read disp	read in new graph parameters from a *.DIS file.
Save graph as PS	save the graph in Adobe's Postscript format.
Save graph as EPS	save the graph in Adobe's Encapsulated Postscript format.
Save graph as PDF	save the graph in Adobe's (Acrobat) PDF format.
Save graph as WMF	save the graph in Windows metafile format.
Save graph as GIF	save the graph in GIF (graphics interchange format) format.

These menu options will bring up a standard (Windows) file selection dialog that can be used to provide the name for the file for open/save operation. Model files (*.INP) files are saved in text format and the parameters of the minor blocks are stored in a separate *.BLX file, with the same prefix name as the *.INP file. When saving the results, GRABLOX stores text information into an *.OUT file, the data into a *.GBM file, and makes a header file (*.HEG) that describes the format of the data file. Measured data are read form a column formatted data (*.DAT) files, which are in (ASCII/ISO8859) text format. The *Save data* entry generates a *.DAT file in the same format. See chapter 7 for more information about the file formats.

GRABLOX does not support direct printing to a printer. The graphs are saved into files in formats that can be printed using a suitable third party programs (e.g., Ghostview, and PaintShopPro, Adobe Acrobat Reader, MS Word, etc.). All graphs are saved in landscape A4 size as they appear on the screen. The GIF format is the only option to generate a bitmap image (size 2970×2100 pixels). The EPS file does not include preview bitmap.

4.2 Gravity menu

The *Gravity* menu contains following items:

Comp → Meas	replace the measured data with the computed response.
Comp → Regi	replace the regional field with the computed response.
Subtract regional	remove the regional field from measured data.
Remove measured	remove all information about measured data.
Compute default	build up a default model based on measured data coordinates.
Scale unit	define dimensions in meters, kilometers or miles.
Gravity field	swap between total field and anomalous response.
Computation method	choose the computation method: blocks, points, vert.derivative
Background method	set the background: given value, layer mean, whole model mean
Depth weighting	disable/enable partially or enable totally
Enable bulkiness	enable/disable rounding of parameter values in inversion.
Enable roughness	enable/disable use of the roughness parameter in inversion.
Enable inversion	enable/disable the inversion (allocate Jacobian).

- The *Comp→Meas* and *Remove measured* items are useful when using the program for testing and educational purposes.
- The *Comp→Regi* item can be used to define the regional field using the current model. Alternatively one can use the *Read Reg. Field* menu item to read in regional field data.
- The *Subtract regional* item can be used to remove the current regional field (if it has been read in or defined otherwise) and/or base anomaly level from the measured data. Removing the regional field reveals small features of the data. By saving the computed data into a file (*Save data* in *File* menu) one can define a new data where the regional gravity effect has been removed.
- The *Compute default* item is useful after new measured data has been read in, because it automatically gives the super-block some reasonable coordinates inside the data area. If the data is regularly gridded GRABLOX also discretizes the model so that the blocks are put below the data points (z coordinates also). Note that 2-D surface data and profile data are handled differently. When dealing with irregular data the user should provide an appropriate position, size and discretization for the super-block with the aid of the BLOXER program.
- When GRABLOX reads in data it does not know the units of the coordinates and distances. The *Scale unit* item defines the actual volume of the block model, and hence, the correct gravity values. Therefore, the user should supply correct scale (meters, kilometers, or miles) before continuing with the modeling or interpretation. The information about the units will be stored in the model file and used in output files.

- If regional field (G_r) has been defined (read from a file or set using the *Comp→Regi* item) the *Gravity field* item can be used to change the computed gravity (G_c) field between the anomalous field (regional field excluded), and the total field (regional field included). Note that the base response (G_b) is always added to the computed response.

- The *Computation method* item defines the three modes of the gravity computation:

  1. The gravity effect is computed using actual prism-like minor blocks.
  2. The gravity effect is computed using point sources, i.e., spherical sources the volumes of which are equal to the volumes of the minor blocks. This method allows faster computation, but is less accurate than the block-model approach.
  3. The vertical gradient of the gravity effect is computed using prism-like minor blocks.

Note that point source method should used only if equal sized blocks are used and the model is finely discretized (more than million blocks, or so). The point source approach is about 18 times faster than block model computation, but because the accuracy of this method is difficult to assess, it should be used only to provide coarse first order approximation to the gravity effect. When using point sources the model and computation grid should be set evenly so that the blocks are located between computation points. In future GRABLOX versions deeply situated blocks may be approximated with point sources to provide faster computation.

- The *Background method* item defines the three modes of the background density:

  1. Normally the background density value defined by the *Bg dens* text field in the left control pane. In this case the background density value is subtracted from the density values of each minor block.
  2. In large-scale crustal studies the background density value is defined separately for each depth using the mean density value of the blocks in that layer. The mean density of each layer is then subtracted from the density of each minor block. This method is useful in large-scale crustal models where the density is increasing downwards.
  3. Sometimes it may be useful to define the background density value as the mean value of all the minor blocks of the model. Thus, the mean density of the whole block model is subtracted from the density values of each minor block. This method allows simple way to define floating background value. If the density increases downwards this method is not as good as the method 2.

The aim of the background density value is to remove the gravity effect of the sides of the super-block when the model contains absolute density values. If background density is not used (or the contrast is large) the super-block shows up as a single source, which creates a very strong gravity anomaly, and the effect of the internal density variations gets hidden. Usually the background density can be zero and the inversion resolves the residual density, i.e., the difference from the background value. When absolute density values, for example, from petrophysical samples are available the given background value can be equal to the mean of the sample values.

The last three items in the *Gravity* menu deal with the inversion:

- The *Bulkiness* restricts the density values so that the density values are rounded to the nearest step value. See chapter 6.8 for more information.

- The *Roughness* is defines discontinuity information in the gravity inversion when Occam method is used. The roughness information must be stored in to the model file as a second parameter in the BLOXER program. If the file does not contain roughness values, this menu item will be inactive. See chapter 6.9 for more information
- The *Enable inversion* item defines whether or not the Jacobian will be computed. When measured data has been read in, the inversion is enabled by default. However, if the model is very large the program may be unable to allocate memory for the large sensitivity matrix (Jacobian). In this case GRABLOX disables the inversion and computes only the forward solution. This menu item can be used to enable the inversion again.

4.3 Edit menu

The *Edit* menu contains following items:

Block type 1/2	swap between equal-sized and increasing height block modes.
Min/Max values	redefine the min & max parameter value on the color scale.
Layer-wise fix/free	set the fix/free/hidden status of the blocks per each layer.
Layer-wise reset	set the density value of the blocks per each layer.
Add margins	adds marginal areas around the super-block (layer inversion).
Del margins	removes the marginal areas around the super-block.
Labels	hides or defines the information shown as parameter labels.
Show/Hide grid	show or hide the block outlines in 2-D graphs.
Contour/Image	show the 2-D gravity data as contour maps or image maps.
Show/Hide regional	show or hide regional field or base response from graphs.
Color scale	change the color scale used in the graphs.
Depth zoom	enhance the vertical scale of the super-block in graphs.

- Although the *xyz* dimensions of the blocks can vary, all blocks must have equal size by default. In the alternative block type (*Block type 2*) the block height is increasing with depth by the value of the topmost block at every successive depth (e.g. 0.5, 1, 1.5, 2 km). This method reduces the resolution at the bottom and decreases the total amount of blocks.
- The *Min/Max values* item changes the limiting parameter values on the color scale. It also determines the absolute minimum and maximum parameter values used in the inversion. These values, in turn, determine the minimum and maximum parameter steps and the stepping used in the bulky inversion. When using this menu item the programs shows (on the console window) the minimum and maximum density values of the current model and asks new values for the parameter scale.
- *Layer-wise fix/free* option allows setting the fix/free status (model weights) of the blocks for each layer manually. The new fix/free values are given on the console window. See chapter 6 for more information about the fix/free status.
- *Layer-wise reset* option allows setting the density value of the blocks for each layer manually. The new density values are given on the console window. See chapter 6 for more information about the fix/free status.
- The *Labels* and *Show/Hide grid* items can be used to clarify the contour/image maps as well as the layer and section graphs. The *Labels* option either removes the labels totally, or shows the density, block height, block depth, fix/free (weights), or roughness values (if available).

- The *Show/Hide regional* item is useful when the difference between the measured (or computed) data and the regional field (or base level) is so large that the fit between measured and computed data is difficult to see in profile graphs.
- *Margins* are important when gravity data is interpreted using a layered model, i.e, when optimizing the height of the blocks defining the overburden sedimentary layer, for example. The margins extend the effect of the topmost layer away from the computation area, and thus, reduce the boundary effect of the super-block. When adding margins the program asks (on the console window) for the width of the margins.
- The preparation of contour plots can be time consuming if there are lots of data points. Moreover, the interpolation can easily create artifacts that arise from irregularly spaced data points. Therefore, the data can be plotted also as an image map.
- The *Color scale* swaps between normal rainbow scale, inverted rainbow scale, normal grayscale, inverted grayscale and a special (seismic/temperature) red-white scale.
- The depth of the X-sections and Y-sections are usually plotted in 1:1 scale with the horizontal dimension. The *Depth zoom* list widget increases the length of the depth axis. It is useful if the height of the block model is considerably smaller than its horizontal dimensions.

The *Exit* menu has only one item. On exit the user is given an opportunity to save the current model and the results, provided that the exit is made without an error condition. If applied, the program asks first the name for the model file (*.INP + *.BLX) and then the filename for the results (*.OUT + *.HEG + *.GBM). If the user cancels the save operation at this point, default filenames GRABLOX.* will be used. Errors that are encountered before the GUI starts up are reported in the GRABLOX.ERR file. When operating in GUI mode, run-time errors arising from illegal parameter values are displayed on the screen.

The new GRABLOX version (1.5) stores the messages that appear on the console window into the GRABLOX.LOG file. Please, not that the log file will be written over when the program is restarted.

5. Using the controls

The model and computation parameters are changed using the controls on the left side of the GUI window of the GRABLOX program. The following chapter discusses the meaning of the various control widgets.

5.1 Left control pane

The *Update* button is used (and must be used) to validate changes made to the model parameters. After pressing the *Update* button, the values of the various text fields defining the position, size, and discretization of the block model are read in, checked for erroneous values, and the old values are replaced. Be warned that currently the use of the *Update* button also invalidates previous results and new computation is needed.

The list widget below the *Update* button defines how the block parameters will be affected when the *Update* button is pressed.

1. The *Shift only* mode updates only the *xyz* position of the block model. The size and the discretization of the super-block will not change if *Shift only* mode is active. This is the default mode and it is purpose is to protect the model from accidental changes.
2. The *Ignore* mode allows one to redefine also the size and discretization of the super-block. However, *Ignore* mode does not affect the density values of existing minor-blocks. Therefore, the internal distribution of model parameters can be severely distorted when the discretization is changed.
3. The *Preserve* mode uses 3-D nearest neighbor interpolation method to determine new parameter values when the block is resized, rediscretized or repositioned. For example, if the position of the super-block is shifted, the position of an anomalous density remains at the same *xyz* position, but its relative position inside the super-block will change. Note that 3-D interpolation of densely discretized models can be very time consuming.

The following text fields define the position, size and discretization of the super block:

1. *X* position (easting) of the SW (bottom-left) corner of the super block.
2. *Y* position (northing) of the SW (bottom-left) corner of the super block.
3. *Z* position (depth to the top) of the super block.
4. *X* size of the super block (in EW direction).
5. *Y* size of the super block (in NS direction).
6. *Z* size of the super block (in vertical direction).
7. *X* discretization (number of blocks) in EW-direction.
8. *Y* discretization (number of blocks) in NS-direction.
9. *Z* discretization (number of blocks) in vertical direction.

Note that in GRABLOX the true dimensions of the block model (and the computation grid) are defined using the *Gravity*/*Scale unit* menu item.

The next four text fields define some basic density parameters:

1. The background density value (g/cm^3).
2. User defined base anomaly level (mGal).
3. The gradient of the base anomaly in *x* (EW) direction (mGal/100 dist. units).
4. The gradient of the base anomaly in *y* (NS) direction (mGal/100 dist. units).

The background density value is subtracted from the density values of each minor block if the *Given value* mode is active in the *Gravity*/*Background method* menu item. Note that the background density can have a zero or even a negative value.

The base level (B_0) and its *x* and *y* gradients (B_x, B_y) are used to define a planar regional base response (G_b) which is added into the computed response. Positive B_x and B_y values mean that the base level increases towards east and north, respectively. Note that in the left control pane the gradient values are defined per 100 distance units. The base response is attached to (distances are measured from) the southwestern (SW) corner of the super-block. Thus, the base response can be defined as $G_b= B_0+(x-x_0)* B_0+(y-y_0)*B_y$.

Important:

The density is defined in units [g/cm^3] (grams per cubic centimeter). Use the BLOXER program transform to transform the units (g/cm^3 ↔ kg/m^3) if needed.

The gravity field (computed, measured, regional and base) is defined in units [mGal] (milliGals), which are derived as 1 mGal = 10^{-5} m/s^2 (note 10^{-6} m/s^2 = 0.1 mGal).

The *Reset param* button is used to reset the density value of the whole super block to the density value defined by the *Param* text field below the *Reset param* button.

The remaining six text fields at the bottom of the leftmost control pane define the computation grid.

1. The spatial *x* step between the grid points.
2. The spatial *y* step between the grid points.
3. The *x* coordinate (easting) of the grid start position.
4. The *y* coordinate (northing) of the grid start position.
5. The *x* coordinate (easting) of the grid end position.
6. The *y* coordinate (northing) of the grid end position.

Note also that the measured data are assumed to be located irregularly; the computations are always made at the actual measurement locations. Therefore the start and end positions of the computation grid are inactive when data has been read in. The *x* and *y* steps, however, have important meaning in contour maps and image maps. When irregularly space data has been read in, the *x* and *y* steps should be set manually to reduce possible artifacts in the contouring or the size of the data rectangles in image maps.

Important:

Although the computation method uses the topography values (z coordinates or height) of the measured data, the forward computation of synthetic models assumes that the computations are made on a constant height level which is equal to the top of the super-block.

5.2 Right control pane

The *Compute* button is used to perform the forward computation using the current values of the model and system parameters.

The *Optimize* push button is used to start the inversion. Note that depending on the total number of minor blocks and the amount of computation points, the inversion can take a considerable amount of time.

The list widget below the *Optimize* button defines the six primary inversion types:

1. *Base* option optimizes the values of the base level and its *x* and *y* gradients.

2. *Density* option optimizes the density of the blocks of the whole model.
3. *Occam d* option optimizes the density of the blocks using Occam's method, where the data error (difference between the measured and the computed data) and the model roughness are minimized together.
4. *Heights* option optimizes the height of the blocks. The height optimization affects the depth extent, i.e, the depth to the bottom of the blocks. The height of the blocks of the bottom layer are never optimized.
5. *Occam h* option optimizes the height of all the free blocks using Occam's method.
6. *Occam h+d* optimizes the height of all the blocks together with the mean density of each layer of the block model.

See chapter 6 for further information about the inversion options.

The second list widget defines additional constraining method in *Density*, *Occam d* and *Height* inversion options.

1. *Normal* method does not give any additional constraints for the parameter values.
2. *Smooth vertical* mode constrain the vertical parameter variation. The resulting model should therefore have gentle density variation along depth axis.
3. *Smooth horizontal* mode constrains the horizontal parameter variation. The resulting model should therefore have gentle density (or height) variation in horizontal plane.
4. *Smooth 3-D* mode applies the two abovementioned options and tries to generate a smooth 3-D model.

The SVD based *Density* and *Height* inversion are so-called unconstrained inversion methods. The abovementioned options provide a poor-man's constraining by limiting the new parameter value based on the mean of the neighboring blocks. The allowed parameter variation is defined by the *Step %* text field discussed below. Since the *Height* inversion only utilizes horizontal smoothing the vertical smoothing does not have any effect and 3-D smoothing is equal to horizontal smoothing. Note also that in *Occam d* optimization the abovementioned options redefine the method used to compute the local roughness. See chapter 6 for further information about the different inversion options.

The third list widget defines the combination of base response components (B_0, B_x, B_y) that is going to be optimized when *Base* optimization method is made. The options are: none, base level, *x* gradient, *y* gradient, *x* and *y* gradient, and all three components. Note that in *Height, Occam h* and *Occam h+d* optimizations, the base response can be optimized together with block height values. Therefore, the none-option should be used to exclude the base response from the optimization if needed. Note that in practice combined inversion of base response parameters and block heights requires that some additional *a priori* information (e.g., well data) is available. For the same reason base response cannot be optimized together with density values. See chapter 6 for further information.

The following five text fields define some inversion parameters.

1. The *Iters* field defines the number of iterations made during one inversion run.
2. The *Thres %* value defines the strength of damping in *Density* and *Height* optimizations. Normally the threshold should be at the level of the smallest singular values that are still "informative". The smaller the threshold value is the weaker is the

damping. Note that the value given for the threshold is divided by 100 (%) so the default value 0.01 is actually 0.0001.

3. The *Step %* value defines the parameter step used to obtain the partial derivatives (forward difference) for the Jacobian when block heights are optimized. It also defines the parameter stepping used in bulky inversion and the maximum parameter change in the additional constraining methods. The step values are defined as a percentage from the min & max parameter values. For example, if the step value is 2.5 (default) and the min & max density values are 0 and 2 (g/cm^3), the parameter stepping is dp= 0.05 (g/cm3) (=(2-0)*2.5/100).
4. The *F-option* parameter is used to pass additional (real valued) numerical value to the inversion. Its role depends on the selected inversion type.
5. The *I-option* parameter is used to pass additional (integer valued) numerical value to the inversion. Its role depends on the selected inversion type.

The next two text fields are used also in basic forward modeling and visualization.

6. The *Trim* value defines a limiting residual density value. If the absolute value of the difference between the background density value (which depends on the *Gravity/ Background method* menu item) and the density of a minor block is less than the trimming value then the effect of that block is not computed at all. This option concerns the forward computation only.
7. The *Layr/Sexn* text field defines the index number of the current layer or section when the model is being visualized (see the meaning of the buttons below).

The following six buttons are used to change the contents of the graph area.

1. The *Contours* button is used to show the contour plot (or image plot) of the gravity data. If measured data has been read in a repetitive use of the *Contours* button swaps the view between the computed, measured, and regional/base response data, and the data error (difference between measured and computed data).
2. The *Layers* button is used to show the density values on a horizontal layer.
3. The *Sections* button is used to show the density values on a vertical cross-section.
4. The *Profiles* button is used to show the curves of computed, regional /base response, and measured data.
5. The *Crossing dir*. button is used to change the direction of the sections views and data profiles between (vertical) SN and (horizontal) WE directions.
6. The <-/-> button is used to change the direction towards which the contours, layers, sections and profiles are being rotated when pressed multiple times. This button affects also the rotation direction of some other tasks, such as the selection of color scale.

Note that pressing the *Layers*, *Sections*, and *Profiles* buttons multiple times changes (rotates) the current layer, section or profile to the next (or previous) one. At the last (or first) layer, section or profile the view jumps to the first (or last) one. One can jump directly into a desired layer or section by first providing the index number on the *Layer/Sexn* text field and then pressing the corresponding push button.

Note that the profile data does not display the actual data, which are assumed to be irregularly spaced. Instead it shows interpolated data, where the "number of profiles" and "point step" is based on the interpolation density, i.e, the *X-stp* and *Y-stp* values at the bottom of the left control pane. If the data are highly irregular, the contouring algorithm produces bad artifacts.

In such a case, the *Contour/Image* item in the *Edit* menu should be used to change the data representation from a contour map to an image map.

The GRABLOX program can be used also for 2.5-D interpretation of single profile data. In this case the data are always read in so that the data profile is located along the x-axis. Consequently, the model needs to be discretized along the *x* direction only (*y* discretization can be equal to one). Moreover, contour and image maps and *Y*-sections cannot be displayed and the profile representation shows the actual data points from the single profile. Note, however, that 2.5-D inversion has not been tested thoroughly in version 1.5.

The three slide controls (*Horiz. rotation, Vert. rotation and Rel. distance*) below the graph window are used to change the point of view for the 3-D model view. The view can be rotated horizontally and vertically, and the relative distance of the point of view can be changed (as if zooming in and out).

6. Interpretation guidelines

The GRABLOX program is intended for crude but practical interpretation of complex 3-D and 2.5-D structures. This chapter discusses the gravity interpretation in general and then defines the six main types of inversion used in GRABLOX, namely, base response optimization (*Base*), normal density and height inversion (*Density, Height*), Occam's inversion of density and height (*Occam d, Occam h*), and combined Occam inversion of height and the mean density of layers (*Occam h+d*). The additional constraining method, bulky inversion, and parameter roughness are also discussed along with basic procedures of 3-D and two-layer interpretation. But first few words about the regional field.

6.1 Regional field

Because we do not know the true density variation of the whole Earth we cannot compute absolute gravity values. Therefore we use a somewhat vague concept known as the regional field that incorporates the large-scale density variations (long wave-length anomalies). When investigating near surface targets the regional field can be removed from the measurements. Although the regional field level can be almost anything the important thing is that it should not include anything that might be explained by density variations within the volume of the super-block.

In GRABLOX the regional field can be defined in three different ways: 1) it can be a user-defined base response (base level and its *x* and *y* gradients), or 2) it can be read from the same file as the measured data (*Read data*), or 3) it can be read from a separate file (*Read regional*). Note that in the third case the *xyz* coordinates of regional data must be exactly the same as those of the measured data. The use of external files allows determination of the regional field more accurately than the planar base response can. In many cases, however, the base response is a good approximation to the regional level problem in gravity interpretation.

The external regional field, which has been read from a file, cannot be optimized. However, the combined use of the base response and external regional field provides a method to define a complex regional field the linear (first-order) part of which can be optimized. Note also that in *Height, Occam h and Occam h+d* inversions the regional field can be optimized together with the height values (and mean layer densities).

6.2 Fix/free status

The fix/free status of a block is as weighting factor that defines the relative importance of the given density or height value of the block. The fix/free value is zero (0) for a totally fixed block (not optimized at all), and the one hundred (100) for a totally free block. Otherwise, the (integer) fix/free values from 1 to 99 define variable weighting factor.

GRABLOX contains only a limited amount of model editing functions. Normally, the BLOXER program must be used to edit the fix/free status of the blocks. The only exception is the *Edit/Layer-wise fix/free* menu item that can be used to manually set a constant fix/free value for each layer.

6.3 Normal inversion

The normal inversion method used in GRABLOX is based on linearized inversion scheme similar to that defined in Jupp and Vozoff (1975) and Hohmann and Raiche (1988). The *Base*, *Density* and *Height* inversion methods use singular value decomposition (SVD) and an adaptive damping method that has been defined in Pirttijärvi (2003). In *Base* and *Density* inversions the partial derivatives of the sensitivity matrix, i.e., the Jacobian, are computed as a side product of the forward computation provided that the inversion is enabled and memory for the large matrix was successfully allocated. In *Height* inversion the partial derivatives are computed numerically as forward differences.

Note that the computation of the SVD is the most time-consuming part of the inversion having O^3 dependence on the total number of minor blocks. Therefore normal inversion of large models can take considerable amount of time.

The two main types of normal SVD based inversion are the *Density* and *Height* inversion methods defined by the first list widget in the right control pane. The base response optimization is a special case of normal inversion. These inversion methods and the associated concepts of additional constraining and parameter bulkiness are discussed in the following.

6.3.1 Density inversion

The *Density* inversion optimizes the individual density values of all free blocks. The density values are limited by the minimum and maximum values of the color scale shown below the *Layer* and *Section* graphs. These min/max values can be changed using the appropriate item in the *Edit* menu.

6.3.2 Height inversion

The *Height* inversion optimizes the height, or actually the depth extent (depth to the bottom of the block) of the blocks. When working with 3-D models that have multiple layers (nz > 3), the *Height* inversion is not very practical, although it can enhance some features in the model. The most practical use of *Height* inversion is to optimize the depth extent of a single layer. For example, the thickness variations of a sedimentary layer above a denser basement can be interpreted using a two-layered model where the densities of the layers are fixed. See chapter 6.4.2 for more information about two-layer inversion.

Note that the height optimization will resize the block below the optimized one even if it was fixed and the bottom layer is never optimized. The depth extent (h) has limiting values, $h_{min}= h-t_i/4$ and $h_{max}= h+t_{i+1}/4$, which are based on the height of the block itself (t_i) and the block beneath it (t_{i+1}). The height of a block can never become equal to zero.

6.3.3 Base response inversion

Optimization of the base level and its gradients in x and y directions (B_0, B_x, B_y) is a simple, but important part of the interpretation process, particularly if no other means are used to set the regional field. Once the data have been read in and an initial model has been defined (e.g., *Compute default*), one can try to optimize the base response to fit the large-scale difference between the measured data and the computed response. The third list widget in the right control pane defines the combination of the base response components that will be optimized (B_0, B_x, B_y, B_x and B_y, or all of them).

Note that the menu item *Comp -> Regi* can be used to cast the current computed data onto regional data. If the model is a so-called zero-model, that is, the block densities are equal to the background density or the super-block is temporarily located very deep, the computed data will be equal to the base response ($G_c= G_b$). By using the *Remove regional* menu item one can subtract the base response from the measured data. In most cases this will clarify the inspection of the contour and image maps of the gravity data and will bring up more details. Note that after removing the base response from measured data the base level and its gradients are reset. By saving the data into file and editing the column indices of the *.DAT file (see chapter 7) one can define a new file for the measured data from which the regional level has been removed. Bear in mind that this can lead to different inversion result because the data error, and hence the data weighting, will be different.

6.3.4 Additional constraining

The additional constraining method defined by the second list widget in the right control pane is used in SVD based *Density* and *Height* inversions to control the parameter values (and steps) based on their surrounding values. In principle, if the new parameter value differs greatly from the mean of the surrounding blocks then its values is cut down to that limit. The maximum allowed parameter difference is computed as a percentage of absolute parameter variation (*step**(max-min)/100). The additional constraining method used in GRABLOX is kind of a poor-man's Occam method.

6.3.5 Bulky inversion

The bulky inversion option is enabled and disabled using the corresponding item in the *Gravity* menu. This option can be used only in the SVD based *Density* inversion. The method uses the *Step %* value to classify (round) the parameter values and the new parameter values are rounded to the nearest class. The step used in this classification is computed as a percentage of absolute parameter variation (*step**(max-min)/100). The bulky inversion option is experimental and its efficacy is quite questionable.

6.4 Occam inversion

The famous citation from Wilhelm of Occam from the 14.th century states: "Non sunt multiplicanda entia praeter necessitatem", which translates "Plurality should not be assumed without necessity" or "One should not complicate matters beyond what is necessary". In geophysical inversion practice Occam's method means that in addition to minimizing the fit between the measured and computed data also the roughness of the model is minimized. Occam's method gives rise to models, which will not fit the data as well as SVD based inversion, but which are smooth because neighboring parameter values are used as constraints. The so-called Lagrange multiplier is used (manually) to define if the data error is given more weight than the model error, or vice versa. See chapter 6.4.3 for more information about the Lagrange multiplier.

What is important in Occam inversion is that if *a priori* data are available and fixed (not optimized) at some point, then Occam's method will constrain the surroundings to fit that point. For example, petrophysical data can be used to constrain the density at the surface. Similarly, well data can be used to define the depth to the basement at given point(s).

The ability to constrain the parameter values reveals another advantage of Occam's method. Even if the measured data are irregularly spaced or do not cover the rectangular area of the super-block, the Occam inversion can still create smooth and continuous models. In such situations the normal SVD based inversion (without additional constraining) gets unstable and tends to generate very rugged (chess-board-like) structures. This feature of Occam's method is advantageous especially in two-layer inversion (chapter 6.4.2).

The parameter roughness is computed as the difference of the density of a minor block from the mean of the density of the surrounding minor blocks ($r_i = p_i - \overline{p}_i$). In practice, the roughness (or its reciprocal, the smoothness) of the model is incorporated into the Jacobian by adding extra rows into it and the data error vector is continued with the model error vector. Since the number of data values is typically equal (or smaller) than the number of blocks in the top layer, the dimension of the linear system increases quite drastically. Because the SVD would be far too slow, Occam inversion uses conjugate gradient method to solve the matrix system for the parameter steps. The three different ways the Occam's method is used in the GRABLOX program are discussed in the following.

6.4.1 Occam inversion methods

The *Occam d* inversion optimizes the density of the individual blocks. As discussed above Occam's method will generate a smooth 3-D model and petrophysical data can be used (*a priori*) to constrain the density of the blocks at the surface. The *Occam d* inversion method uses the additional constraining method to enforce horizontal or vertical smoothness (the normal method is equal to the 3-D smoothing).

The *Occam h* inversion optimizes the height, or actually the depth extent of the blocks. The *Occam h* optimization will generate smoother layer boundaries than the normal *Height* inversion (even with additional constraining). Although height inversion can provide some enhancement in normal 3-D models that have many layers, the main use of height inversion is in layer inversion. See the next chapter for more information about layer inversion.

The *Occam h+d* inversion optimizes the height of the blocks together with the mean density of each free layer. This method is developed particularly for layer inversion and it does not require that the layers have constant density values. Instead the layers can contain density variations and only the mean density is being optimized. Note that layer can be not be optimized. See the next chapter for more information about layer inversion.

6.4.2 About 3-D interpretation

The SVD based unconstrained (*Density* and *Height*) inversion methods represent the old school of inversion. Without the additional constraining to control the parameter variation, the SVD based inversion, particularly the *Height* inversion, can give very rugged results. The biggest problem of the SVD based inversion is that it is too slow when the number of blocks is large. Therefore the Occam method together with the iterative conjugate gradient solver is preferred when the model is finely discretized. In the following few comments about practical 3-D interpretation are given. See also the next chapter to learn how to build the initial model based on the data.

One practical method to interpret 2-D gravity data is to focus from larger models into smaller ones. In crustal studies, for example, the coarsely discretized gravity data (e.g. 25×25 km grid) are interpreted first using large blocks (25×25 km) down to the depth of the Moho and below. This gives a large-scale global model. Then the volume of the local model is removed from the global volume (the blocks are made hidden using BLOXER). The depth extent of the local model should be such that it suits the interpretation of the local study area. If the local study area is 100×100 km the depth extent can be 25-30 km. The edited global model (with a cavity in it) is used to compute the regional field over the local study area. This regional field will incorporate the density variations around and below the local model.

The regional data and the (re-gridded) gravity data over the local study area should use a finer sampling of 5×5 km, for example. Note that the grid points should locate at the center of the blocks (2.5 km shift). If regional data are not based on a larger global model it can be derived using low-pass filtering of the gravity data. Alternatively, the regional tendencies of the data should be fit optimizing the base level and its gradients. Once the local data are available the local model with smaller block size of 5×5 km can be generated (*Compute default*). In the local model the number of horizontal layers (and horizontal discretization) should be defined so that the computation time would not become a critical factor for practical inversion. Preferably, the alternative block height method should be used so that block height increases downwards.

Petrophysical density data that has been gridded (using the PETROCK program) over the local study area beforehand can be imported into the local model using the BLOXER program (see BLOXER manual). The petrophysical data should define only the top layer of the initial model. The remaining parts of the model should be given constant density value equal to the mean density of the top layer or a typical background density value. In Finland, for example, the mean density that is normally used to compute the Bouguer reduction is 2670 kg/m^3. The same density value should be used for the background as well. Usually the densities of the top layer are kept fixed in the inversion. Sometimes, if the petrophysical data do not represent properly the underlying model the fix/free status can be reset or variable weights can be given (edited) to those blocks.

Once the gravity data, possible regional data and initial model have been defined one can start the inversion. Depending on the number of blocks and the number of data points the computation can take a considerable amount of time (see chapter 6.6). Multiple iterations can be performed in a row (over night) using the *Iters* text field in the right control pane. The *F-option*, Lagrange multiplier, can be used to enhance either data fit or model smoothness. Note also that the limit values of the color scale define absolute minimum and maximum values of the density values in the inversion.

The models resulting from block model interpretation, particularly from Occam inversion, are very smooth and special care must be taken when assessing the actual properties and density values of the geological model. Also, the user must always bear in mind that the interpretation of gravity data is not unique - an infinite number of possible models can fit the measured data equally well. To test the validity of the model, the density values and fix/free status can be edited manually using the BLOXER program and the differences in the fit of the various models can be used to assess the limiting values of density, depth and depth extent that are usually the most interesting quantities of the interpretation.

Due to the nature of gravity data one should bear in mind that the horizontal variation of the density distribution is easier to interpret, but the depth to the top and the depth extent of the targets are very difficult to interpret using a block model. For this purpose, the results of 3-D block model interpretation can be used as an initial model for the interpretation programs that use parametric models (e.g., Modelvision, Gravmag). The migration from the 3-D block model to parametric models will be implemented in future GRABLOX versions.

Finally, note that the GRABLOX program, and hence also the whole methodology of 3-D gravity inversion is still under development.

6.4.3 Two-layer interpretation

The three height inversion methods, *Height*, *Occam h*, and *Occam h+d*, can be used to estimate the thickness variation of a sedimentary layer or soils above the bedrock, if the density contrast between the two layers is sufficient. The procedure for the simplest form of layer inversion, the two-layer interpretation, is briefly described in the following.

First the user needs to read in the data. If the data are regularly gridded the *Gravity*/*Compute default* menu item can be used generate automatically a regular model where the blocks are positioned below the data points. If the data are irregularly spaced, the *Compute default* method might not work and the user must provide the horizontal discretization manually. Vertically the super-block is divided into two layers. Note that to be able to change the discretization the *Ignore* mode (list widget in the left control pane) should be active when updating the model. Note also that if the data are regularly gridded and contain z-coordinates the *Compute default* option adjusts the top topography of the model correctly. However, if the data are irregularly spaced and contain z-coordinates, then one should use the BLOXER program to interpolate the top surface. Alternatively, any third party program can be used to interpolate the data (and *z* coordinates) onto a regular grid, which is then edited into GRABLOX's data file format (see chapter 7 for file formats). If the data does not contain *z* coordinates the data and the top of the model is positioned at height *z*=0. The vertical height of the super-block should be such that the bottom does not hinder the interpretation (resize if needed). If well data is available the overburden thickness can be fixed at those points using the BLOXER program.

Once the position and discretization are set the user needs to add margins around the super-block (*Edit/Add margins* item). The purpose of the margins is to reduce the effects of the borders of the super-block where happens an abrupt density change. Naturally, if the basement outcrops (comes to the surface) around and inside the model, then margins are not needed (e.g., a syncline or a lake). The density of the basement layer is set equal to the background density. The overburden layer is given an initial value which is either larger of smaller than the basement density depending on the problem. Layer densities can be set using the *Edit/Layer-wise reset* menu item. The same background value should be used for the topography corrections if they are made.

The *Occam h+d* inversion can be used for simultaneous optimization of the layer thickness and mean density. In this case, however, *a priori* data from drill-holes are needed to reduce the non-uniqueness from the inverse problem. Alternatively, the basement must be outcropping (come to the surface) somewhere inside the model area.

In *Height*, *Occam h*, and *Occam h+d* inversions the base level and its gradients (or a combination of them) can be optimized together with the height values (and mean density). One should bear in mind that the gravity effect of a thin infinite horizontal slab or plate is constant and thus represents change in the base anomaly level. Thus the base level optimization and the optimization of the mean density of a layer in *Occam h+d* inversion are trying to fit the same property. Therefore, *a priori* data (well data or outcropping) is needed to reduce ambiguity of the inverse problem.

In principle, height inversion methods (*Height*, *Occam h*) can also be used to estimate the depth extent of the structures. After optimizing the 3-D density distribution one can reset the density of the bottom layer to the background value, fix all the other layers except for the next to the last layer, and then optimize the height to find the variable depth extent of the blocks.

6.4.4 Lagrange multiplier

The *F-option* parameter in the right control pane is used in Occam inversion as a Lagrange multiplier. It determines if the inversion should give more importance to minimizing the data error instead of minimizing the model roughness, or vice versa. Since both the data values and the model parameters are normalized, the default value of the Lagrange muliplier (*L*=1.0), usually gives approximately the same importance for data and model error. Increasing the value (e.g., L= 10.) will emphasize the model roughness and the inversion will make a smoother model with the expense of increasing data error. On the contrary, decreasing the value (e.g., L= 0.1) will emphasize the data error and the inversion will make more rugged model, which will fit the data better. If computation time is not an object try testing different values of Lagrange multiplier.

Note that the Occam inversion enforces quite smooth parameter (density or height) variation. In real world the parameter changes can be quite abrupt. The Lagrange multiplier sets a global value for desired model roughness. The fix/free status and the roughness parameter, however, provide means to define local constraints. The fix/free status defines certain kind of rigidity around a fixed point. The roughness parameter discussed in the next chapter can be used to define discontinuities.

6.4.5 Parameter roughness

The concept of parameter roughness can be used to define discontinuities in Occam inversions. Rough blocks are not added to the mean of the surrounding blocks when the constraining roughness is computed for the Occam method. The objective is to enforce discontinuities that prevent the Occam method to continue the smooth model across it. The discontinuity information should be based on *a priori* information (or an assumption). Please, note that this method is still under development.

The parameter roughness is controlled by the roughness parameter, which is stored into the model file as the second parameter after the density. Roughness can be edited into the model or derived from imported topography (surfaces, lines, points) using the BLOXER program. For each block the roughness is defined using a single (real) value that consists of the sum of following values:

0	fully continuous block
1	fully discontinuous block (special case)
2	discontinuous towards positive *x* axis
4	discontinuous towards negative *x* axis
8	discontinuous towards positive *y* axis
16	discontinuous towards negative *y* axis
32	discontinuous towards positive *z* axis
64	discontinuous towards negative *z* axis

For example, roughness value 126 (2+4+8+16+32+64) means that the block is totally rough, 30 (2+4+8+16) means that the block is horizontally rough and vertically smooth, whereas 96 (32+64) means that the block is horizontally smooth and vertically discontinuous. Note that roughness value 1 is a special case, because normally the roughness must be an even value.

6.5 Depth weighting

The concept of depth weighting was introduced by Li and Oldenburg (1996 and 1998) in the 3-D inversion of magnetic and gravity data. Normally unconstrained (and un-weighted) inversion would place the masses in the uppermost blocks, because their sensitivity is much greater than that of the deeper blocks. Consequently, the resulting inverse model is not likely to be geologically realistic. Depth weighting tries to overcome this problem. The idea is to enhance the sensitivity of blocks at successively greater depth so that so that the inversion would place the masses at correct depths. In GRABLOX the depth weights (d_w) are computed numerically from the ratio ($d_w= G(z=0)/G(z)$ of the gravity field of a block that is placed just below the surface $G(z= 0)$ and a block at the corresponding depth ($G(z)$).

The biggest problem with depth weighting is that it distorts the sensitivity matrix in such a way that the convergence becomes much slower and several more iterations are needed to find the solution. To circumvent these problems GRABLOX has an option to relax the depth weighting using the log$_{10}$-ratio of the current RMS error and the ratio between optimal (1%) and maximal RMS error (100%). When the RMS error is large the depth weighting is strong. As the RMS error decreases during the inversion so does the depth weighting.

The *Gravity/Depth weight* menu item is used to choose the depth weighting method (*Enable relaxed* is the default method). The status of the current depth weighting method can be deduced from the last line of the information text in the graphs. The last line defines the current inversion method in plain text (*Depth*, *Occam d*, *Height*, *Occam h*, *Occam h+d*). At the end of the last line are two numbers inside parentheses (e.g., (0/1)). The first number reveals the current additional constraining method (0= none, 1= vertical, 2= horizontal, 3= 3-D smooth). The second number defines the depth weighting (0= none, 1= relaxed, 2= full).

6.6 Computation times

The computation time of the forward solution depends directly on the number of blocks and the number of points where the gravity field will be computed. As an example, consider a model where the number of blocks is N= 35×35×5= 6125 and the number of data points is M= 35×35= 1225. On a 2.4GHz Intel P4 PC the forward computation takes about 24 seconds.

The SVD based inversion has O^3 dependency on the dimension of the Jacobian (sensitivity matrix). In the abovementioned example one iteration takes over 2 hours to compute and most of this time is spent computing the decomposition.

Although the Occam inversion solves a much larger matrix system than the SVD based inversion (SVD: N×M vs. Occam: N×(N+M)), the use of an iterative conjugate gradient solver makes the computation much faster. In the example above a single iteration takes about 56 minutes.

7. File formats

Please, note that the formats of the input and output files have not been finalized in this version of GRABLOX and that they are likely to change in the future.

7.1 Model files

The following example illustrates the format of the input model *.INP file:

```
 3
  2980.0   6570.0  3840.0  7594.0   8.000   8.000
 .1000E+01 0.0000  0.0000  0.0000
  1   0   1   1   1   0
  3172.0   6634.0     0.0   384.0   768.0    48.0
  6  12   6
  1   1   0
  1
  432     1   1
  0   0.3000E+01 0.2500E+01 0.3500E+01 0.1000E+04
```

- The first line defines the number of following lines (3) that are specific to the GRABLOX program and its computation (and which will be skipped over by the BLOXER program).
- The 2.nd line defines the computation grid: x and y position of the grid start position (SW corner), x and y position of the grid end position (NE corner), and the grid spacing in x and y directions.
- The 3.rd line defines the basic model parameters: the (background) density value (g/cm^3), the base anomaly level (mGal), and the horizontal gradient of the base anomaly in x (EW) and y (NS) directions (mGal/100 distance units).
- The 4.th line defines some computational options:

1. Dimension: 1= meters, 2= kilometers, 3= miles.
2. Computation mode: 0= block gravity, 1= point gravity, 2= vertical gradient.
3. Gravity field: 0= anomalous field, 1= total field.
4. Background density: 0= given value, 1= mean layer density, 2= mean-block density
5. Parameter roughness: 0= not defined, 1= defined.
6. Reserved for future use.
7. Reserved for future use.
8. Reserved for future use.

- The 5.th line defines the *x*, *y* and *z* coordinates of the top-south-western corner of the super block, and the size of the super block in *x*, *y* and *z* directions. Note that the BLOXER program uses all the parameters after the 5.th line.
- The 6.th line defines the discretizations (number of blocks) in *x*, *y* and *z* directions.
- The 7.th line defines some properties of the block file:
  The 1.st parameter defines the block (*.BLX) file format (IFO):
  0 = The *.BLX file does not exist. In this case the super block will be discretized automatically and the default parameter value is given to all blocks.
  1 = Normal text file and real (floating point) values
  2 = Normal text file and 2-byte integer values
  3 = Binary file and real (floating point) values
  4 = Binary file and 2-byte integer values
  5 = Binary file and 2-byte integer values, the byte order of which are reversed when the data file is being read (allows Unix ↔ PC conversion).
  The 2.nd parameter defines the block type:
  1 = All blocks are assumed to have equal size (default)
  2 = Block height is increasing with depth. A negative value of this parameter would mean that block reduction has been used.
  The 3.rd parameter is reserved for future use.
- The parameter on the 8.th line defines the zooming level (1, 2, 3, …, 10).
  Note that if the zooming level is bigger than 1, the next line would contain the *x*, *y*, and *z* coordinates of the center of the zoom-block, and the dimension of the zoom-block in *x*, *y* and *z* directions. Note that GRABLOX does not utilize the zoom options.
- The 9.th line (in this case) defines the number of blocks, the number of characteristic parameters, and the number of the column that contains the density value in the *.BLX file. Note that if block reduction had been used and the block file would have been saved using the *File/Save red. model* menu item, the number of actual minor blocks might be smaller than the value based on discretization.
- The last line contains the definition values of the density:
  1. Scale type: 0= linear scale, 1= logarithmic scale.
  2. Default parameter value (used instead of that at the 3.rd line)
  3. Parameter minimum value (for color scale and inversion limits)
  4. Parameter maximum value (for color scale and inversion limits)
  5. The scaling factor (multiplier for labels in layer and section views).

In principle there is no need to edit model files manually. However, if the *.BLX file is missing one can still open the model by resetting the file format parameter (IFO=0) on the 7.th line so that block model will be generated automatically. Also if the *.BLX file was generated using binary 2-byte integer format on a Unix workstation, then the model can be read on a IBM/PC by resetting the file format parameter (IFO=5). Note that real-valued binary files cannot be transported across different computer platforms. The third reason to edit *.INP files is when activating the parameter roughness on the 4.th line. Because the BLOXER program must be used to incorporate roughness into the model and BLOXER ignores the GRABLOX specific header lines (2-4), the user needs to manually enable the parameter roughness. Only then GRABLOX reads the parameter roughness data from the *.BLX file. Please, note that the use of parameter roughness is still at an experimental stage.

The *.BLX file has simple column format, where the columns correspond to: the *x, y, z* size of the blocks (columns 1-3), the *x, y, z* position of the blocks (columns 4-6), the fix/free/hidden/ reduced block status (column) 7, and the parameter columns (8-17). Normally, the density values are in the 8.th column. If roughness parameter is used it will be stored right after density values (9.th column). The horizontal *xy* position of a minor-block is fixed to the center of the rectangular block area, but the vertical *z* position is fixed to the top of the block. Moreover, the positive *z* axis points downwards. Note that the order of the do-loops used to save the model data is: *z* loop, *y* loop, *x* loop. This means that for each layer (from top to bottom) the blocks are saved so that for each *y* coordinate value (from South to North) the data is saved along x-axis (from West to East). Please, see the user's manual of the BLOXER program for additional information considering the block model.

7.2 Data files

The following example illustrates the format of a *.DAT file, which is used to read in measured data. The same format is used also when using the *File*/*Save data* menu item.

```
Example data

153 1 2 0 3 0

 -200.00  -200.00 0.2007733E-01
 -150.00  -200.00 0.2855821E-01
 -100.00  -200.00 0.3860382E-01
  -50.00  -200.00 0.4719916E-01
    0.00  -200.00 0.5056060E-01
   50.00  -200.00 0.4759879E-01
 etc ...
```

The 1.st line is a header that will be used in the response graphs as a secondary title. This line can be left empty line. The 2.nd and the 4.th line are used for comments and can be left empty. The 3.rd line contains six values.

1. The total number of data points.
2. The column index of the *x* (east) coordinate,
3. The column index of the *y* (north) coordinate.
4. The column index of the *z* (topography) coordinate.
5. The column index of the data column.
6. Regional field (0= no regional field, 1= regional field exists).

Note that GRABLOX can be used also in 2.5-D interpretation. If either the *x* or the *y* coordinate is missing (the corresponding column index = 0) the data are assumed to be in profile format and will be put along the *x* axis (*y*= 0). If both the *x* and *y* coordinates are missing, the data are assumed to be in profile format and will put along the *x* axis and the *x* coordinates are given dummy values (0, 10, 20, …).

Note that the *x* and *y* columns can be defined reversely (i.e., *y* column can be before *x* column). The *z* coordinate, however, must be defined after the *x* and *y* coordinates and the data columns must be located after the coordinate columns. The coordinates themselves can be irregularly spaced and do need not to span a regular grid. If the data file contains regional field, then it will be read from the column right after the data column.

The abovementioned file format is used also when saving the data into a file using the *File*/*Save data* menu item. In this case, however, the computed data are stored in the fourth column, the regional field (if it exists) in the next column, the base anomaly level in the next column and the measured data (if it exists) in the last column.

7.3 Output files

Information concerning the model and system parameters used in the computation is stored in an output file (*.OUT). The *.OUT file contains also information about the inversion. Since the *.OUT file is quite self-explaining it will not be described here.

The computed gravity field data are stored in a separate *.GBM file and the description of this file is saved into a header *.HEG file. Note that the *.GBM file will contain also the measured data if it has been read in. An example of the header file format is shown below.

```
Header file for saved gravity field data (GRABLOX generated)
N:o lines, n:o columns, x, y ,z, data, format 1, format 2
  231  5  1  2  3  4  1  0

Name of the data file (text):
D:\Grablox\Example.gbm
```

The 1.st and the 2.nd line are used merely for comments. The 3.rd line defines:

1. Number of lines (grid positions or measurement locations).
2. Total number of columns (needed when reading binary data).
3. Column index of the *x* coordinate.
4. Column index of the *y* coordinate.
5. Column index of the *z* coordinate.
6. Column index of the computed data.
7. Regional option (0= no regional, 1= regional data exists)
8. An additional parameter reserved for future use

The 4.th and 5.th line are used as a separator and a comment line. The 6.th line defines the name (and directory path) of the *.GBM file.

The format of the *.GBM file is the same as that of the *.DAT output data file described in the previous chapter. The only exception is that the *.GBM file does not contain any header information. Note that the column order of the *.GBM file is defined also at the end of the *.OUT file.

7.4 Graph options

Several graph parameters are stored in the GRABLOX.DIS file. Note that if the format of the file should become invalid, one should delete the *.DIS file and a new one with default parameter values will be generated automatically the next time the program is started. The format of the GRABLOX.DIS file is shown below. Note that one can edit the *.DIS file manually and use the *File/Read disp* menu item make the changes visible.

```
 36   32   26   26   20
  1    1    1    0    0    0
300  460 0.60 0.85
150.  30.   4.

Gravity field modeling and inversion
Gravity anomaly
Vertical gradient
Computed
Measured
Regional
Difference
X / East
Y / North
Z / Depth
Distance
Density (g/cm^3)
Log10(Density)
(mGal)
(mGal/unit)
```

- The 1.st line defines the five character heights used for (1) the main title and the graph axis titles, (2) the axis labels, (3) the plot legend text, (4) the model description text, and (5) the axis labels in the 3-D model view.
- The 2.nd line defines four option parameters that modify the graph appearance.
  1. Option to include (1) or exclude (0) the model information text to/from the graph.
  2. Option to include (1) or to exclude (0) the model view to/from the graph.
  3. Option to define the corner where the legend text is positioned. Values 1-4 put the legend in SW, SE, NE or NW corner of the page (outside the graph). Values 5-8 put the legend in the SW, SE, NE, or NW corner inside the graph.
  4. Option that defines the color scale: 0= rainbow, 1= reverse rainbow, 2= grayscale, 3= reverse grayscale, and 4= temperature (red) scale.
  5. The remaining two parameters are reserved for future use.
- The 3.rd line defines the *x* (horizontal) and *y* (vertical) distance of the origin of the main graph (in pixels) from the bottom-left corner of the page, and the length of the longest axis relative to the size of the remaining (origin shifted) width and height of the plot area. The total size of the plot area is 2970×2100 pixels representing landscape A4 paper size.
- The 4.th line defines horizontal and vertical viewing angles and a perspective viewing distance used in the 3-D model views.
- The 5.th line should be left empty.
- The following lines define various text items of the graph (max. 70 chars). These are:
  - Main title of the graphs.
  - 2 subtitles of the contour maps (also the names of the y axes of the profile graph).
  - 3 titles for the contour maps and/or the three legends for the profile graph.
  - 3 axis titles for the contour maps and the 3-D model view.
  - X axis title of the profile graph (note that units m/km/ml are omitted by default).
  - 2 possible color scale titles in layer and section maps.
  - 2 text fields that determine the units of the subtitles (gravity anomaly or gradient).

8. Additional information

I started to develop the GRABLOX program at the Geological Survey of Finland in early 2003, when I was working for the 3-D crustal model project funded by the Academy of Finland. The forward computation is based on the GPRISM algorithm by Sven-Erik Hjelt (1974). Further information about the geophysical use of gravity measurements can be found from any textbook on applied geophysics.

The program is written mostly in standard Fortran90 but contains some extensions of Compaq Visual Fortran 6.6. The user interface and the graphs are based on DISLIN graphics library (version 8.2) by Helmut Michels (http://www.dislin.de). Note that the version of the DLL file depends on the build version of the EXE file and one should update also the DLL file when updating the GRABLOX program. The program distribution package always includes the proper DISLIN.DLL file.

Since the DISLIN graphics library is independent form the operating system the program can be compiled and run on other operating systems (Solaris, Linux) without any major modifications. This requires that proper versions of Motif and DISLIN libraries have been installed and a suitable Fortran90 compiler is available. However, the source code of GRABLOX is not publicly available at the moment and I do not intend to prepare RPM distribution files for Linux systems in the near future. Therefore, Linux users should wait for some time (sorry). If you have suggestions for improvements, please, inform me.

Because the GRABLOX program is under active development, this user guide may not describe all the latest changes and additions made to the software. Additional information can be obtained also from Author's WWW-site at: http://www.gf.oulu.fi/~mpi.

9. References

Hohmann G.W. and Raiche A.P., 1988. Inversion of controlled-source electromagnetic data. In: Nabighian, M.N. (Ed.) Electromagnetic methods in applied geophysics, Volume 1, Theory. Soc. of Expl. Geophys., Tulsa, p. 469-503.

Hjelt S.-E., 1974. The gravity anomaly of a dipping prism. Geoexploration, 12, 29-39.

Jupp D.L.B and Vozoff K.,1975. Stable iterative methods for the inversion of geophysical data. Geophys. J. R. Astr. Soc. 42: 957-976.

Li, Y. and Oldenburg, D.W., 1996. 3-D inversion of magnetic data. Geophysics, 61, 394-408.

Li Y. and Oldenburg D.W., 1998. 3-D inversion of gravity data. Geophysics, 63, 109-119.

Pirttijärvi M., 2003. Numerical modeling and inversion of geophysical electromagnetic measurements using a thin plate model. PhD thesis, Acta Univ. Oul. A403, University of Oulu.

Pirttijärvi M., 2004. BLOXER - Interactive visualization and editing software for 3-D block models, Version 1.5, User's guide, Geological Survey of Finland, Report Q16.2/2004/1.

10. Terms of use and disclaimer

The GRABLOX program is still under development, and therefore it is not publicly available.

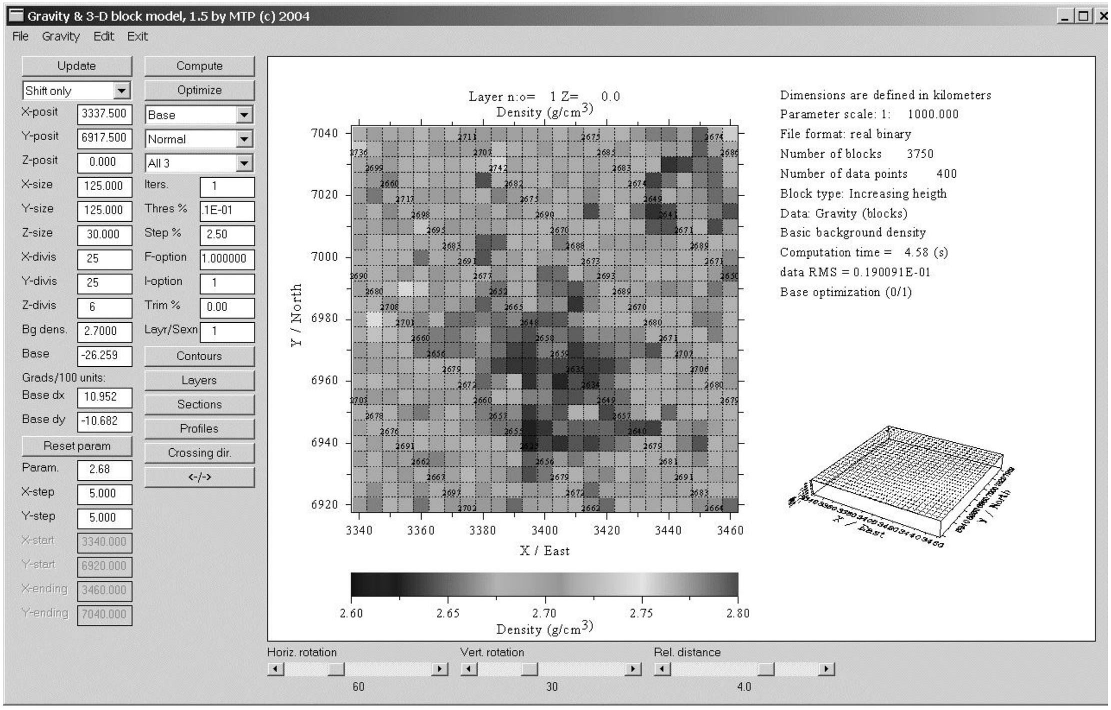
The program is provided as is. The Author, University of Oulu, and Geological Survey of Finland disclaim all warranties, either expressed or implied, with regard to this software. In no event shall the Author, University of Oulu, and Geological Survey of Finland be liable for any indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, arising out of or in connection with the use or performance of this software.

11. Contact information

Markku Pirttijärvi
Division of Geophysics
Department of Geosciences
PO Box 3000
FIN-90014 University of Oulu
Finland

Tel: +358-8-5531409
Fax: +358-8-5531484
URL: http://www.gf.oulu.fi/~mpi
E-mail: markku.pirttijarvi@oulu.fi

Appendix A: Layer view of the 3-D block model



Shift only

X-posit 3337.500

Y-posit 6917.500

Z-posit 0.000

X-size 125.000

Y-size 125.000

Z-size 30.000

X-divis 25

Y-divis 25

Z-divis 6

Bg dens. 2.7000

Base -26.259

Grads/100 units:

Base dx 10.952

Base dy -10.682

Param. 2.68

X-step 5.000

Y-step 5.000

X-start 3340.000

Y-start 6920.000

X-ending 3460.000

Y-ending 7040.000

Base

Normal

All 3

Iters. 1

Thres % .1E-01

Step % 2.50

F-option 1.000000

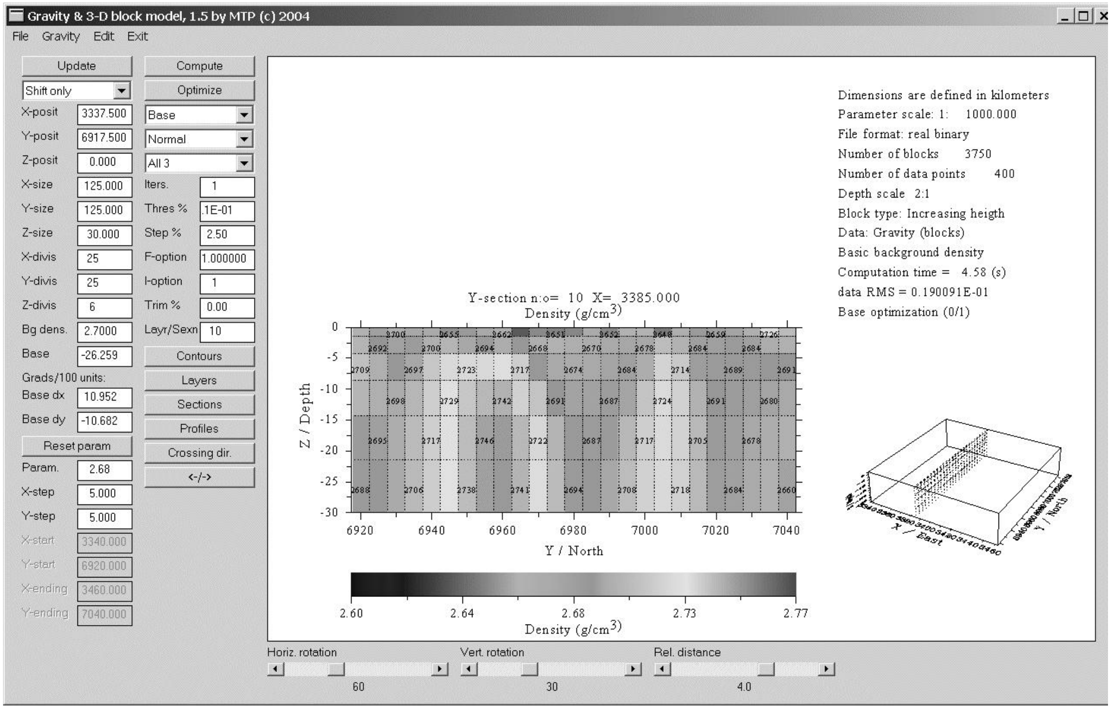
I-option 1

Trim % 0.00

Layr/Sexn 1

Dimensions are defined in kilometers
Parameter scale: 1: 1000.000
File format: real binary
Number of blocks 3750
Number of data points 400
Block type: Increasing heigth
Data: Gravity (blocks)
Basic background density
Computation time = 4.58 (s)
data RMS = 0.190091E-01
Base optimization (0/1)

Appendix B: Section view of the 3-D block model



Shift only

X-posit 3337.500

Y-posit 6917.500

Z-posit 0.000

X-size 125.000

Y-size 125.000

Z-size 30.000

X-divis 25

Y-divis 25

Z-divis 6

Bg dens. 2.7000

Base -26.259

Grads/100 units:

Base dx 10.952

Base dy -10.682

Param. 2.68

X-step 5.000

Y-step 5.000

X-start 3340.000

Y-start 6920.000

X-ending 3460.000

Y-ending 7040.000

Base

Normal

All 3

Iters. 1

Thres % 1E-01

Step % 2.50

F-option 1.000000

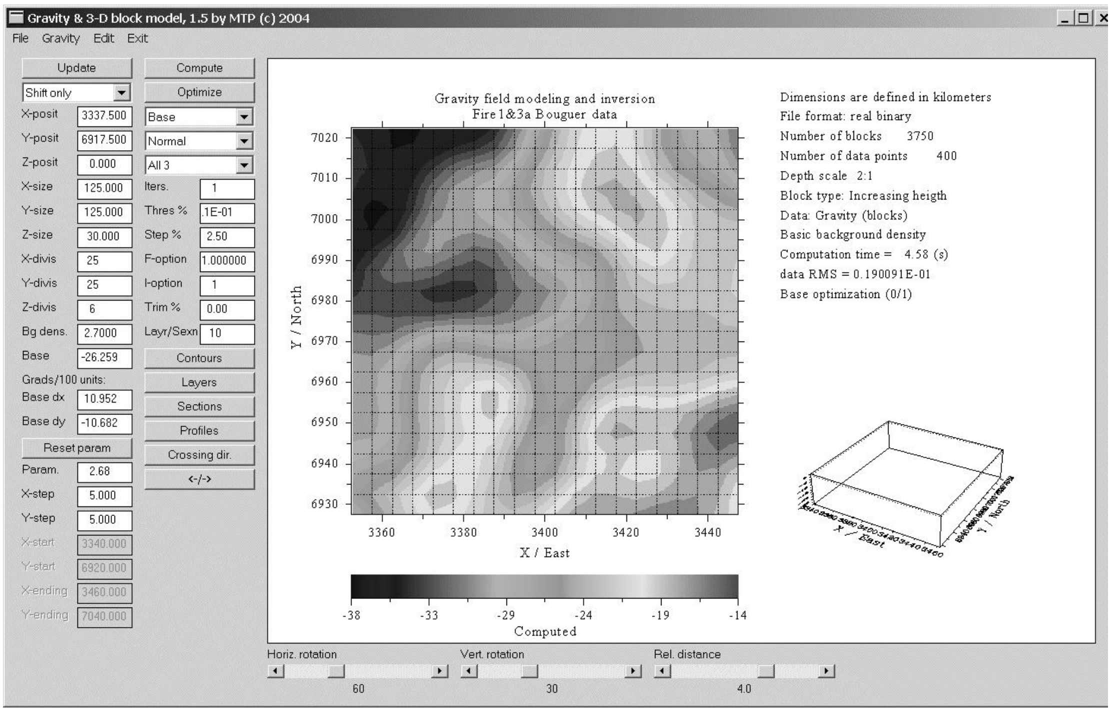
I-option 1

Trim % 0.00

Layr/Sexn 10

Dimensions are defined in kilometers
Parameter scale: 1: 1000.000
File format: real binary
Number of blocks 3750
Number of data points 400
Depth scale 2:1
Block type: Increasing heigth
Data: Gravity (blocks)
Basic background density
Computation time = 4.58 (s)
data RMS = 0.190091E-01
Base optimization (0/1)

Appendix C: Contour plot of (computed) gravity field



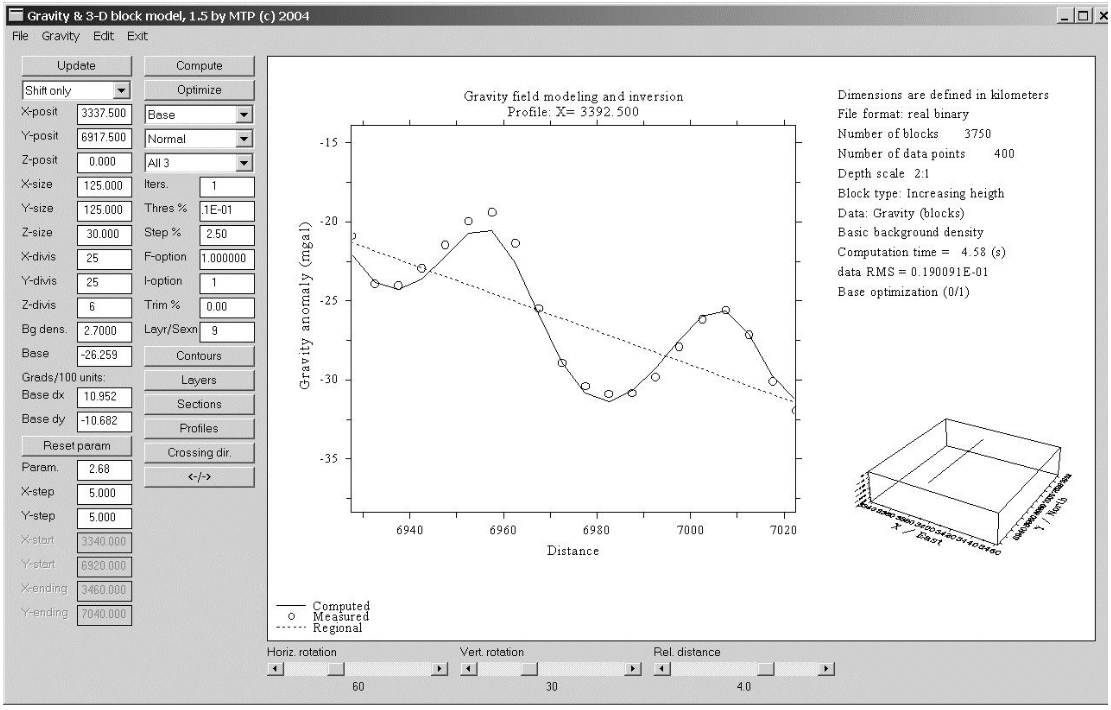
Shift only
X-posit 3337.500
Y-posit 6917.500
Z-posit 0.000
X-size 125.000
Y-size 125.000
Z-size 30.000
X-divis 25
Y-divis 25
Z-divis 6
Bg dens. 2.7000
Base -26.259
Grads/100 units:
Base dx 10.952
Base dy -10.682
Param. 2.68
X-step 5.000
Y-step 5.000
X-start 3340.000
Y-start 6920.000
X-ending 3460.000
Y-ending 7040.000

Base
Normal
All 3
Iters. 1
Thres % .1E-01
Step % 2.50
F-option 1.000000
I-option 1
Trim % 0.00
Layr/Sexn 10

Dimensions are defined in kilometers
File format: real binary
Number of blocks 3750
Number of data points 400
Depth scale 2:1
Block type: Increasing heigth
Data: Gravity (blocks)
Basic background density
Computation time = 4.58 (s)
data RMS = 0.190091E-01
Base optimization (0/1)

Horiz. rotation 60
Vert. rotation 30
Rel. distance 4.0

Appendix D: Profile plot of computed, regional, and measured gravity field



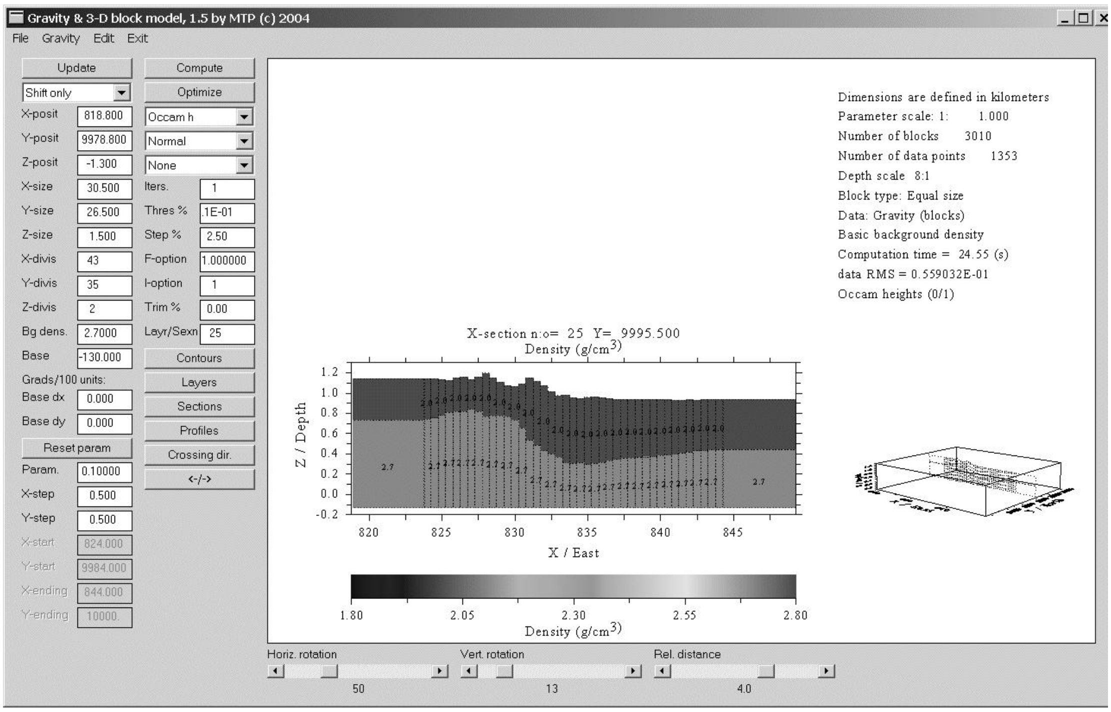
Shift only
X-posit 3337.500
Y-posit 6917.500
Z-posit 0.000
X-size 125.000
Y-size 125.000
Z-size 30.000
X-divis 25
Y-divis 25
Z-divis 6
Bg dens. 2.7000
Base -26.259
Grads/100 units:
Base dx 10.952
Base dy -10.682
Param. 2.68
X-step 5.000
Y-step 5.000
X-start 3340.000
Y-start 6920.000
X-ending 3460.000
Y-ending 7040.000

Base
Normal
All 3
Iters. 1
Thres % .1E-01
Step % 2.50
F-option 1.000000
I-option 1
Trim % 0.00
Layr/Sexn 9

Dimensions are defined in kilometers
File format: real binary
Number of blocks 3750
Number of data points 400
Depth scale 2:1
Block type: Increasing heigth
Data: Gravity (blocks)
Basic background density
Computation time = 4.58 (s)
data RMS = 0.190091E-01
Base optimization (0/1)

Horiz. rotation 60
Vert. rotation 30
Rel. distance 4.0

Appendix E: Two-layer interpretation of gravity



Shift only

X-posit 818.800
Y-posit 9978.800
Z-posit -1.300
X-size 30.500
Y-size 26.500
Z-size 1.500
X-divis 43
Y-divis 35
Z-divis 2
Bg dens. 2.7000
Base -130.000
Base dx 0.000
Base dy 0.000
Param. 0.10000
X-step 0.500
Y-step 0.500
X-start 824.000
Y-start 9984.000
X-ending 844.000
Y-ending 10000.

Occam h
Normal
None
Iters. 1
Thres % .1E-01
Step % 2.50
F-option 1.000000
I-option 1
Trim % 0.00
Layr/Sexn 25

Dimensions are defined in kilometers
Parameter scale: 1: 1.000
Number of blocks 3010
Number of data points 1353
Depth scale 8:1
Block type: Equal size
Data: Gravity (blocks)
Basic background density
Computation time = 24.55 (s)
data RMS = 0.559032E-01
Occam heights (0/1)

Horiz. rotation 50
Vert. rotation 13
Rel. distance 4.0