

# **GRAV3D Version 3.0**

## **A Program Library for Forward Modelling and Inversion of Gravity Data over 3D Structures.**

UBC-Geophysical Inversion Facility  
Department of Earth and Ocean Sciences  
University of British Columbia  
Vancouver, British Columbia

May 2005



© UBC-Geophysical Inversion Facility 2001 - 2005

On this page: [Description](#) | [Package contents](#) | [Licensing](#) | [Installation](#)

## Description

GRAV3D is a program library (version 3.0 as of August 2005) for carrying out forward modelling and inversion of surface, airborne, and/or borehole gravity data in three dimensions. The program library carries out the following functions:

- Forward modelling of the vertical component of the gravity response to a 3D volume of density contrast. The model is specified using a mesh of rectangular cells, each with a constant value of density contrast, and topography is included. The gravity response can be calculated anywhere within the model volume, including above the topography simulating ground or airborne surveys, and inside the ground simulating borehole surveys.
- Inversion of surface, airborne, and/or borehole gravity data to generate 3D models of density contrast.
  - The inversion is solved as an optimization problem with the simultaneous goals of (i) minimizing an objective function on the model and (ii) generating synthetic data that match observations to within a degree of misfit consistent with the statistics of those data.
  - To counteract the inherent lack of information about the distance between source and measurement, the formulation incorporates a depth or distance weighting term.
  - By minimizing the model objective function, distributions of subsurface density contrast are found that are both close to a reference model and smooth in three dimensions. The degree to which either of these two goals dominates is controlled by the user by incorporating *a priori* geophysical or geological information into the inversion. Explicit prior information may also take the form of upper and lower bounds on the density contrast in any cell.
  - The regularization parameter (controlling relative importance of objective function and misfit terms) is determined in either of three ways, depending upon how much is known about errors in the measured data.
- The large size of useful 3D inversion problems is mitigated by the use of wavelet compression. Parameters controlling the implementation of this compression are available for advanced users.

The research underlying this program library was funded principally by the mineral industry consortium "Joint and Cooperative Inversion of Geophysical and Geological Data" (1991 - 1997) which was sponsored by NSERC (Canada's **National Science and Engineering Research Council**) and the following 11 companies: BHP Minerals, CRA Exploration, Cominco Exploration, Falconbridge, Hudson Bay Exploration and Development, INCO Exploration & Technical Services, Kennecott Exploration Company, Newmont Gold Company, Noranda Exploration, Placer Dome, and WMC.

The theoretical framework for GRAV3D is provided in the following paper (see the UBC-GIF website [publications page](#) for details):

- Li Y., and Oldenburg D.W. (1998), "3D inversion of gravity data", *Geophysics*, **63**, No.1, 109-119.

Two short papers including examples of applying GRAV3D in mineral exploration contexts are:

- *Cost effectiveness of geophysical inversions in mineral exploration: Applications at San Nicolas*, Nigel Phillips, Doug Oldenburg, and Jiuping Chen, Yaoguo Li, Partha Routh, 2001, *The Leading Edge*, Volume 20, Issue 12 p. 1351
- *Applications of Geophysical Inversions in Mineral Exploration Problems*, Oldenburg D.W., Li Y., Farquharson C.G., Kowalczyk P., Aravanis T., King A., Zhang P., and Watts A. (1998), *The Leading Edge*, 17, 461 - 465.

## Software package contents

The package that can be licensed includes the following components:

- **Executable programs** for performing 3D forward modelling and inversion of gravity surveys. The GRAV3D library (WindowsXX or Linux platforms) consists of three major programs and one utility:
  - GZFOR3D: performs forward modelling.
  - GZSEN3D: calculates sensitivity and the depth weighting function.
  - GZINV3D: performs 3D magnetic inversion.
  - GZPRE3D: multiplies the sensitivityfile by the model to get the predicted data.
- **A graphical user interface** is supplied for the WindowsXX platforms **only**. Facilities include
  - GRAV3D-GUI.EXE: a primary interface for setting up the inversion and monitoring the progress of calculations;
  - GM-DATA-VIEWER: a utility for viewing raw surface or airborne data (but not borehole data), error distributions, and for comparing observed to predicted data directly or as difference maps;
  - MESHTOOLS3d: a utility for displaying resulting 3D models as volume renderings. Susceptibility volumes can be sliced in any direction, or isosurface renderings can be generated.
- **Documentation** is elsewhere via the menu to the left. **Note** there is currently no documentation for the GRAV3D-GUI user interface. However, it is very similar to the MAG3D GUI, and there is documentation for

MAG3D-GUI.EXE.

- **Example data sets and exercises** are provided on the [IAG](#) CD-ROM.

## Licensing

A **constrained educational** version of the program is available with the [IAG](#) CD-ROM. The educational version is fully functional so that users can learn how to carry out effective and efficient 3D inversions of gravity data. **However, RESEARCH OR COMMERCIAL USE IS NOT POSSIBLE because the educational version will NOT work with more than 200 data points or 12,000 cells in the 3D mesh.**

Licensing for an **unconstrained academic** version is available - see the [licensing policy document](#) (on the UBC-GIF website).

**NOTE:** all academic licenses will be **time-limited to one year**. You can re-apply after that time. This ensures that everyone is using the most recent versions of codes.

Licensing for **commercial use** is managed by distributors, not by the UBC-GIF research group. Details are in the [licensing policy document](#).

## For learning and documentation:

For links to documentation, related utilities, and examples, see the menu to the left.

## Installing GRAV3D version 3.0 - educational version

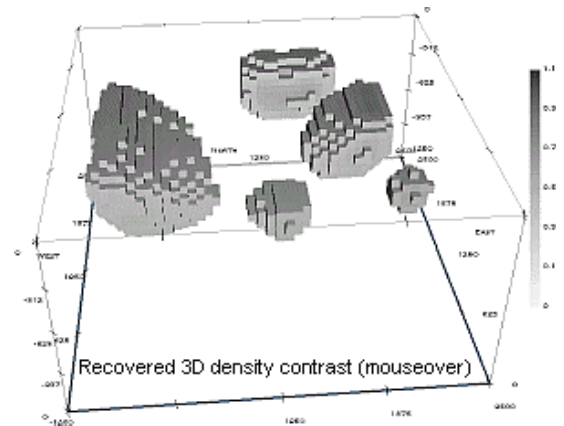
### For users with a copy of IAG only!

1. **Copy all files in [this folder](#) onto your computer.**  
Place them all together in a new folder such as c:\ubcgif\grav3d\
2. No further installation is necessary.
3. Follow instructions in one of this CD-ROM's exercises, or refer to the program documentation.
4. Recall that this is an educational version. **Codes will not work with more than 200 data points or 12,000 cells in the 3D mesh.**

## The Program and its purpose

The program library GRAV3D is a suit of algorithms for inverting gravity data gathered over a three dimensional earth. Version 1.0 of this facility is described in detail in Li and Oldenburg (1998), and subsequent versions added greater flexibility and efficiency to the basic forward modelling and inversion algorithms. In particular it allowed larger problems to be solved through the use of wavelet transforms, and it allowed geophysical constraints, in the form of upper and lower bounds on the density of each cell, to be included. Notes about Version 3.0 are below.

The problem addressed by GRAV3D involves gravimetric data gathered anywhere at or above the surface of the Earth. These data are the vertical component of the gravity field caused by a three dimensional distribution of density contrast within the volume of ground directly beneath the survey area. This subsurface volume (with optional topography) is modelled as a set of rectangular cells each with constant density contrast. For forward calculations the anomalous density in each cell is known and the data that be measured over this known Earth model are calculated. The inverse problem involves estimating the density contrasts of all the cells based upon measurements gathered during a field survey. In the remainder of this introduction we point out some highlights, and we emphasize the importance of understanding what kind of models the program can recover, and how the program operates. Changes implemented for GRAV3D Version 3.0 are outlined at the end of this chapter.



## Summary of sections on theoretical background

The introduction provides a basic understanding of how gravity data relate to the Earth, and the goal of inverting such data.

The solution to the forward problem is described, and an example of a forward calculation is provided.

The method of solving the gravity inverse problem is outlined. In summary, the inversion is solved as an optimization problem with the simultaneous goals of i) minimizing an objective function on the model and ii) generating synthetic data that match observations to within a degree of misfit consistent with the statistics of those data.

- By minimizing the model objective function, distributions of subsurface density contrast are found that are both close to a reference and smooth in three dimensions. The degree to which either of these two goals dominates is controlled by defining length scales for smoothness. This is a crucial step and allows the user to incorporate a priori geophysical or geological information into the inversion. Explicit prior information may also take the form of upper and lower bounds on the density contrast in any cell.
- The relative importance of the objective function the misfit term is controlled by a regularization parameter. This parameter is determined in one of three ways, and depends upon how much is known about errors in the measured data. Section 1.6 discusses this problem, and it is important to understand how the choice of options affects the outcome from the inversion program.

Potential fields data have no inherent information about the distance between source and measurement, therefore the incorporation of a depth or distance weighting term in the formulation is critical. Section 1.4 describes this issue, and explains the options available for controlling how cells in the model enter into the solution regardless of their depth or distance from the measurements.

The large size of useful 3D inversion problems is mitigated by the use of wavelet compression. Parameters controlling the implementation of this compression are available for advanced users, and section 1.5 provides some details on how wavelet compression is applied.

## Summary of program elements, instructions for use, and examples

Files and file formats are similar to those of other UBC-GIF codes, although there are some aspects specific to GRAV3D

V3.0. Highlights are noted below, but you must read Chapter 2 of the manual for details.

The mesh file defines how the 3D subsurface volume of ground is discretized. Specify cell sizes that will image your target with adequate detail without resulting in a too many model cells. Then add padding cells outside the region of investigation. Recall that the program must invert a sensitivity matrix that has a size proportional to  $N \times M$ , where  $N$  is the number of data points and  $M$  is the total number of cells in the model. This sensitivity matrix should reside within the computer's memory for efficient execution. Problems with more than 10,000 to 20,000 model cells, and/or more than a few thousand data points would be considered large, and may be expected to require a considerable amount of computing time.

Also, it is considered prudent to upward continue data that are gathered very close to the ground so that measurements appear as if collected at an elevation roughly equal to half a cell width. This is especially true in the presence of severe topography.

Files that define topography, observation locations and observations (measured data) are self explanatory. The manual includes details on how to ensure consistency regarding elevations and topography.

There are separate model files defining density contrasts for the initial model, reference model, upper and lower bounds on density contrast, and final output models, all with similar structures. These models are defined via the mesh file, and each cell has a constant density contrast. Also, you should be clear on how cells above topography are managed.

Forward modelling, sensitivity calculation and inversion programs are run from the command line with parameters specified in separate files. In the descriptions of these parameter files look for details regarding depth or distance weighting, wavelet compression, and how to specify model objective function parameters. There are also important notes on inversion "mode". Your choice of mode determines what method the program uses to find the regularization parameter. This choice is based upon what type of information you have on data errors, and error statistics. Output files are also specified, and you should take particular note of the information provided in the log file. This file summarizes the progress of inversion, and careful examination of its information is a critical aspect of quality control. The log file contents depend upon which mode was use for inversion; see the final section of chapter 3 for details.

The manual concludes with two synthetic examples. The first illustrates the newly added facility for specifying upper and lower bounds on recovered cell density contrasts. The second example illustrates inversion of a larger problem.

## Conclusions

Successful application of inversion results to geological problems demands understanding of the inversion process. While models will be as smooth and as close to a reference model as data misfit will allow, the actual results are controlled by careful selection of length scales, weighting functions, and constraining parameters. The size of the problem is reduced using wavelet compression and the management of misfit is controlled by the choice of mode and associated parameters. This degree of flexibility makes it imperative that the user has a good understanding of general inversion theory and the specifics of its implementation for GRAV3D. Finally, please be sure to read the notes regarding changes implemented for Version 3.0.

## NOTES: GRAV3D Ver 3.0 (June 2005) - changes to the code and the manual

As might be expected, more recent UBC-GIF codes have features that have been found important for solving practical problems but these features were not included in earlier program libraries. The upgrades described below address this issue. The revised codes are more uniform in capabilities and are more computational efficient.

### Improvements since version 2.0

- A new preconditioner for solving the Gauss-Newton system of equations results in significantly improved performance.
- All values, except for the stored sensitivity matrix, are now in double precision. This results in more accurate calculations.
- When entering UTM coordinates, which have a lot of significant digits, the user no longer has to subtract a constant.
- A file `sensitivity.txt` is output after running `gzsen3d.exe`. It contains the average sensitivity for each cell. This file can be used for depth of investigation analysis or for use in designing special model objective function weighting.
- A file `gzinv3d_nopos.den` is output during the first part of the inversion. It contains the densities without the bounds constraints imposed.
- A file `gzinv3d_XX.den` is output after each beta iteration.

- In `gzinv3d.exe` the user can enter either alpha values or length scales.
- The reference model is now included in the calculation of the model norm.
- The reference model is now scaled by the depth weighting before starting the no-positivity iterations.
- Sensitivity calculations carried out by `gzsen3d.exe` are now more efficient.

### Changes to run-time files for GRAV3D Version 3.0

None.

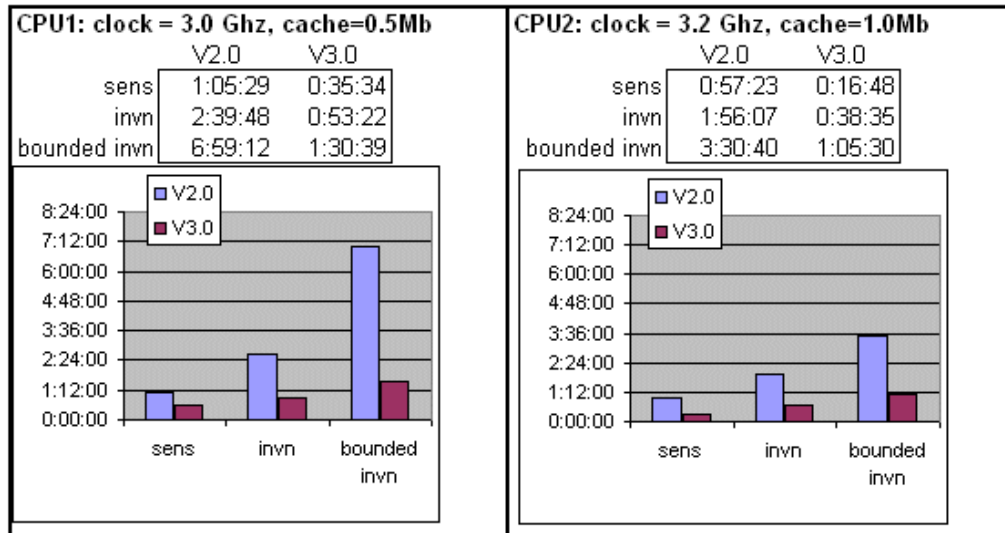
### Changes to this manual for GRAV3D Version 3.0

This introductory note, and a few additions to "execution".

### Notes on computation speed

#### GRAV3D computation times on two computers, for both versions 2.0 and 3.0.

# of data: 1839 # of cells: 920,856



Run times for GRAV3D Ver2.0 and Ver3.0 are compared in the figures above for a moderately complicated problem. This was a real example involving 1839 data points and 920856 cells in the model, and moderate topography.

- The Version 3.0 code is significantly faster in all cases.
- Speed also depends strongly on the computer. Times when using two computers are shown below. Both were Pentium IV processors with 1 Gbyte of RAM. The significant difference however concerns the size of the CPU's cache memory. Most good computers sold since roughly 2005 have 1.0Mb of cache memory and this results in a significant improvement in time to complete computation-intensive jobs.
- The complexity of the problem also affects computation times. In the table, compare times when bounds are included to times when no bounds are included. All other aspects of the inversions are identical.
- These results are presented for illustration only. The time to compute any given problem is strongly dependent upon the number of data points, the size of the mesh, and how you set up all parameters for the inversion, including data, constraints, regularization, compression, etc.

This page: | [Introduction](#) | [Forward modelling](#) | [Inversion](#) | [Depth weighting](#) | [Wavelet compression](#) | [Regularization](#) | [Example](#) |

## 1.1 Introduction

This manual presents theoretical background, numerical examples, and explanation for implementing the program library GRAV3D. This suite of algorithms, developed at the UBCGeophysical Inversion Facility, is needed to invert gravimetric responses over a 3 dimensional distribution of density contrast, or anomalous density (henceforth referred to as 'density' for simplicity). The manual is designed so that geophysicists who are familiar with the gravity experiment, but who are not necessarily versed in the details of inverse theory, can use the codes and invert their data. In the following, we describe the basics of the algorithm, but readers are referred to Li and Oldenburg (1997, 1998) for an in-depth discussion of various aspects of the algorithm. Note that an understanding of these components is necessary for the user to have a global view of the algorithms and to use the program library correctly.

A gravity experiment involves measuring the vertical components of the gravity field produced by anomalous (either excess or deficient) mass beneath the surface. A distribution of anomalous mass, characterized by density  $\rho(x; y; z)$ , produces its own gravity field,  $\vec{F}_s$ , which is superimposed on the ambient gravity field. By measuring the resultant field and removing the ambient field from the measurements through numerical processing, one obtains the field due to the anomalous mass.

The vertical component of the gravity field produced by the density  $\rho(x; y; z)$  is given by

$$F_z(\vec{r}_0) = \gamma \int_V \rho(\vec{r}) \frac{z - z_0}{|\vec{r} - \vec{r}_0|^3} dv \quad (1)$$

where  $\vec{r}_0$  is the vector denoting the observation location and  $\vec{r}$  is the source location.  $V$  represents the volume of the anomalous mass, and  $\gamma$  is the gravitational constant. Here we have adopted a Cartesian coordinate system having its origin on the earth's surface and the z-axis pointing vertically downward.

The data from a typical gravity survey are a set of field measurements acquired on a 2D grid over the surface. These data are first processed to yield an estimate of the anomalous field, which is due to the excess or deficient mass below the data area. The goal of the gravity inversion is to obtain, from the extracted anomaly data, quantitative information about the distribution of the anomalous density in the ground.

The GRAV3D library consists of three programs:

1. GZFOR3D: calculates the surface gravity data produced by a given density model.
2. GZSEN3D: calculates the sensitivity matrix for use in the inversion.
3. GZINV3D: inverts the anomalous gravity field to generate a density model.

In the following, we outline the basics of the forward and inverse procedures used by these programs.

## 1.2 Forward Modelling

Forward modelling of gravity data is a linear problem and can be carried out by performing the integration in eq.(1). We divide the region of interest into a set of 3D prismatic cells by using a 3D orthogonal mesh and assume a constant density within each cell. We discretize the density model in this manner since it is best suited for our inversion methodology. Given such a discretization, the gravity field at the  $i$ 'th location can be written as,

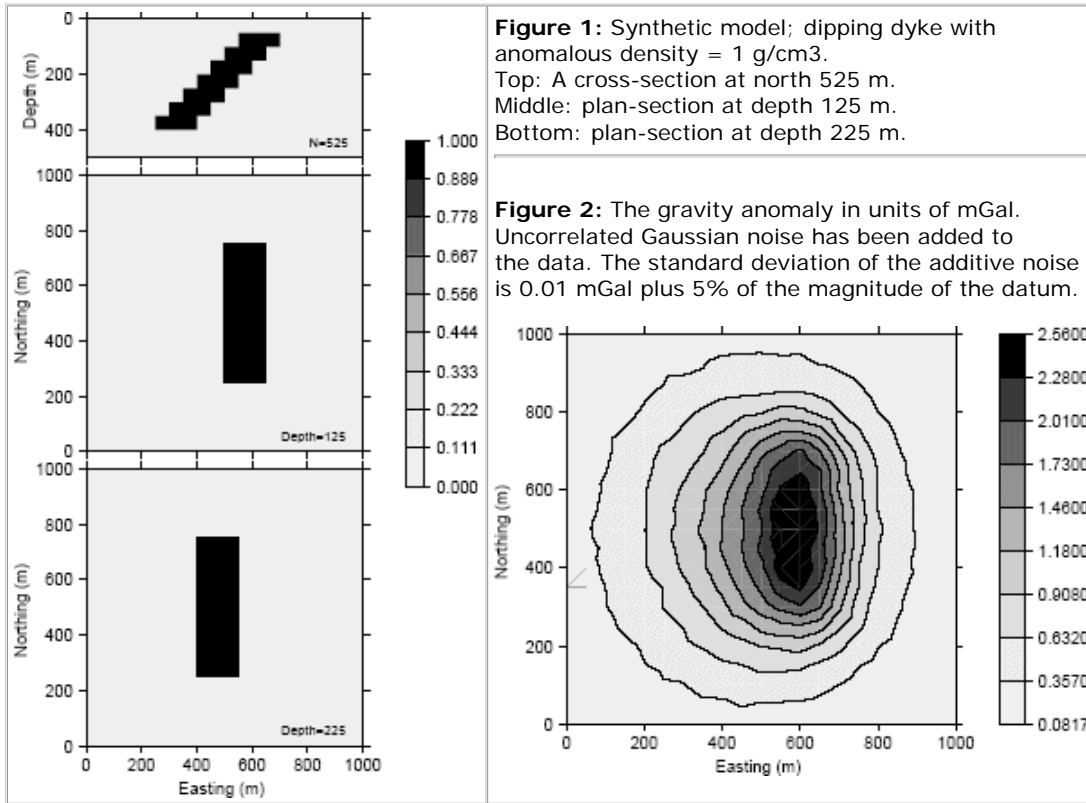
$$\begin{aligned} d_i &\equiv F_z(\vec{r}_{0i}) \\ &= \sum_{j=1}^M \rho_j \left\{ \gamma \int_{\Delta V_j} \frac{z - z_0}{|\vec{r} - \vec{r}_{0i}|^3} dv \right\} \\ &\equiv \sum_{j=1}^M \rho_j G_{ij}. \end{aligned} \quad (2)$$

In eq.(2),  $\rho_j$  and  $\Delta V_j$  are the density and volume of the  $j$ 'th cell,  $d_i$  is introduced as a generic symbol for the  $i$ 'th datum, and  $G_{ji}$ , defined by the expression in brackets, quantifies the contribution of the  $j$ 'th cell to the  $i$ 'th datum. The solution for the integral in eq.(2) can be found in Nagy (1966) and we have adopted the solution by Haaz (1953) here. Expressed in matrix notation, the gravity data consisting of  $N$  observations are given by

$$\mathbf{G}\vec{\rho} = \vec{d}, \quad (3)$$

where  $\vec{d} = (d_1; \dots; d_N)^T$  is the data vector and  $\vec{\rho} = (\rho_1; \dots; \rho_M)^T$  is the vector containing the density values of the  $M$  cells. The program GZFOR3D performs forward modelling calculation to generate the surface gravity data produced by a given density model. Program GZSEN3D calculates the complete matrix  $\mathbf{G}$  to be used in the inversion, with optional wavelet compression applied.

To illustrate the forward modelling program, and to introduce the example used in the inversion section of this manual, we calculate the gravity anomaly on the surface of the earth produced by the density model shown in Fig.1. The model consists of a dipping dyke with anomalous density of  $1.0 \text{ g/cm}^3$  situated in a uniform background. The model is represented by 4000 cells (with 20 in each horizontal direction and 10 in the vertical direction). The anomaly is calculated at a height of 0.5 m above the surface over a 21 by 21 grid with 50 m spacing in both directions. A contour presentation of the data is shown in Fig.2.



### 1.3 Inversion Methodology

The inverse problem is formulated as an optimization problem where an objective function of the density model is minimized subject to the constraints that the data be reproduced to within an error tolerance. The details of the objective function are problem-dependent and can vary according to the available a priori information, but generally the objective function should have the flexibility of constructing a model that is close to a reference model  $\rho_0$  and producing a model that is smooth in three spatial directions. An objective function that accomplishes this is

$$\begin{aligned} \phi_m(\rho) = & \alpha_s \int_V w_s w^2(z) (\rho - \rho_0)^2 dv + \alpha_x \int_V w_x \left( \frac{\partial w(z)(\rho - \rho_0)}{\partial x} \right)^2 dv \\ & + \alpha_y \int_V w_y \left( \frac{\partial w(z)(\rho - \rho_0)}{\partial y} \right)^2 dv + \alpha_z \int_V w_z \left( \frac{\partial w(z)(\rho - \rho_0)}{\partial z} \right)^2 dv \end{aligned} \quad (4)$$



where the functions  $w_s$ ,  $w_x$ ,  $w_y$ , and  $w_z$  are spatially dependent, whereas  $\alpha_s$ ,  $\alpha_x$ ,  $\alpha_y$ , and  $\alpha_z$  are coefficients which affect the relative importance of the different components of the objective function. The greater the ratio  $\alpha_x/\alpha_s$  and so on, the smoother the recovered model is in that axis direction. A length scale of smoothness can be defined for each direction as  $L_x = \sqrt{\alpha_x/\alpha_s}$ ,  $L_y = \sqrt{\alpha_y/\alpha_s}$ ,  $L_z = \sqrt{\alpha_z/\alpha_s}$ ; and specifying greater values for the length scales will produce smoother models.

The objective function in eq.(4) has the flexibility to allow many different models to be constructed. The reference model  $\rho_0$  may be a general background model that is estimated from previous investigations or it could be the zero model. The relative closeness of the final model to the reference model at any location is controlled by the function  $w_s$ . For example, if the interpreter has high confidence in the reference model at a particular region, he can specify  $w_s$  to have increased amplitude there compared to other regions of the model. The weighting functions  $w_x$ ,  $w_y$ , and  $w_z$  can be designed to enhance or attenuate structures in various regions in the model domain. If geology suggests a rapid transition zone in the model, then a decreased weighting for flatness can be put there and the constructed model will exhibit higher gradients provided that this feature does not contradict the data.

The function  $w(z)$  in eq. (4) is a depth weighting that depends upon the model discretization and observation location. The weighting function is used to counteract the decay of the kernel functions,  $G_{ij}$ , with depth. Because of this decay, an inversion which minimizes  $\|\rho - \rho_0\|^2$  subject to fitting the data will generate a density distribution that is concentrated near the surface. We introduce a weighting of the form  $w(z) = (z + z_0)^{\beta/2}$  into the objective function to counteract this effect. The values of  $\beta$  and  $z_0$  are discussed in the next section. They are important in defining the model objective function and thus the type of final model.

The next step in setting up the inversion is to define a measure of misfit. Here we use the 2-norm measure

$$\phi_d = \left\| W_d (d - d^{obs}) \right\|^2. \quad (5)$$

We assume that the noise contaminating the data is independent and Gaussian with zero mean. Specifying  $W_d$  to be a diagonal matrix whose  $i$ 'th element is  $1/\sigma_i$ , where  $\sigma_i$  is the standard deviation of the  $i$ 'th datum, makes  $\phi_d$  a chi-squared random variable distributed with  $N$  degrees of freedom. Accordingly  $E[\phi_d] = N$  provides a target misfit for the inversion.

The inverse problem is solved by finding a density  $\rho(\vec{r})$  which minimizes  $\phi_m$  and misfits the data according to the noise level. This is accomplished by minimizing  $\phi(\rho) = \phi_d + \mu \phi_m$  where  $\mu \in [0, \infty]$  is a regularization parameter that controls the relative importance of data misfit  $\phi_d$  and model objective function  $\phi_m$ . A value of  $\mu$  is sought so that the data are neither fit too well nor too poorly. To perform a numerical solution, we first discretize the objective function in eq.(4) according to the mesh that defines the density model. This yields

$$\begin{aligned} \phi_m(\vec{\rho}) &= (\vec{\rho} - \vec{\rho}_0)^T \left( W_s^T W_s + W_x^T W_x + W_y^T W_y + W_z^T W_z \right) (\vec{\rho} - \vec{\rho}_0) \\ &\equiv (\vec{\rho} - \vec{\rho}_0)^T \left( W_m^T W_m \right) (\vec{\rho} - \vec{\rho}_0) \\ &= \| W_m (\vec{\rho} - \vec{\rho}_0) \|^2 \end{aligned} \quad (6)$$

where where  $\vec{\rho}$  and  $\vec{\rho}_0$  are  $M$ -length vectors. The individual matrices  $W_s$ ,  $W_x$ ,  $W_y$ ,  $W_z$  are straightforwardly calculated once the model mesh and the weighting functions  $w(z)$  and  $w_s$ ,  $w_x$ ,  $w_y$ ,  $w_z$  are defined. The cumulative matrix  $W_m^T W_m$  is then formed. The details of these quantities are specified by the user through the inputs to programs GZSEN3D and GZINV3D.

Since the density contrast is limited to a small range for any practical problems, and often there are well-defined bounds on the density contrast based on direct sampling or other geological information, we also impose constraints to restrict the solution to lie between a lower and upper bound. Thus the solution is obtained by the following constrained minimization problem,

$$\begin{aligned} \text{minimize : } & \phi = \phi_d + \mu \phi_m, \\ \text{subject to : } & \vec{\rho}_{\min} \leq \vec{\rho} \leq \vec{\rho}_{\max}, \end{aligned} \quad (7)$$

where  $\vec{\rho}_{\min}$  and  $\vec{\rho}_{\max}$  are vectors containing the lower and upper bounds on the model values. When the standard deviations of data errors are known, the acceptable misfit is given by the expected value  $\phi_d^*$  and we will search for the value of  $\mu$  that produces the expected misfit. Otherwise, an estimated value of  $\mu$  will be prescribed. The details of various aspects of choosing a regularization parameter will be discussed in a following section.

We use a primal logarithmic barrier method with the conjugate gradient technique as the central solver. In the logarithmic barrier method, the bound constraints are implemented as a logarithmic barrier term. The new objective function is given by,

$$\phi(\lambda) = \phi_d + \mu\phi_m - 2\lambda \sum_{j=1}^M [\ln(\rho_j - \rho_j^{min}) + \ln(\rho_j^{max} - \rho_j)], \quad (8)$$

where is the  $\lambda$  barrier parameter, and the regularization parameter  $\mu$  is fixed during the minimization. As the name suggests, the logarithmic barrier term forms a barrier along the boundary of the feasible domain and prevents the minimization from crossing over to the infeasible region. The method solves a sequence of nonlinear minimizations with decreasing  $\lambda$  and, as  $\lambda$  approaches zero, the sequence of solutions approaches the solution of eq.(7).

The above methodology provides a basic framework for solving 3D gravity inversion with arbitrary observation locations. The basic components are the forward modelling, a model objective function that incorporates a "depth" weighting, a data misfit function, a regularization parameter that ultimately determines how well the data will be fit, and the logarithmic barrier method to obtain the solution with bound constraints. Without getting into the algorithmic details, we discuss three of these basic components in the next sections, namely, the depth weighting, efficient forward mapping, and the choice of the regularization parameter. An understanding of these components is necessary for the user to have a global view of the algorithm and to use the program library correctly.

### 1.4 Depth Weighting

It is well known that gravity data have no inherent depth resolution. As a result, structures will concentrate near the surface when a simple smallest or flattest model is produced, regardless of the true depth of the causative bodies. In terms of model construction, this is a direct manifestation of the kernels' decay with depth. Because of their rapidly diminishing amplitude at depth, the kernels of surface data are not sufficient to generate a function that possess significant structure at depth. In order to overcome this, the inversion needs to introduce a depth weighting to counteract this natural decay. Intuitively, such a weighting will approximately cancel the natural decay and give cells at different depths equal probability of entering the solution with a non-zero density.

For data acquired over a relatively flat surface, the sensitivity decays predominantly in the depth direction. Numerical experiments indicate that a function of the form  $(z+z_0)^{-2}$  closely approximates the kernels' decay directly under the observation point provided that a reasonable value is chosen for  $z_0$ . Having the exponent to be 2 is consistent with the fact that the gravity field decays as inverse distance squared. The value of  $z_0$  can be obtained by matching the function  $(z+z_0)^{-2}$  with the field produced by a column of cells directly beneath the observation point. In accordance with the discretization used in the inversion, we use a depth weighting function of the form

$$w(\vec{r}_j) = \left[ \frac{1}{\Delta z_j} \int_{\Delta z_j} \frac{dz}{(z + z_0)^2} \right]^{1/2}, \quad j = 1, \dots, M \quad (9)$$

for the inversion and  $\vec{r}_j$  is used to identify the  $j$ th cell and  $\Delta z_j$  is its thickness. This weighting function is first normalized so that the maximum value is unity. Numerical tests indicate that when this weighting is used, the density model constructed by minimizing a model objective function in eq.(4), subject to fitting the data, places the recovered anomaly at approximately the correct depth.

For data sets acquired over rugged terrain, the simple depth decay does not describe the dominant variation in sensitivity. Therefore a weighting function that varies in three dimensions is needed. We generalize the depth weighting to form a distance weighting:

$$w(\vec{r}_j) = \frac{1}{\sqrt{\Delta V_j}} \left\{ \sum_{i=1}^N \left[ \int_{\Delta V_j} \frac{dv}{(R_{ij} + R_0)^2} \right]^2 \right\}^{1/4}, \quad j = 1, \dots, M \quad (10)$$

$\Delta V_j$  is the volume of  $j$ th cell,  $R_{ij}$  is the distance between a point in  $\Delta V_j$  and the  $i$ th observation, and  $R_0$  is a small constant used to ensure that the integral is well defined (chosen to be a quarter of the smallest cell dimension). Similarly, this weighting function is normalized to have a maximum value of unity. For inversion of data sets acquired in areas with high topographic relief, it is advantageous to use this more general form of weighting function.

The weighting function is directly incorporated in the sensitivity file generated by program GZSEN3D. This program allows the user to specify whether to use the depth weighting or the distance weighting depending on the terrain of the observed data.

## 1.5 Wavelet Compression of Sensitivity Matrix

The two major obstacles to the solution of large scale gravity inversion problems are the large amount of memory required for storing the sensitivity matrix and the CPU time required for the application of the sensitivity matrix to model vectors. The GRAV3D program library overcomes these difficulties by forming a sparse representation of the sensitivity matrix using a wavelet transform based on compactly supported, orthonormal wavelets. For more details, the users are referred to Li and Oldenburg (1997). In the following, we give a brief description of the method necessary for the use of the GRAV3D library.

Each row of the sensitivity matrix in a 3D gravity inversion can be treated as a 3D image and a 3D wavelet transform can be applied to it. By the properties of the wavelet transform, most transform coefficients are nearly or identically zero. When the coefficients with small magnitude are discarded (the process of thresholding), the remaining coefficients still contain much of the necessary information to reconstruct the sensitivity accurately. These retained coefficients form a sparse representation of the sensitivity in the wavelet domain. The need to store only these large coefficients means that the memory requirement is reduced. Further, the multiplication of the sensitivity with a vector can be carried out by a sparse multiplication in the wavelet domain. This greatly reduces the CPU time. Since the matrix-vector multiplication constitutes the core computation of the inversion, the CPU time for the inverse solution is reduced accordingly. The use of this approach increases the size of solvable problems by nearly two orders of magnitude.

Let  $\mathbf{G}$  be the sensitivity matrix, and  $\mathcal{W}$  be the symbolic matrix-representation of the 3D wavelet transform. Then applying the transform to each row of  $\mathbf{G}$  and forming a new matrix consisting of rows of transformed sensitivity is equivalent to the following operation,

$$\tilde{\mathbf{G}} = \mathbf{G}\mathcal{W}^T, \quad (11)$$

where  $\tilde{\mathbf{G}}$  is called the transformed matrix. The thresholding is applied to individual rows of  $\tilde{\mathbf{G}}$  by the following rule to form the sparse representation  $\tilde{\mathbf{G}}_s$ ,

$$\tilde{g}_{ij}^s = \begin{cases} \tilde{g}_{ij}, & |\tilde{g}_{ij}| \geq \delta_i \\ 0, & |\tilde{g}_{ij}| < \delta_i \end{cases}, \quad i = 1, \dots, N \quad (12)$$

where  $\delta_i$  is the threshold level, and  $\tilde{g}_{ij}$  and  $\tilde{g}_{ij}^s$  are the elements of  $\tilde{\mathbf{G}}$  and  $\tilde{\mathbf{G}}_s$ , respectively.

The threshold level  $\delta_i$  are determined according to the allowable error of the reconstructed sensitivity, which is measured by the ratio of norm of the error in each row to the norm of that row,  $r_i(\delta_i)$ . It can be evaluated directly in the wavelet domain by the following expression:

$$r_i(\delta_i) = \sqrt{\frac{\sum_{|\tilde{g}_{ij}| < \delta_i} \tilde{g}_{ij}^2}{\sum_j \tilde{g}_{ij}^2}}, \quad i = 1, \dots, N. \quad (13)$$

Here the numerator is the norm of the discarded coefficients. For each row we choose  $\delta_i$  such that  $r_i(\delta_i) = r^*$ , where  $r^*$  is the prescribed reconstruction accuracy. However, this is a costly process. Instead, we choose a representative row,  $i_0$ , and calculate the threshold level  $\delta_{i_0}$ . This threshold is then used to define a relative threshold  $\epsilon = \delta_{i_0} / \max_j |\tilde{g}_{i_0j}|$ . The absolute threshold level for each row is obtained by

$$\delta_i = \epsilon \max_j (|\tilde{g}_{ij}|), \quad i = 1, \dots, N. \quad (14)$$

The program that implements this compression procedure is GZSEN3D. The user is asked to specify the relative error  $r^*$  and the program will determine the relative threshold level  $\epsilon$ . Usually a value of a few percent is appropriate for  $r^*$ . For experienced users, the program also allows the direct input of the relative threshold level.

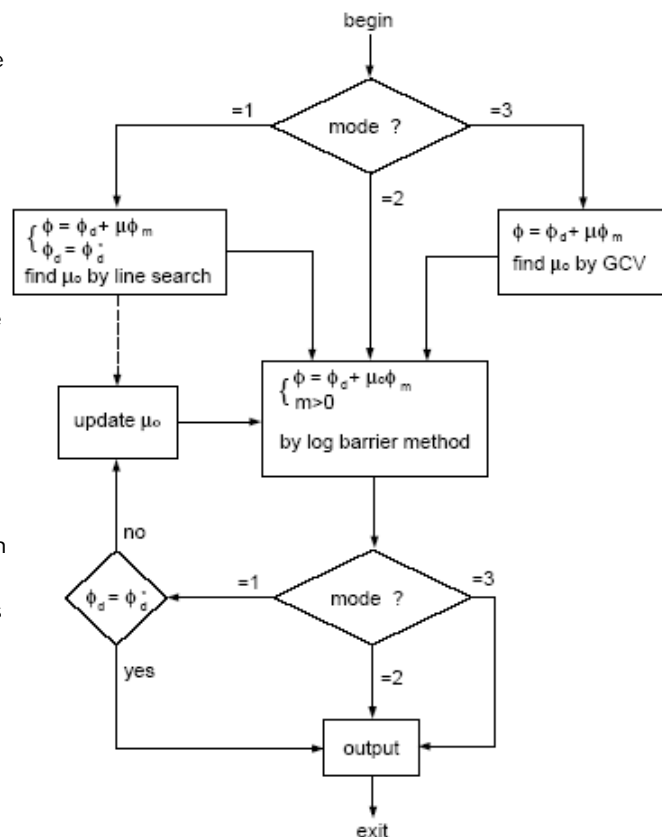
## 1.6 Choice of Regularization Parameter

The choice of the regularization parameter  $\mu$  ultimately depends upon the magnitude of the error associated with the data. The inversion of noisier data requires heavier regularization, thus a greater value of  $\mu$  is required. In this section, we discuss the various implementations for the choice of  $\mu$  in the GRAV3D library.

If the standard deviation associated with each datum is known, then the data misfit defined by eq.(5) has a known expected value  $\phi_d^*$ , which is equal to the number of data when the errors are assumed to be independent Gaussian noise with zero mean. The value of  $\mu$  should be such that the expected misfit is achieved. This entails a line search based on the misfit curve as a function of  $\mu$ . Because of the positivity constraint, our problem is nonlinear. Thus for each  $\mu$  a nonlinear solution using a logarithmic barrier method must be obtained. This is computationally demanding and we therefore have developed the following strategy to reduce the cost.

It is observed that, when plotted on a log-log scale, the misfit curves for 3D inversion with and without bound constraints often parallel each other in the vicinity of the expected misfit. The curve with positivity must lie above the curve without positivity. Therefore, we can first perform a line search without positivity to find a  $\mu_0$  that gives rise to  $\phi_d^*$ . This search also generates the slope,  $s_0$ , of misfit curve at  $\mu_0$ . This process is very efficient and the required CPU time is much smaller compared to the time required for the solution with positivity. We next assume that  $s_0$  can be used to approximate the slope of the misfit curve when the positivity is imposed. A rigorous line search incorporating positivity starts with an initial guess of  $\mu=0.5\mu_0$ . This usually yields a misfit that is very close to the target value. If the misfit is not sufficiently close to  $\phi_d^*$ , however, a new guess for  $\mu$  is obtained which makes use of the approximate slope  $s_0$ . The inversion with updated  $\mu$  can be solved efficiently if the logarithmic barrier algorithm is started from an initial model close to the final solution. That model is obtained by perturbing the solution corresponding to the previous  $\mu$  away from the zero bound. The line search using this strategy is often successful in reaching the target  $\phi_d^*$  after testing two to four values of  $\mu$ . This strategy is implemented in GZINV3D as the first method for choosing the tradeoff parameter.

In practical applications the estimate of data error is often not available. Then the degree of regularization, hence the value of  $\mu$ , needs to be determined based on other criteria. A commonly used method in linear inverse problems is the generalized cross-validation (GCV) technique. The use of GCV in inverse problems with inequality constraints such as positivity is far more involved and numerically expensive to implement. However, applying GCV on the 3D gravity inversion without bounds still produces a reasonable estimate of the data error when the data are dominated by anomalies of the same sign. That error can serve as a starting point for further adjustment by the user based on his or her judgement. Since no other information is assumed, we have chosen to use the value of  $\mu$  obtained in this manner directly in the final inversion, which has bound constraints imposed. In this case, only one logarithmic barrier solution is needed. Numerical tests have indicated that this simplistic use of GCV is in fact surprisingly effective unless the data have a large negative bias or are distributed sparsely. GZINV3D has implemented this approach as the third method for choosing the tradeoff parameter.



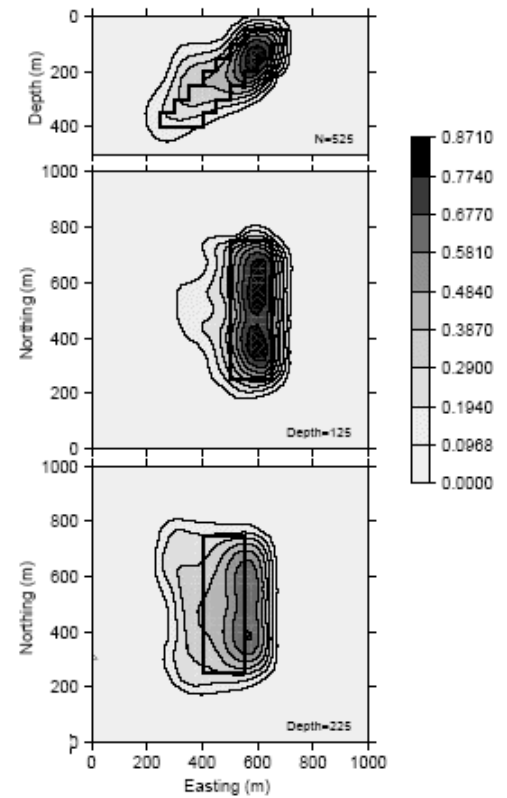
The flowchart shown to the right illustrates the structure of the program GZINV3D. It has three options for determining the tradeoff parameters. The controlling parameter is *mode*. When *mode* = 1, the line search based on known target value of data misfit is used. Two stages, as discussed above, are used, and several solutions for different values of  $\mu$  must be tested to obtain one that produces the target misfit. When *mode* = 2, the user specifies a tradeoff parameter and a single solution is produced. When *mode* = 3, the program first performs GCV analysis on the inversion without positivity and then uses the resultant value of  $\mu$  in the final inversion.

## 1.7 Example Inversion

We now illustrate results of using the program GZINV3D to invert the surface gravity data that was generated for the example in Section 1.2 above. The model region is again discretized into 20 by 20 by 10 cells of 50 m on a side. We invert the 441 noise-contaminated data illustrated in Fig.2 to recover the density contrasts within the 4000 cells.

For the model objective function, we choose  $L_x = L_y = L_z = 100.0$  and unit 3D weighting functions for all components and the default parameters for  $w(z)$ . A zero reference model is used. We have also imposed a lower bound of 0.0 and an upper bound of 2.0.

The final model is shown to the right in Fig. 4. The top panel is a cross-section through the centre of the model and the other two panels are horizontal slices at different depths. The tabular shape of the anomaly and its dipping structure are very clear and the depth extent is reasonably recovered. The dip angle inferred from the recovered model is close to the true value. The amplitude of the recovered model is slightly less than the true value. Over all, however, this recovered model compares favorably with the true model shown in Fig.1.



**Contents of this page:**

[Introduction](#)

- 7 general files:
- [mesh](#): contains the finite difference mesh for the 3D modelling and inversions
  - [topo](#): contains the topographic data of the earth's surface
  - [obs.loc](#): contains the locations for the survey
  - [obs.grv](#): contains the observations
  - [model.den](#): contains the cell values for the density mode
  - [bounds.den](#): specifies upper and lower bounds for density values.
  - [w.dat](#): contains special weightings that alter the type of model produced in the inversions.
- 

## 2.1 Introduction

The GRAV3D library consists of three major programs and one utility:

1. GZFOR3D: performs forward modelling.
2. GZSEN3D: calculates sensitivity and the depth weighting function.
3. GZINV3D: performs 3D gravity inversion.
4. GZPRE3D: multiplies the sensitivity file by the model to get the predicted data. This rarely used utility multiplies a model by the sensitivity matrix in gZinv3d.mtx to produce the predicted data. This program is included so that users who are not familiar with the wavelet transform and the structure of gZinv3d.mtx can utilize the available sensitivity matrix to carry out model studies.

Each of the above programs requires input files, as well as the specification of parameters, in order to run. However, some file are used by a number of programs. Before [detailing the procedures for running](#) each of the above programs, we first present information about these general files.

## 2.2 General Files for GRAV3D Programs

There are seven general files which are used in GRAV3D version 3.0. All are in ASCII text format. **Input** files can have any name you like. Only program **output** files have restricted file names. Also the filename extensions are not important. Many prefer to use the \*.txt filename convention so that files are more easily read and edited in the Windows environment. The files are:

1. mesh: 3D mesh defining the discretization of the 3D model region
2. topo.dat: specifies the surface topography
3. obs.loc: specifies the observation locations
4. obs.grv: specifies the observed gravity anomalies with estimated standard deviation
5. model.den: density model file
6. bounds.den: optional file containing values for upper and lower bounds
7. w.dat: contains the 3D weighting functions

### FILE: mesh

This file contains the 3D mesh which define the model region. mesh has the following structure:

```
NE NN NV
EO NO VO
ΔE1 ΔE2 ... ΔENE
ΔN1 ΔN2 ... ΔNNN
ΔV1 ΔV2 ... ΔVNV
```

Parameter definitions:

- NE Number of cells in the East direction.
- NN Number of cells in the vertical direction.
- NV Number of cells in the North direction.

$E_0, N_0, V_0$  Coordinates, in meters, of the southwest top corner, specific in (Easting, Northing, Elevation). The elevation can be relative, but it needs to be consistent with the elevation used to specify the observation location in `obs.loc` or `obs.den` and in `topo.dat` (see the relevant file for description).

$\Delta_{En}$  Cell widths in the easting direction (from W to E).

$\Delta_{Nn}$  Cell widths in the northing direction (from S to N).

$\Delta_{Vn}$  Cell depths (top to bottom).

The mesh can be designed in accordance with the area of interest and the spacing of the data available in the area. In general, the mesh consists of a core region which is directly beneath the area of available data, and a padding zone surrounding this core mesh. Within the core mesh, the size of the cells should be comparable with the spacing of the data. There is no restriction on the relative position of data location and nodal points in horizontal direction. The cell width in this area is usually uniform.

The maximum depth of the mesh used for inversion should be large enough so that the no density contrast below that depth would produce a noticeable anomaly with the length scale covered by the data area. A rule of thumb is that the maximum depth should be at least half of the longest side of the data area. Based upon the user's knowledge of the survey area, one may adjust the maximum depth as necessary. The cell thickness in vertical direction usually increases slightly with depth. In the shallow region, the ratio of thickness to width of about 0.5 is good, especially when surface topography is present. At depth, a cell thickness close to the cell width is advisable. Once this core mesh is designed, it can be extended laterally by padding with a few cells, possibly of variable width. This padding is necessary when the extracted anomalies are close to the boundary of the core mesh or if there are influences from anomalies outside the area which cannot be easily removed. Problems with more than 20,000 to 30,000 model cells, and/or more than a few thousand data points would be considered large, and can be expected to require a considerable amount of computing memory and time.

The vertical position of the mesh is specified in elevation. This is to accommodate the inversion of a data set acquired over a topographic surface. When there is strong topographic relief and one wishes to incorporate it into the inversion, special care should be taken to design the mesh. A conceptually simple approach is first to design a rectangular mesh whose top (specified by  $V_0$ ) is just below the highest elevation point, and then to strip off cells that are above the topographic surface. This is the approach taken in GRAV3D. The number of cells to be stripped off in each column is determined by the user-supplied topography file [topo.dat](#). Only the remaining cells will be used in the forward modelling or included in the inversion as model parameters.

**Example of the mesh file:** The following is a 10 x 10 x 5 mesh where each cell is 50m by 50m by 50m.

```
10 10 5
0 0 0
50.0 50.0 50.0 50.0 50.0 50.0 50.0 50.0 50.0 50.0
50.0 50.0 50.0 50.0 50.0 50.0 50.0 50.0 50.0 50.0
50.0 50.0 50.0 50.0 50.0
```

#### FILE: `topo.dat`

This optional file is used to define the surface topography of the 3D model by the elevation at different locations. `topo.dat` has the following structure:

```
! comment
!
npt
E1 N1 elev1
E2 N2 elev2
Enpt Nnpt elevnpt
```

Comments: top lines beginning with ! are comments.

`npt` = number of points specifying

$E_i, N_i, elev_i$  = Easting, Northing and elevation of the  $i^{\text{th}}$  point. The elevation in this file and  $V_0$  in the meshfile must be specified relative to a common reference.

The lines in this file can be in any order as long as the total number is equal to `npt`. The topographic data need not be supplied on a regular grid. GRAV3D assumes a set of scattered points for generality and uses triangulation-based interpolation to determine the surface elevation above each column of cells. To ensure the accurate discretization of the topography, it is important that the topographic data be supplied over the entire area above the model and that the

supplied elevation data points are not too sparse.

If topo.dat is not supplied, the surface will be treated as being flat.

#### Example of topo.dat file:

```
!! topographic data
4
  0.0   0.0  50.0
  0.0 1000.0  50.0
1000.0  0.0 -50.0
1000.0 1000.0 -50.0
```

NOTE: Although the cells above the topographic surface are removed from the model, they must still be included in the model file, model.sus, as if they are a part of the model. For input model files these cells can be assigned any value. The recovered model produced by inversion program MAGINV3D also includes the cells that are excluded from the model, but these cells will have a value of -100.0 as identifier

#### FILE: obs.loc

This file is used to specify the observation locations. It is used by the forward modelling code only. The following is the file structure of obs.loc:

```
! comments ...
!
ndat
E1   N1   Elev1
E2   N2   Elev2
Endat Nndat Elevndat
```

#### Parameters are:

! comments top lines beginning with ! are comments.

ndat number of observations.

En, Nn, Elevn easting, northing and elevation of the observation, measured in meters. Elevation should be above the topography for surface data, and below the topography for borehole data. The observation locations can be listed in any order.

Easting, northing and elevation information should be in the same coordinate system as define in the mesh.

**Example of obs.loc file:** We provide an example file below.

#### Example data file

```
! Test data
!
441      !! # of data
0.00   0.00  1.0
0.00  50.00  1.0
0.00 100.00  1.0
:
1000. 00 900.00 1.0
1000. 00 950.00 1.0
1000.00 1000.00 1.0
```

#### FILE: obs.grv

This file is used to specify the observation locations and the observed gravity anomalies with estimated standard deviation. The values of parameters specifying the observation locations are identical to those in obs.loc. The output of the forward modelling program GZFOR3D has the same structure except that the column of standard deviations for the error is omitted. The following is the file structure of obs.grv:

```
! comments ...
!
ndat
E1   N1   Elev1   Grav1   Err1
E2   N2   Elev2   Grav2   Err2
:
```



**Parameters are:**

- ! comments top lines beginning with ! are comments.
- n dat number of observations.
- En, Nn, Elevn easting, northing and elevation of the observation, measured in meters. Elevation should be above the topography for surface data, and below the topography for borehole data. The observation locations can be listed in any order.
- Gravn gravity anomaly data, measured in mGals.
- Errn standard deviation of Gravn. This represents the absolute error. It CANNOT be zero or negative.

**NOTE:** It should be noted that the data Gravn are **extracted anomalies** after standard data reduction and regional removal are applied. A suitable data set is one in which there are just a few data values for each mesh cell at the Earth's surface. Also, it is prudent to upward continue data so they appear as if they were gathered at an elevation roughly equal to half a cell width above the Earth's surface.

**NOTE:** Careful attention to the regional is particularly important for gravity data. This is because any residual anomaly near mesh edges, or which could be caused by features too large to be modelled with the given mesh, must be accounted for in the inversion result, even if the causative geologic materials are in reality outside the mesh. The result will be erroneous density anomalies in the recovered model.

**Example of obs.grv file:**

```
! Examples file
!
441      !! # of data
0.00    0.00 2.00 0.978189E-01 0.293238E-02
0.00    50.00 2.00 0.109359E+00 0.320770E-02
0.00    100.00 2.00 0.122059E+00 0.351328E-02
...
0.00    500.00 2.00 0.236872E+00 0.562092E-02
0.00    550.00 2.00 0.224701E+00 0.556866E-02
```

**FILE: model.den**

This file contains the cell values of the density contrast model. model. The density contrast must have values in gm/cm<sup>3</sup>. The following is the file structure of model.den:

```
den1,1,1
den1,1,2
:
den1,1,NV
den1,2,1
:
deni,j,k
:
denNN,NE,NV
```

Each den<sub>i,j,k</sub> is the density contrast at location [i j k].

[i j k]=[1 1 1] is define as the cell at the top-south-west corner of the model. The total number of lines in this file should equal NN x NE x NV, where NN is the number of cells in the North direction, NE is the number of cells in the East direction, and NV is the number of cells in the vertical direction.

The lines must be ordered so that k changes the quickest (from 1 to NV), followed by j (from 1 to NE), then followed by i (from 1 to NN). If the surface topography (topo.dat) file is supplied, the values above the surface will be ignored. These values should be assigned a large negative value (e.g. -100.0) to avoid confusion with the other model elements.

**FILE: bounds.den**

This file contains the cell values of the lower and upper bounds on the sought density model. It is only required optionally

by gzinv3d. The bounds have the same dimension as the density contrast. The following is the file structure of bounds.den:

```
lb1,1,1    ub1,1,1
lb1,1,2    ub1,1,2
:
lb1,1,NV   ub1,1,NV
lb1,2,1    ub1,2,1
:
lbi,j,k     ubi,j,k
:
lbNN,NE,NV ubNN,NE,NV
```

**Parameters are:**

- lb<sub>i,j,k</sub> is the lower bound on cell [i j k].
- ub<sub>i,j,k</sub> is the upper bound on cell [i j k].

The ordering of the cells is the same as that for model cells: [i j k]=[1 1 1] is defined as the cell at the top-south-west corner of the model. The total number of lines in this file should equal NN NE NV, where NN is the number of cells in the North direction, NE is the number of cells in the East direction, and NV is the number of cells in the vertical direction. The lines must be ordered so that k changes the quickest (from 1 to NV), followed by j (from 1 to NE), then followed by i (from 1 to NN). If the surface topography (topo.dat) file is supplied, the bounds for cells above the surface will be ignored. These values should be assigned a large negative value (e.g. -100.0) to avoid confusion.

**FILE: w.dat**

This file contains the values for a user supplied weighting function. The following is the file structure for w.dat:

```
W.S1,1,1 . . . W.SNN,NE,NV
W.E1,1,1 . . . W.ENN,NE-1,NV
W.N1,1,1 . . . W.NNN-1,NE,NV
W.Z1,1,1 . . . W.ZNN,NE,NV-1
```

**Parameters are:**

- W.S<sub>i,j,k</sub> = cell weights for the smallest model.
- W.E<sub>i,j,k</sub> = cell weights for the interface perpendicular to the easting direction.
- W.N<sub>i,j,k</sub> = cell weights for the interface perpendicular to the northing direction.
- W.Z<sub>i,j,k</sub> = cell weights for the interface perpendicular to the vertical direction.

Within each part, the values are ordered in the same way as in model.den, however, they can be all on one line, or broken up over several lines. Since the weights for a derivative term are applied to the boundary between cells, the weights have one fewer value in that direction. For instance, the weights for the derivative in easting direction has (NE-1)\*NN\*NV values, whereas the number of cells is NE\*NN\*NV.

If the surface topography (topo.dat) file is supplied, the cell weights above the surface will be ignored. These weights should be assigned a value of -1.0 to avoid confusion. If null is entered instead of the file w.dat, then all of the cell weights will be set equal 1.0.

**Contents of this page:**

3 general files: [GZFOR3D](#) performs forward modelling.  
[GZSEN3D](#) calculates sensitivity and the depth weighting function.  
[GZINV3D](#) performs 3D gravity inversion.  
[GZPRE3D](#) multiplies the sensitivity file by the model to get the predicted data.  
Log file: [Log file](#) Explanation of the log file contents.

### 3.1 Introduction

In a MS-WindowsXX operating environment codes are best run using the Graphical User Interface (GUI).

However, all programs in the package also can be run by typing the program name followed by command line arguments. With such a format, they can be executed directly on the command line or in a shell script. When a program is executed without any arguments, it will print a simple message describing the usage. The command format is described below.

**Command format:**

```
PROGRAM arg_1 arg_2 arg_3 ...
```

PROGRAM = name of the executable program. If the program is not in the current directory, its path must be included also.

arg\_n = a command line argument which is the name of a file. It is usually one of those described in the preceding section or a control file containing input parameters.

### 3.2 GZFOR3D

This program performs forward modelling. Command line usage:

```
gzfor3d mesh obs.loc model.den [topo.dat]
```

**Input files :**

All files are in ASCII text format - they can be read with any text editor. Input files can have any name the user specifies.

mesh = the 3D mesh

obs.loc = the observation locations

model.den = the density model

topo.dat = surface topography (optional). If omitted, the surface will be treated as being flat.

**Output file:**

This file is created by GZFOR3D - it's name can NOT be specified.

gzfor3d.grv = the computed gravity anomaly data. Since the data in this file are accurate, the column of the standard deviations for the error is not included.

### 3.3 GZSEN3D

This program performs the sensitivity and depth weighting function calculations. Command line usage:

```
gzsen3d gzsen3d.inp
```

For a sample input file type: `gzsen3d -inp` . Format of the control file `gzsen3d.inp`:

```
mesh
obs.grv
topo.dat
iwt
beta      znot
wvlet
itol      eps
```

**Input files and parameters:**

mesh		3D mesh
obs.grv		data file. Contains the observation locations and the observed gravity anomalies with estimated standard deviation.
topo.dat		surface topography (optional). If null is entered, the surface will be treated as being flat.
iwt		an integer identifying the type of generalized depth weighting to use in the inversion. =1 for depth weighting (only for surface data); =2 for distance weighting (surface - especially when topography is severe - and/or borehole).
beta, znot		parameters defining the depth weighting function. When iwt=1, beta and znot are used as $\beta$ and $z_0$ to define the <b>depth</b> weighting. When iwt=2, beta and znot are used as $\beta$ and $z_0$ to define the <b>distance</b> weighting. If null is entered on this line (line 5), then the program sets beta=2 and calculates the value of znot based upon the mesh and data location. This is true for iwt=1 or 2. For most inversions, however, setting this input line to "null" is recommended. The option for inputting beta and znot is provided for experienced users who would like to investigate the effect of the generalized depth weighting for special purposes. The value of beta should normally be close to 2.0. Note the difference from the value used in MAG3D. Smaller values of give rise to weaker weighting.
wvlet		a five-character string identifying the type of wavelet used to compress the sensitivity matrix. The types of wavelets available are Daubechies wavelet with 1 to 6 vanishing moments (daub1, daub2, and so on) and Symmlets with 4 to 6 vanishing moments (symm4, symm5, symm6). Note that daub1 is the Haar wavelet and daub2 is the Daubechies-4 wavelet. The Daubechies-4 wavelet should be used for most inversions, while the others are provided for users' experimentation. If null is entered, no compression is performed and the program generates a dense matrix in its original form.
itol, eps		an integer and a real number that specify how the wavelet threshold level is to be determined. itol=1: program calculates the relative threshold and eps is the relative reconstruction error of the sensitivity. A reconstruction error of 0.05 is usually adequate. itol=2: the user define the threshold level and eps is the relative threshold to be used. If null is entered on this line, a default relative reconstruction error of 0.05 is used and the relative threshold level is calculated (i.e., itol=1, eps=0.05).

#### Example of gzsen3d.inp control file

```

mesh      ! mesh file
obs.nois  ! data file
null      ! topography
2         ! iwt=1 depth, =2 distance
null      ! beta, znot | null
daub2     ! wavelet type
1 0.05    ! itol, eps | null

```

#### Two output files:

1) gzinv3d.mtx is the sensitivity matrix file to be used in the inversion. This file contains the sensitivity matrix, generalized depth weighting function, mesh, and discretized surface topography. It is produced by the program and it's name is not adjustable. It is very large and may be deleted once the work is completed.

2) A file sensitivity.txt is output after running gzsen3d.exe. It contains the average sensitivity for each cell. This file can be used for depth of investigation analysis or for use in designing special model objective function weighting. (Added for grav3d version 3.0)

### 3.4 GZINV3D

This program performs 3D gravity inversion. Command line usage is:

```
gzinv3d gzinv3d.inp
```

For a sample input file type: gzinv3d -inp.

Format of the control file gzinv3d.inp is as follows:

```

irest
mode
par tol c
obs.mag

```

```

maginv3d.mtx
ini.sus
ref.sus
bounds.sus
LE LN LZ
w.dat
idisk

```

### Control parameters:

- irest** | restarting flag:  
=0: start inversion from scratch.  
=1: restart inversion after it is interrupted. Restart requires two file written out by MAGINV3D before the interruption: maginv3d.aux and maginv3d.kap (see below).
- mode** | an integer specifying one of three choices for determining the tradeoff parameter (see Figure 4 of [background](#)).
- mode=1: the program chooses the tradeoff parameter by carrying out a line search so that the target value of data misfit is achieved.
  - mode=2: the user inputs the tradeoff parameter.
  - mode=3: the program calculates the tradeoff parameter by applying the GCV analysis to the inversion without positivity.
- par,**  
**tolc** | two real numbers that are used differently. Their use depends upon the value of mode.
- mode=1: the target misfit value is given by the product of par and the number of data N, i.e.,. The second parameter, tolc, is the misfit tolerance. The target misfit is considered to be achieved when the relative difference between the true and target misfits is less than tolc. Normally, the value of par should be 1.0 if the correct standard deviation of error is assigned to each datum. When 0.0 is entered for tolc, the program assumes a default value of tolc=0.02.
  - mode=2: par is the user-input value of tradeoff parameter. In this case, tolc is not used by the program.
  - mode=3: none of the two input values are used by the program. However, this line of input still needs to be there.

NOTE: When mode=1 both par and tolc are used. When mode=2 only par is used. When mode=3, neither par nor tolc are used. However, the third line should always have two values.

- L<sub>E</sub>,**  
**L<sub>N</sub>,**  
**L<sub>Z</sub>** | length scales in easting, northing, and depth directions respectively. These parameters determine the weighting coefficients (  $\alpha_s$ ;  $\alpha_x$ ;  $\alpha_y$ ;  $\alpha_z$  ) in the model objective function. The recommended value of the length scale is two to five cell widths in the corresponding direction. If the input sets  $L_E=L_N=L_Z=0.0$ , the inversion will recover a smallest model. If NULL or null is entered, the length scale will be equal to two times the maximum cell width at the centre of the mesh. For example, if the cells are 50m x 50m x 25m at the centre of the mesh, then the default values are  $L_E=L_N=L_Z=100m$ .

- idisk** | parameter which determines how the sensitivity matrix will be accessed.  
=0: sensitivity matrix will be stored in memory. If there is not enough memory, idisk will be set to 1 automatically.  
=1: sensitivity matrix will be accessed from disk when needed. UPDATES FOR Ver4.0 IN THE APPENDIX.

### Example of gzinv3d.inp control file:

The inversion is started from scratch with a zero reference model. The inversion will try to converge to a target misfit equal to the number of data. The sensitivity matrix will be stored in memory.

```

0          !! irest
1          !! mode
1.0 0      !! par, tolc
obs.nois   !! obsf
gzinv3d.mtx !! mtx file
null       !! initial model
0.0        !! reference model
0.0 1.0    !! lower and upper bounds
100, 100, 100 !! alphaS, alphaE, alphaN, alphaZ
null       !! 3D weighting
0          !! idisk

```

### Input files:

Input files can be any file name. If there are spaces in the path or file name, you MUST use quotes (") around the entire

path+filename. Details are in the "[Elements](#)" chapter.

obs.grv		input data file. The file must specify the standard deviations of the error. By definition these are greater than zero.
gzinv3d.mtx		sensitivity matrix and depth weighting functionfile (calculated by GZSEN3D).
ini.den		initial model stored in the same way as model.den. If null is entered, the default value determined from the bounds is used. For a constant initial model, enter a value.
ref.den		reference model stored in the same way as model.sus. If null is entered, the default value of 0.0 is used. For a constant reference model, enter a value.
bounds.den:		lower and upper bounds on the recovered density value stored in the formats bounds.den. If null is entered on this line, the default value of -2.0 and 2.0 are used for all cells. For constant bounds, enter two values for the lower and upper bounds in that order. For example, to construct a model that is bounded between [0, 1.0], this line should be entered 0 1.0.
w.dat		weighting function (optional). If null is entered, the program assumes uniform weight of 1.0 .

### Output files:

Output files are created by the programs. They have fixed file names. Detail file formats are in the "[Elements](#)" chapter.

gzinv3d.log		The log file containing more detailed information for each iteration and summary of the inversion.
gzinv3d.den		Recovered density contrast model.
gzinv3d.pre		The predicted data.
gzinv3d.aux		Auxiliary file listing the data misfit model norm, and Lagrange multiplier at different stages of the inversion. This is used only for the purpose of restarting the inversion.
gzinv3d.rho		Temporary file containing the density contrast model produced at different stages of the inversion. This is used only for the purpose of restarting the inversion.
gzinv3d_nopos.den		This model file is output during the first part of the inversion. It contains the densities without the bounds constraints imposed. (Added for grav3d version 3.0)
gzinv3d_XX.den		These model files are output after each beta iteration. (Added for grav3d version 3.0)

### Log file explanation

The log file gzinv3d.log contains more detailed information about the convergence of the inversion. Depending on how the inversion is set up by the users, the content of the log file is slightly different. In general, there are two stages. In the first stage, the program estimates an approximate regularization parameter. In the second stage, the program performs the inversion with bound constraints using a logarithmic barrier method of minimization, which consists of outer iterations with the barrier parameter and inner iterations of conjugate gradient solution of an linear system within each barrier iteration. The log file information is organized according to these two levels of iterations.

We can refer to the flow chart [Figure 3](#) in the Background chapter to understand the output in the log file. Below we briefly describe the content of the log file according to the parameter mode chosen for the inversion.

#### Mode 1:

In this mode, a user-defined target misfit is supplied by specifying chifact. The first stage of inversion performs a line search without the bound constraints to estimate an approximate value of the regularization parameter and the slope of the misfit curve. The log file identifies this segment and lists summaries of each linear solution such as the number of conjugate gradient (CG) iterations, data misfit, and model norm. The second stage of inversion carries out a number of logarithmic barrier solutions. Each solution corresponding to a single regularization parameter is obtained by a sequence of barrier iterations, and nested inside each barrier iteration is a set of CG iterations. This portion of the log file begins with the value of regularization parameter, initial values of data misfit and model norm. It is then organized in segments corresponding to barrier iterations and lists the number of CG iterations, data misfit, model norm, barrier parameter, and the value of the barrier function at the conclusion of that iteration. As barrier iterations progress, the data misfit, barrier parameter, and barrier function should decrease monotonically. The model objective function may increase or decrease depending on the nature of the initial model of the logarithmic barrier minimization. However, both data misfit and model objective function should plateau at the end.

One or more solutions may be obtained to complete the line search and to achieve the target misfit. Each solution will have a distinct portion in the log file. The trial values of the regularization are dynamically predicted during the inversion and they are not necessarily in any order. Upon completion of line search and the inversion, the log file will list a

summary of the data misfit and model norm corresponding to different values of the regularization parameter sorted in increasing order.

#### **Mode 2:**

In this mode, the user specifies a value of regularization parameter and the program performs a single logarithmic barrier minimization to obtain the solution. The log file mainly consists of the information for one logarithmic solution as described in Mode 1.

#### **Mode 3:**

In this mode, the program first performs a GCV estimate of data noise to obtain an approximate value of the regularization parameter, and it then carries out a single logarithmic barrier minimization to obtain the inverse solution. During the first stage, a number of trial values of the regularization parameter is tested, and two linear systems are solved for each value. Thus the first part of the log file is organized according to the value of regularization parameter. For

each value, the numbers of CG iterations, data misfit, model norm, and GCV value are listed. At the end of GCV search, the log file lists the data misfit, model norm, GCV value according to the regularization parameter sorted in increasing order. The regularization parameter corresponding to the lowest GCV value is used to obtain the final solution in the second stage. The second part of the log file corresponds to a single logarithmic barrier minimization and it is identical to that when mode 2 is chosen.

### **3.5 GZPRE3D**

This utility multiplies a model by the sensitivity matrix in gZinv3d.mtx to produce the predicted data. This program is included so that users who are not familiar with the wavelet transform and the structure of gZinv3d.mtx can utilize the available sensitivity matrix to carry out model studies.

#### **Command line usage:**

```
gzpre3d gZinv3d.mtx obs.loc model.den
```

#### **Input files:**

gZinv3d.mtx = the sensitivityfile from GZSEN3D.

obs.loc = the observation locations.

model.den = the density contrast model.

#### **Output file:**

gzpre3d.mag = predicted data.

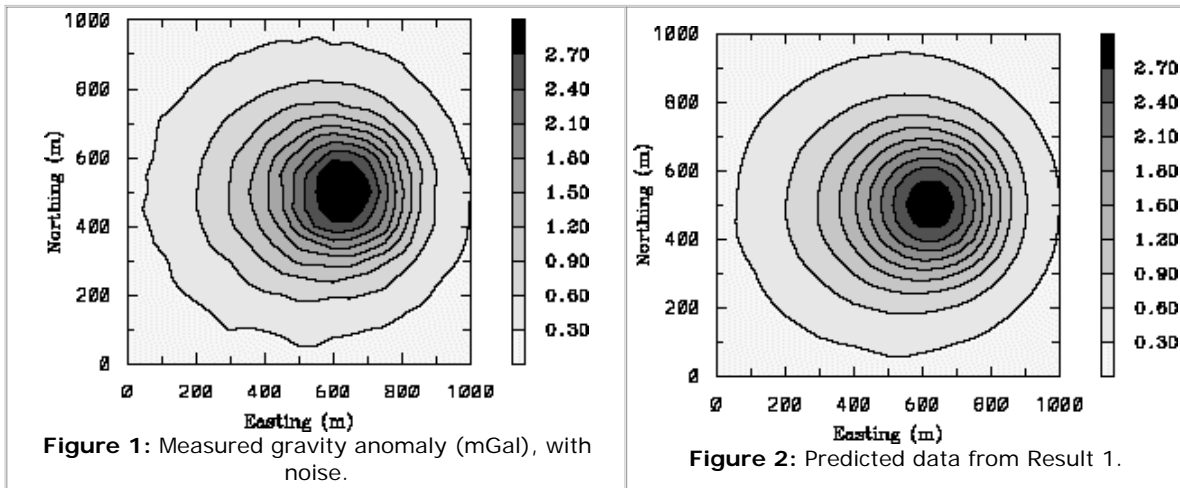
### 4.1 Introduction

This page includes two synthetic examples to illustrate the GRAV3D Library. The first model consists of a dipping slab of positive density contrast buried in a uniform background. This is a relatively small model and we use it to illustrate both the basic operation of the library as well as the incorporation of variable bound constraints. The second model consists of several blocks of different configurations buried in a uniform background. This model has a relatively large number of observations and the number of cells in the inversion is large. It is used to illustrate the utility of wavelet compression in speeding up the inversion of large data sets. For both examples, we calculate the synthetic data from the true model and then add uncorrelated Gaussian noise to simulate noisy observations.

### 4.2 Example 1

Figure 3a below shows one cross-section and two plan-sections of the true model. This model consists of a dipping dyke in a uniform background. The dyke has a westerly dip of  $45^\circ$  and width of 200 by 300 m in the easting and northing directions. It extends from 50 m to 400 m in depth. The density contrast is  $1.0 \text{ g/cm}^3$ .

The observations are simulated above the surface at an elevation of 1 m, and over a 21 by 21 grid with a grid interval of 50 m in both directions. The standard deviation of noise added to the data is 2% plus 0.01 mGal. The noisy data are displayed in Figure 1. It shows a peak that around (600, 500) m and it decays steeply on the east side but slower on the west side due to the dipping structure of the dyke. The small-scale variations reflect the additive noise in the data.



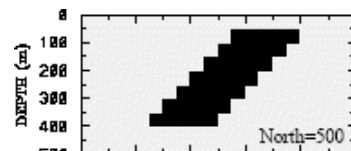
#### Result 1: positive density contrasts

Since the simulated data are entirely positive, it is reasonable to invert for a model consisting of entirely positive density contrast. We accomplish this by imposing a lower bound of 0.0 for positivity and a large upper bound of 4.0. The model objective function has the length scales set to 100 m in all three directions, and the simple depth weighting with default parameters is used. The predicted data from this inversion is shown in Figure 2 above; it is a smooth representation of the noisy data.

The recovered model is shown in one cross-section and two plan-sections in Figure 3b. The model is characterized by a broad density high at the location corresponding to the true dipping dyke and there is clear indication of a dipping structure. The recovered density has a minimum of 0 as constrained by the lower bound of 0. The maximum value is slightly greater than  $1.0 \text{ g/cm}^3$ , which is in fact very close to the true maximum density value in this case. Overall, we have a reasonably good inversion that delineates the essential structure of the true model.

#### Result 2: tighter upper bound constraint

The second inversion result was obtained using a slightly tighter upper bound to illustrate the use of simple upper bound. This is useful when a reliable estimate





of maximum density contrast is available. Imposition of such a bound can often improve the solution. For this inversion, we have set the lower and upper bounds to be 0.0 and 0.8. The maximum value of density contrast in the true model is 1.0. The result is shown in Figure 3c. This model is not very different from the one recovered in the preceding inversion, but the anomalous density appears to be slightly wider. This is to be expected since we now have a smaller density contrast and the required anomalous body should have a large dimension to reproduce the same observed anomalies.

**Figure 3:** True synthetic models, and models recovered by various inversions

- 3a: Synthetic model
- 3b: Recovered model using constraining bounds of [0, 4]
- 3c: Recovered model using constraining bounds of [0, 0.8]
- 3d: Variable bounds for result 3.
- 3e: Recovered model using variable constraining bounds.

**NOTE:** All figures are shown on one page [here](#) (suitable for printing).

### Result 3: variable bounds

One of the new features in the GRAV3D Version 2.0 is the ability to impose variable bounds on the density contrast to be recovered in the inversion. This provides users with one more tool for inputting geological information to improve the inversion. For instance, we may expect a lower density contrast in the region of an orebody than that in another region. Similarly, one region of the subsurface may have a negative contrast while the rest has a positive contrast. In special cases, imposing a lower and upper bounds that are very close to each other will effectively fix the recovered density contrast to a known value during the inversion.

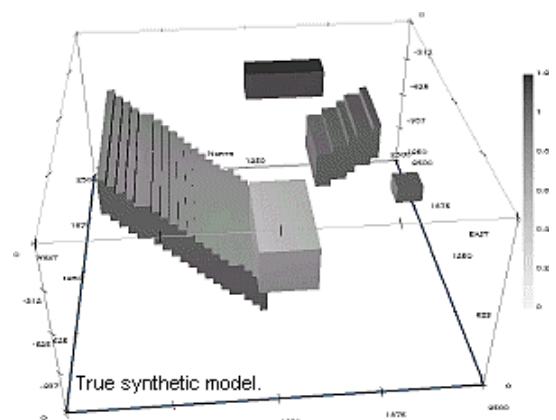
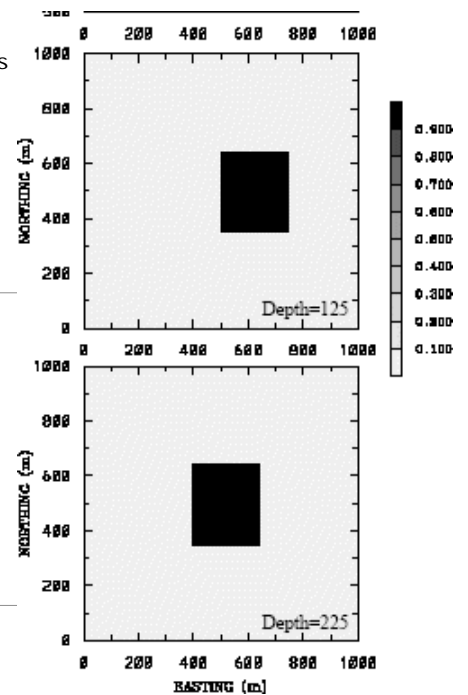
In this example, we illustrate both the variable bounds as well as the use of tight bounds to fix the model values. We invert the data introduced above by imposing a lower bound of zero constant throughout the model, and a variable upper bounds shown in Figure 3d. We assume that the top surface of the dipping dyke is known, but we do not know the lateral extent of the dyke nor its depth extent. Therefore, we impose an upper bound of 0.01 above the dyke and an upper bound of 1.0 below that surface. This effectively constrains the density contrast above in the upper region of the entire model to being very close to zero while allowing the density contrast to vary between 0 and be as high as 1.0 g/cm<sup>3</sup> as required by the data.

The recovered model is shown in Figure 3e. The top surface of the dipping dyke is well imaged as expected because of the imposed bound constraints. However, constraints on the top surface of the dyke has greatly helped image the bottom surface of the dyke. Furthermore, the lateral extent of the dyke is well imaged although we have not constrained it at all. This demonstrates the advantages of imposing specific geologic information through variable bounds that help better define the targets.

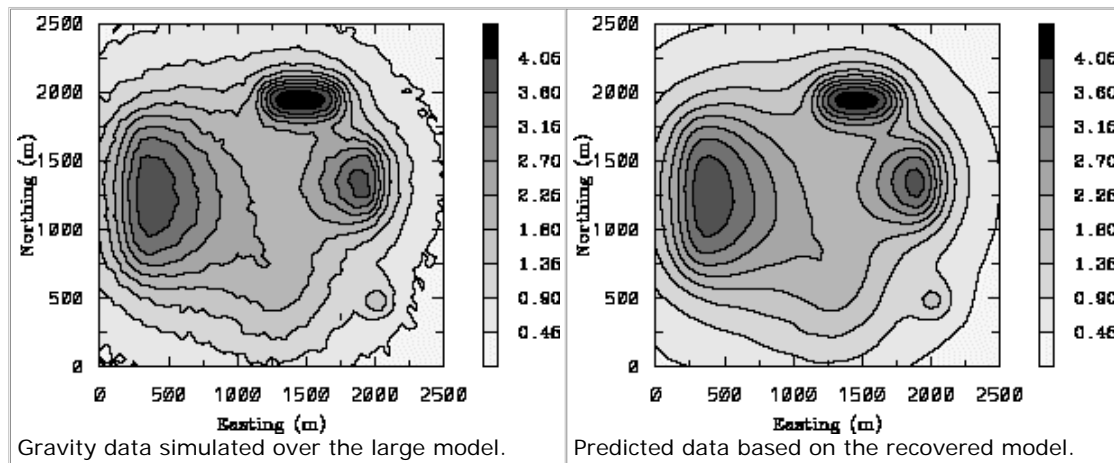
It should be noted that it is possible to "create" nearly any model you like using bounding constraints. They should not be employed unless geological information is very reliable.

### 4.3 Example 2

The figure to the right displays a volume rendered image of the large test model. Move your mouse over to see the recovered model. It consists of five blocks of different density contrasts in a uniform background. There is one large dipping dyke to the left that extends to a large depth. Four smaller blocks of various shapes are located at shallower depths to the right. Three cross-sections of this "true" model are shown via mouseover in last figure below. This figure indicates the vertical lateral locations the five blocks. The model occupies a volume of 2.5 km by 2.5 km by 1.25 km.



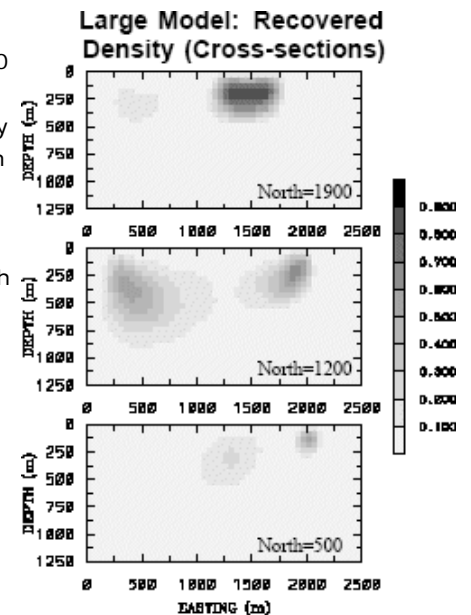
The figure below left shows the gravity data simulated on the surface over a 51 by 51 grid of 50-m grid spacing. A total of 2601 data is generated. Noisy observations were simulated by adding Gaussian random noise. The data plot shows four of the five anomalies and the effect of the added noise.



To invert this data set, we discretize the model region using 50-m cubes. This results in 62500 cells and the corresponding sensitivity matrix requires over 600 Mb to store. We use this example to illustrate the wavelet compression of the GRAV3D version 2.0. Using Daubechies-4 wavelet and a reconstruction accuracy of 5%, a compression ratio of 30 was achieved. The resulting matrix is stored in 42 Mb. With the compressed sensitivity matrix, the inversion was carried out readily without much demand on computer memory or CPU time.

The predicted data from the inversion are shown in the figure above right, which is a smooth version of the observation. The recovered density contrast model is shown via mouseover in the first Example 2 figure above. The cutoff value is  $0.17 \text{ g/cm}^3$ . All five anomalous blocks are imaged.

The recovered model is shown in three crosssections in the figure to the right. Move your mouse over to see the true model. The amplitudes of the recovered anomalous blocks are lower than the true value. The depth extent of the large dipping block is also smaller. This is expected since the area of the data is limited. Over all, this is a good representation of the true model, and the inversion utilizing the wavelet compression has allowed the inversion to be carried out with very moderate demand on computational resources.



**NOTE:** All figures are shown on one page [here](#) (suitable for printing).

Example 1:

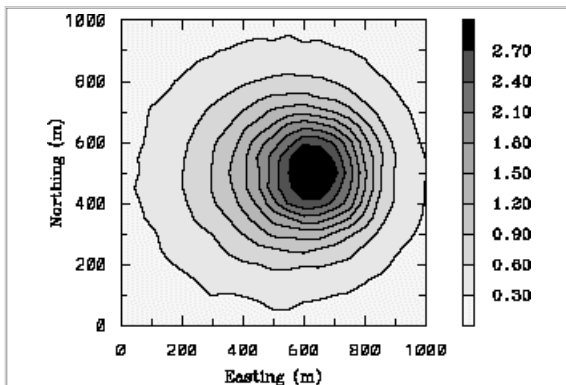


Figure 1: Measured gravity anomaly (mGal), with noise.

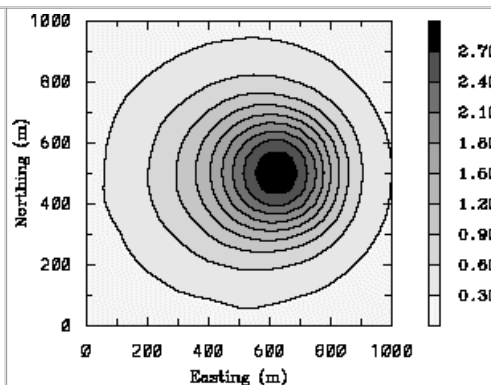
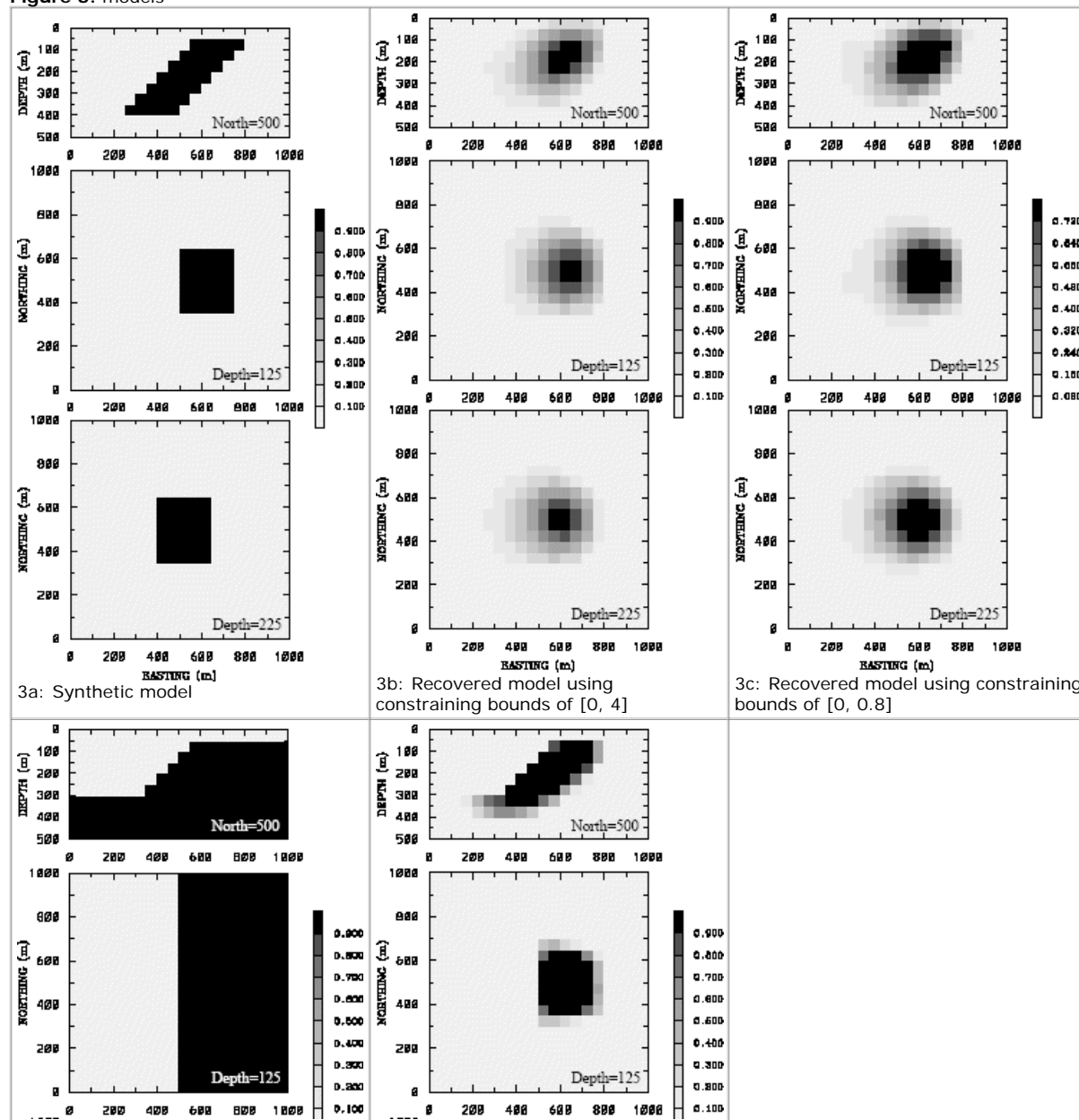
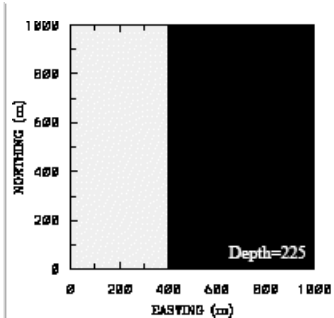


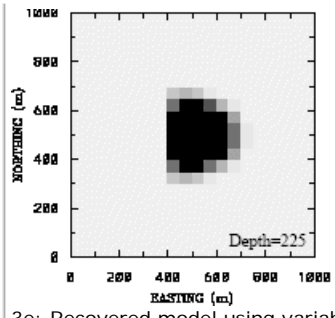
Figure 2: Predicted data from Result 1.

Figure 3: models



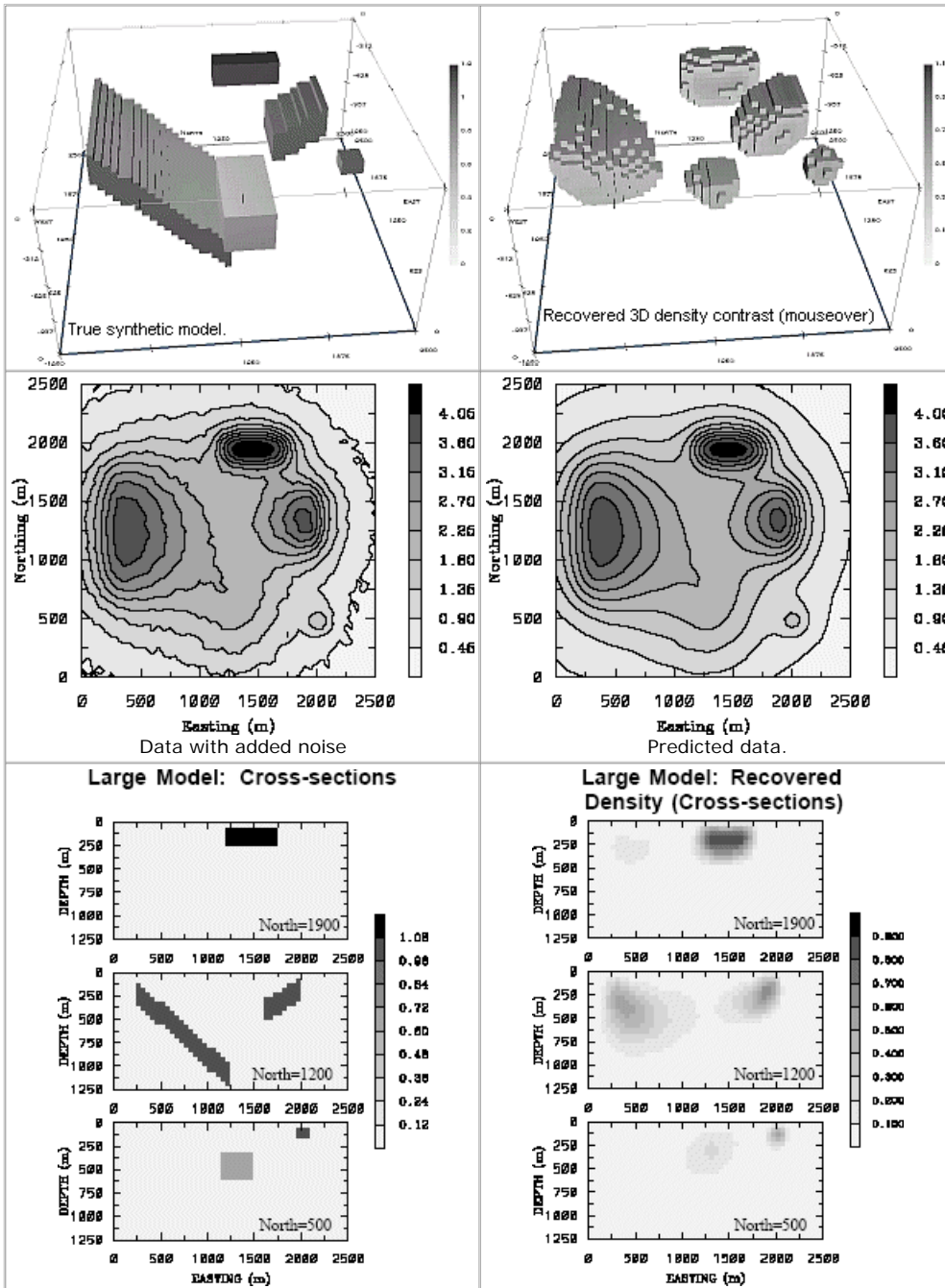


3d: Variable bounds for result 3.



3e: Recovered model using variable constraining bounds.

### Example 2



## **GRAV3D:** *Graphical User Interface*

---

The graphical user interface (GUI) for setting up and running 3D gravity inversions is identical to the GUI for MAG3D, except for units and default values for depth weighting and reference, initial or bounds models. Therefore the instructions for using the MAG3D GUI should answer all questions about using the GRAV3D GUI. [Click here for the MAG3D GUI instructions.](#)

Haaz, I. B., 1953, Relationship between the potential of the attraction of the mass contained in a finite rectangular prism and its first and second derivatives, *Geofizikai Kozlemanyek*, II, No 7.

Li, Y. and Oldenburg, D. W., 1997, Fast inversion of large scale magnetic data using wavelets, 67th Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 490–493.

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

Nagy, D. 1966, The gravitational attraction of a right rectangular prism, *Geophysics*, 31, 361–371.