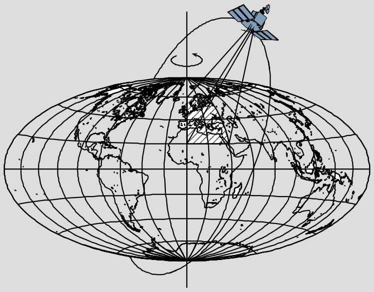
Estimating Parameters of Subsurface Structures from Airborne Gravity Gradiometry Data Using a Monte-Carlo Optimization Method

by

Sibel Uzun



Report No. 506

Geodetic Science

The Ohio State University
Columbus, Ohio 43210

December 2013

Estimating Parameters of Subsurface Structures from Airborne Gravity Gradiometry Data Using a Monte-Carlo Optimization Method

by

Sibel Uzun

Report No. 506

Geodetic Science

The Ohio State University

Columbus, Ohio 43210

December 2013

Abstract

The inverse problem of estimating parameters (i.e, location, depth) of subsurface structures can be considered as an optimization problem where the parameters of a constructed forward model (gravitational model) are estimated from gravimetric observations collected on or above the Earth's surface by minimizing the difference between the predicted model and the observations. This problem could be solved by traditional techniques such as the iterative Least-Squares Solution, or by innovative methods such as random search techniques. This dissertation presents a Monte-Carlo optimization method called Simulated Annealing (SA) to estimate the parameters of subsurface structures from airborne gravity gradient measurements. The SA algorithm is a directed random search technique and is based on Monte-Carlo sampling (Metropolis Algorithm) where the sequence of parameters is generated from a Markov chain with a convergent *target* distribution that depends on a parameter called "temperature". The Metropolis algorithm is applied sequentially to generate a sequence of parameters, at a fixed temperature, and then the temperature is slowly decreased to zero as the SA algorithm proceeds. Reducing the temperature enables the algorithm to narrow its search space, thereby increasingly focusing on the solution that minimizes the cost function. The algorithm is terminated when no further change occurs in the cost function, according to a predefined termination criterion. This technique is implemented for two applications where the parameters of a simulated anomaly, due to a right rectangular prism buried in a real field, and a real anomaly, due to a fault buried in a real field, are estimated. It is shown that the SA algorithm is able to estimate the parameters (location, depth and orientation angle) of the prism from one or more observation profiles of gravity gradients, according to the degree of the noise level of observations that are used. In the second application, the parameters of the real fault are estimated. Here, it is shown that the SA algorithm provides a very good estimate of the dip angle. It is thus shown that the SA algorithm is a robust inversion technique that may be applied to the geophysical inverse problem using gravitational gradients.

Preface

This report was prepared for and submitted to the Graduate School of the Ohio State University as a dissertation in partial fulfillment of the requirements for the PhD degree.

Table of Contents

Chapter 1: Introduction

1.1 Global Optimization Techniques for Inverse Problems

Detecting the shape of a geophysical object under the Earth's surface from surface measurements of the gravity or magnetic field is a geophysical inverse problem that is ill-posed insofar as its solution is non-linear and, most likely, non-unique. Generally, an inversion problem can be considered as an optimization process that looks for a suitable model that best fits certain observed data, by minimizing an error function representing the discrepancy between the survey data and the model description (Montesinos et al., 2005). For nonlinear problems, the solution can often be obtained iteratively by starting from approximate initial values for a linearized model, and its validity basically depends on a good choice of the starting values. Linearized inversion techniques based on gradient approximations and/or matrix inversion may also suffer from some numerical instabilities due to ill-conditioned matrices (Sambridge and Mosegaard, 2002). Moreover, since the inversion of potential field data is an inherently non-unique problem, these mostly local techniques may not always be suitable unless some additional constraints are introduced to reach stable solutions (Montesinos et al., 2005). In this sense, global optimization techniques should be considered among a number of alternatives that may be applied with success to geophysical inverse problems.

In global optimization methods, the solution for an inverse problem is obtained by searching the entire parameter space, for instance by random walk rather than any deterministic mathematical procedure. Once a forward model is established, the inverse problem can be solved by randomly jumping inside the parameter space with trial and error and comparing the subsequent forward model to the observations. Some global optimization techniques choose a particular set of parameters from the parameter space by random walk, compute the discrepancy between the computed forward model and the observations in the form of a cost function or "energy" function, and then move to another set of parameters in the parameter space by random walk. This is continued until the difference between the model and the observed data is minimized (Roy, 2008). These methods do not use derivative information of the cost function. Therefore, they have less possibility of getting trapped in local minima. Since the solution is obtained directly by sampling the parameter space, numerical instabilities due to ill-conditioned matrices are also widely eliminated. Therefore, these techniques are referred to as global optimization tools (Roy, 2008).

The global optimization methods based on random walk sampling in parameter space include Monte-Carlo Inversion, Simulated Annealing, and Genetic Algorithms. The Monte-Carlo inversion is an unguided random walk technique while Simulated Annealing and Genetic Algorithms are guided random walk techniques (Roy, 2008). These methods are also called *heuristic* methods which means that they look for optimal or near-optimal solutions without guaranteeing absolute optimality. One of the drawbacks of heuristic methods is that they are in general computationally expensive. Moreover, a model resolution may not be obtained as easily as in the gradient-based techniques.

1.2 Background

Subsurface detection and estimation is a kind of geophysical inverse problem where one makes inferences about the structure of buried bodies, without accessing them, by using the mathematical tool of inversion. The subsurface detection and estimation problem may vary, depending on the selected model parameters. Model parameters to be estimated could be density contrast or geometric parameters of the subsurface structure (i.e., depth, horizontal coordinates, and shape of the anomalous body), which leads to linear and nonlinear inversion problems, respectively. The methods involving the estimation of density contrasts for elements of a partitioned subsurface can use a linear approach whereas the methods considering the geometric properties of the subsurface structure as unknowns and assuming that the density contrast is known involve nonlinear inversion techniques (Montesinos et al., 2005). The second case can be considered as fitting the response of an idealized or assumed subsurface Earth model to the finite set of actual data collected on the Earth's surface. Here, synthetic data are generated from the idealized or assumed Earth model (forward model) and then a solution of the model parameters of interest is obtained iteratively until an acceptable match is achieved between the observed data and the synthetic data (Sen and Stoffa, 1995).

As noted in the previous section, fundamentally, there are two types of optimization approaches, either a gradient-based approach or a random search (Tarantola, 2005). Monte-Carlo methods are pure random search techniques that pick model parameters uniformly at random in the parameter space and compare the resulting forward model with the observations. The pure Monte-Carlo Search techniques are computationally very expensive since they evaluate all possible models in the parameter space (Sen and Stoffa, 1995). Markov-Chain Monte-Carlo (MCMC) methods are directed search techniques which utilize a probabilistic transition rule to determine the chance of moving from one model parameter to another in the parameter space. They generate sequences of model parameters from a Markov chain in *equilibrium* whose distribution is a certain target distribution. The simulated annealing method uses this MCMC scheme to simulate model parameters iteratively from the target distribution that is controlled by a parameter called *temperature.* As the temperature decreases slowly to zero during the execution of the SA algorithm, the generated values from the Markov chain with target distribution concentrate more and more on the neighborhood of globally minimum solution.

The Simulated Annealing (SA) algorithm is based on the Metropolis algorithm which was originally introduced by Metropolis et al. (1953) to sample from an equilibrium distribution of an interacting particle system. The relationship between this method and any general optimization problem was first proposed by Kirkpatrick et al. (1983) who applied it for multivariate or combinatorial optimization problems (the problem of finding the minimum of a function based on many parameters) in which the global minimum of a given function defined in a *discrete* domain was found. Bohachevsky et al. (1986) presented a generalized SA algorithm for the *continuous* variable problem. Corana et al. (1987) presented SA algorithms which optimize functions having many local minima over an *n*-dimensional continuous parameter space. Goffe et al. (1994) provided extensions and a detailed overview of the implementation of Corana's algorithm and applied it to four econometric problems.

The use of Simulated Annealing (SA) in geophysical inverse problems was first introduced by Rothmann (1985). Since this introduction, the method has been applied to many geophysical inverse problems such as 2-D resistivity inversion using very fast simulated annealing (Chunduru et al., 1996), seismic inversion to estimate two-way travel times and reflection coefficients (Mosegaard and Vestergaard, 1991), inversion of magnetics and resistivity survey data (Dittmer and Szymanski, 1995), airborne electromagnetic inversion (Yin and Hodges, 2007), and one-dimensional seismic waveform inversion (Sen and Stoffa, 1991). Roy et al. (2005) employed a very fast simulated annealing algorithm to both synthetic and real airborne gravity data collected over Lake Vostok, East Antarctica, to estimate the depth of the lake and the thickness of the sediments. Nagihara and Hall (2001) estimated a salt body's thickness by minimizing the difference between the measured and the model-predicted gravity response. They modeled a salt body by a simple geometry consisting of a source volume that was divided into a number of blocks. They tested the SA algorithm using synthetic data with and without noise.

With the work of Kirkpatrick et al. (1983), the Simulated Annealing has been applied to many science and engineering applications such as image restoration (Geman and Geman, 1984), reconstruction of coded images (Smith et al., 1985), and some geodetic applications such as the first-order design of geodetic network (Berne and Baselga, 2004), the second-order design of geodetic networks (Baselga and Asce, 2011), and a global optimization solution of iteratively reweighted least-squares adjustment (robust estimation) (Baselga, 2007). However, no attempt has been made to solve the problem of finding the subsurface structure from real observed airborne gravity gradients using the simulated annealing method. Therefore, in this study, a Simulated Annealing algorithm, based on the work of Corana et al. (1987) is adopted in order to detect and characterize subsurface anomalies from airborne gravity gradient measurements.

In this study, gravity gradients are obtained from an airborne gravity gradiometer survey system. Gravity gradients have much higher sensitivity to short wavelength anomalies than gravitational accelerations since gradients are the second derivatives of the respective field. This property makes gravity gradiometry a suitable technique to detect near-subsurface mass anomalies. The most important advantage of gravity gradiometry is its insensitivity to common mode linear accelerations of the moving vehicle since it measures differences in accelerations. Therefore, unlike in airborne gravimetry, the problem of separating the gravitational signal from the linear accelerations of the vehicle can be avoided (Jekeli, 1988). Furthermore, gravity gradiometry can measure more than one gradient of the several components of the gravity vector, which can lead to better subsurface detection in comparison to single component measurements.

The subsurface estimation problem has been studied extensively using nonlinear or iteratively linearized inversion techniques, and their advantages and limitations have been published in the literature. The main aim of this research is to address some advantages of using the Simulated Annealing method and to show its success in comparison to available inversion techniques in terms of locating and characterizing any subsurface structure from gravity gradients. The algorithm will be tested to detect a simulated anomaly due to a rectangular prism buried in a real field and to estimate parameters of a real anomaly due to a fault buried in a real field. In real world applications, the first example can be considered as characterizing voids which can be modeled as a right rectangular prism. In the second application, the estimation of parameters of a real fault in the Wichita uplift region of southwestern Oklahoma is attempted. Therefore, the second application is important for earthquake studies. Fault detection is also important in hydro-geological studies and in mineral and petroleum exploration since many minerals are associated with faults or cracks in geologic provinces.

1.3 Organization of this Dissertation

This dissertation is designed as follows: Chapter 2 introduces gravitational gradients, and gives a brief overview of airborne gravity gradiometry. Chapter 3 reviews Markov chains in finite state space and explains the theory about how they are used in Markov-Chain Monte-Carlo (MCMC) methods. This is followed by the Metropolis algorithm which is the basis of the Simulated Annealing (SA) method, described in detail for the subsurface anomaly estimation problem. Chapter 4 includes numerical experiments, and Chapter 5 contains conclusions and some comments for future research.

Chapter 2: Airborne Gravity Gradiometry

2.1 Gravitational Gradient

According to Newton's Law of gravitation, the gravitational potential, V, at a point in a Cartesian coordinate system, $\boldsymbol{x} = (x, y, z)$, due to a closed volume v with density ρ, is defined by

$$V(\boldsymbol{x}) = G\iiint_v \frac{\rho(\boldsymbol{x}')}{|\boldsymbol{x}-\boldsymbol{x}'|} dx'dy'dz' , \tag{2.1}$$

where $|\boldsymbol{x}-\boldsymbol{x}'| = \sqrt{(x-x')^2+(y-y')^2+(z-z')^2}$ is the distance between the source point $\boldsymbol{x}'=(x', y', z')$ and the computation point $\boldsymbol{x} = (x, y, z)$, G is Newton's gravitational constant, $G = 6.672\times10^{-11} m^3 s^{-2} kg^{-1}$ (Moritz, 2008). The gravitational acceleration is the gradient of the gravitational potential, and it is a 3×1 vector evaluated at $\boldsymbol{x} = (x, y, z)$,

$$\underline{g} = \nabla V = [\frac{\partial V}{\partial x}, \frac{\partial V}{\partial y}, \frac{\partial V}{\partial z}]^T = [g_x, g_y, g_z]^T = g(\boldsymbol{x}); \tag{2.2}$$

∇ represents the gradient vector operator in Cartesian coordinates. Each component of the vector $\underline{g}$ indicates the acceleration of gravitation in a coordinate direction at the point $\boldsymbol{x}$.

The gradient of the (transposed) gravitational acceleration vector is a 3×3 matrix, called the *gravitational gradient tensor*,

$$\Gamma = \nabla \underline{g}^T = \begin{bmatrix} \frac{\partial g_x}{\partial x} & \frac{\partial g_y}{\partial x} & \frac{\partial g_z}{\partial x} \\ \frac{\partial g_x}{\partial y} & \frac{\partial g_y}{\partial y} & \frac{\partial g_z}{\partial y} \\ \frac{\partial g_x}{\partial z} & \frac{\partial g_y}{\partial z} & \frac{\partial g_z}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{\partial^2 V}{\partial x \partial x} & \frac{\partial^2 V}{\partial x \partial y} & \frac{\partial^2 V}{\partial x \partial z} \\ \frac{\partial^2 V}{\partial y \partial x} & \frac{\partial^2 V}{\partial y \partial y} & \frac{\partial^2 V}{\partial y \partial z} \\ \frac{\partial^2 V}{\partial z \partial x} & \frac{\partial^2 V}{\partial z \partial y} & \frac{\partial^2 V}{\partial z \partial z} \end{bmatrix}^T = \begin{bmatrix} \Gamma_{xx} & \Gamma_{xy} & \Gamma_{xz} \\ \Gamma_{yx} & \Gamma_{yy} & \Gamma_{yz} \\ \Gamma_{zx} & \Gamma_{zy} & \Gamma_{zz} \end{bmatrix}^T = \Gamma(\boldsymbol{x}) . \tag{2.3}$$

With $(x, y, z) \rightarrow (x_1, x_2, x_3)$, the gravitational gradient tensor can be rewritten as

$$\Gamma = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} & \Gamma_{13} \\ \Gamma_{21} & \Gamma_{22} & \Gamma_{23} \\ \Gamma_{31} & \Gamma_{32} & \Gamma_{33} \end{bmatrix}, \qquad (2.4)$$

which is used mostly in later chapters. The diagonal and off-diagonal elements of Γ are the *in-line* and *cross* gradient terms, respectively (Jekeli, 2006). Figure 2.1 shows all tensor components in an East-North-Up (ENU) reference frame.

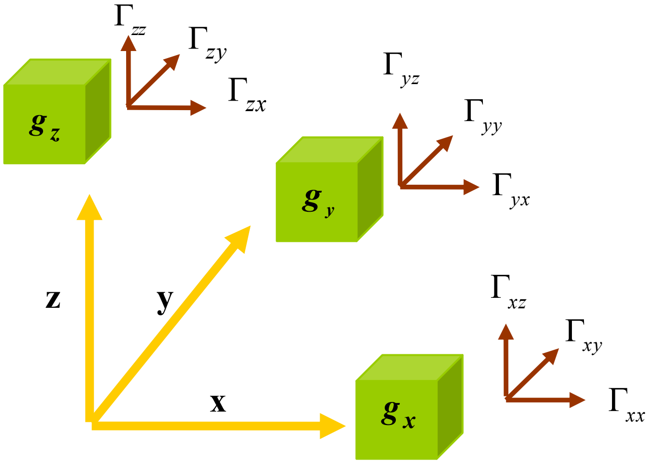


Figure 2.1: Gravitational gradient tensor components in the ENU frame (Bell Geospace, 2008)

The gradient tensor elements are given in units of Eötvös, $1\,\mathrm{E} = 1 \times 10^{-9} s^{-2}$ named after Roland Eötvös who devised the first torsion balance gradiometer in 1890 to perform his gravitational investigations (Szabó, 1998). Each element in the matrix Γ indicates the rate of change of a gravitational vector's component with respect to its position in a coordinate direction (*x*, *y*, *z*). For example, Γ_{zz} is the vertical change of the z component (g_z) of $\underline{g}$ in the z direction. Similarly, Γ_{xy} is the horizontal change of the *x* component (g_x) of $\underline{g}$ in the *y* direction.

The sum of the diagonal elements of Γ satisfies *Laplace's equation* in free space $(\rho = 0)$,

$$\Gamma_{xx} + \Gamma_{yy} + \Gamma_{zz} = 0 . \tag{2.5}$$

Moreover, the gravitational gradient tensor is a symmetric matrix about its diagonal,

$$\Gamma_{xy} = \Gamma_{yx}, \quad \Gamma_{yz} = \Gamma_{zy}, \quad \Gamma_{xz} = \Gamma_{zx} . \tag{2.6}$$

Thus, there are 5 independent gradient tensor elements, two on the diagonal and the three off-diagonal elements in the gravitational gradient tensor matrix. In addition, the gravitational potential, V, satisfies *Poisson's equation* in the Earth's interior (Jekeli, 2006),

$$\nabla^2 V = \Gamma_{xx} + \Gamma_{yy} + \Gamma_{zz} = -4\pi G\rho . \tag{2.7}$$

2.2 Airborne Gravity Gradiometry

The theoretical foundation of moving-based gradiometry naturally depends on Newton's Second Law of Motion that is satisfied in an inertial frame (nonrotating frame), the *i*-frame,

$$\ddot{\underline{x}}^i = \underline{g}^i + \underline{a}^i , \tag{2.8}$$

where $\ddot{\underline{x}}^i$ is the kinematic acceleration of a moving body obtained from the position vector $\underline{x}^i$, $\underline{a}^i$ is the specific force sensed by an accelerometer mounted on the body, and $\underline{g}^i$ is the gravitational acceleration. Here, the superscript indicates in which frame the coordinates of the vectors are given, and each dot represents time derivatives.

Assume that the body contains a set of accelerometers, and rotates and accelerates with respect to the inertial space. Define the body and accelerometer frames, *b*-frame and *a*-frame, respectively, and assume the two frames to be parallel to each other. Assume that the accelerometer is mounted on the body with a displacement or lever arm vector, $\boldsymbol{b}$. Then the translation between the *b*-frame and the *a*-frame is obtained from the position vectors, $\underline{x}^i_{accel}$ and $\underline{x}^i_{body}$,

$$\boldsymbol{b}^i = \underline{x}^i_{accel} - \underline{x}^i_{body} . \tag{2.9}$$

To obtain accelerations of the *b*-frame from sensed accelerations in the *a*-frame, firstly one can take the time derivative of eq. (2.9) twice. The kinematic acceleration in the inertial space is then expressed as

$$\ddot{\underline{x}}^i_{accel} = \ddot{\underline{x}}^i_{body} + \ddot{\boldsymbol{b}}^i . \tag{2.10}$$

Rotation from the *b*-frame to the *i*-frame can be defined by a 3×3 transformation matrix denoted by C_b^i (Jekeli, 2001). This matrix is an orthogonal matrix, and the following two relationships hold for this matrix: $\left(C_b^i\right)^{-1} = \left(C_b^i\right)^T = C_i^b$ and $C_i^b C_b^i = I$, where I is the identity matrix. Rotating the vector $\boldsymbol{b}$ from the *b*-frame to the *i*-frame is defined by $\boldsymbol{b}^i = C_b^i \boldsymbol{b}^b$. Additionally, the first and the second derivative of the vector $\boldsymbol{b}^i$ with the assumption that the accelerometer is fixed to the body ($\dot{\boldsymbol{b}}^b = 0$) are respectively, $\dot{\boldsymbol{b}}^i = \dot{C}_b^i \boldsymbol{b}^b$ and $\ddot{\boldsymbol{b}}^i = \ddot{C}_b^i \boldsymbol{b}^b$.

Inserting the term $\ddot{\boldsymbol{b}}^i = \ddot{C}_b^i \boldsymbol{b}^b$ into the eq. (2.10) and applying eq. (2.8) to the eq. (2.10), we have,

$$\underline{a}^i_{accel} + \underline{g}^i_{accel} = \underline{a}^i_{body} + \underline{g}^i_{body} + \ddot{C}_b^i \boldsymbol{b}^b . \tag{2.11}$$

To obtain accelerations in the *b*-frame, the rotation matrix C_i^b is applied to the eq. (2.11),

$$\underline{a}^b_{accel} = \underline{a}^b_{body} + \underline{g}^b_{body} - \underline{g}^b_{accel} + C_i^b \ddot{C}_b^i \boldsymbol{b}^b , \tag{2.12}$$

where, $\underline{a}^b_{accel}$ is the sensed accelerations in the *a*-frame, but with coordinates in the *b*-frame, $\underline{a}^b_{body}$ is the acceleration of the body, $\underline{g}^b_{body}$ and $\underline{g}^b_{accel}$ are the gravitation of the origin of the *b*-frame and of the location of the accelerometer, respectively, and the last term represents rotation of the *b*-frame with respect to the *i*-frame. Supposing that two accelerometers are differentially close to each other within the body, then the gradient of the acceleration sensed by these two accelerometers is given by

$$\frac{\partial \underline{a}^b}{\partial (\boldsymbol{b}^b)^T} = -\frac{\partial \underline{g}^b}{\partial (\boldsymbol{b}^b)^T} + C_i^b \ddot{(} \quad , \tag{2.13}$$

because the terms $\underline{a}^b_{body}$, $\underline{g}^b_{body}$ and $C_i^b \ddot{C}_b^i$ do not depend on $\boldsymbol{b}^b$ explicitly. Hence, the linear acceleration of the vehicle $\underline{a}^b_{body}$ is removed and the gravitational gradients, $\frac{\partial \underline{g}^b}{\partial (\boldsymbol{b}^b)^T}$ can be obtained from the sensed accelerations by the gradiometer, $\frac{\partial \underline{a}^b}{\partial (\boldsymbol{b}^b)^T}$ and the rotational acceleration, $C_i^b \ddot{C}_b^i$. Here, the derivatives of the vector $\underline{a}$ with respect to $\boldsymbol{b}^T$ is defined by

$$\frac{\partial \underline{a}}{\partial \boldsymbol{b}^T} = \begin{bmatrix} \frac{\partial a_1}{\partial b_1} & \frac{\partial a_1}{\partial b_2} & \frac{\partial a_1}{\partial b_3} \\ \frac{\partial a_2}{\partial b_1} & \frac{\partial a_2}{\partial b_2} & \frac{\partial a_2}{\partial b_3} \\ \frac{\partial a_3}{\partial b_1} & \frac{\partial a_3}{\partial b_2} & \frac{\partial a_3}{\partial b_3} \end{bmatrix}. \tag{2.14}$$

If the platform on which a gradiometer is attached is stabilized in inertial space, then the rotational acceleration term in eq. (2.13) drops out since $C_b^i = 0$. In this case, the gradiometer senses directly gravitational gradients. Otherwise, rotations of the platform with respect to inertial space must be determined by gyroscopes. To obtain the gravitational gradients $\Gamma^b = \partial \underline{g}^b / \partial (\boldsymbol{b}^b)^T$ in a different frame, an additional rotation is required; for example, the gradients in an Earth-fixed frame, the *e*-frame, are obtained by $\Gamma^e = C_b^e \Gamma^b C_e^b$.

Gravity gradiometers were first deployed in aircraft, ships and helicopters. The currently deployed airborne gravity gradiometer instrument (GGI) was developed by Bell Aerospace for the Navy to aid submarine inertial navigation, and tested by the Air Force Geophysical Research Laboratory for the purpose of measuring the Earth's regional gravitational field (Jekeli, 1993). Currently it is operated by Bell Geospace Inc. among others. The GGI consists of two pairs of opposing accelerometers mounted on a rotating disc as illustrated in Figure 2.2.

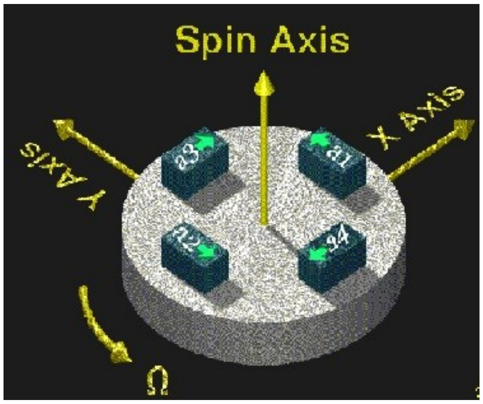


Figure 2.2: Bell Geospace's GGI (Bell Geospace, 2012)

Each pair of accelerometers is placed on the disc in diametrically opposite direction with the sensitive axes being in the plane of the disc and orthogonal to the radius from the center. The input axes of the opposing pairs point in opposite direction so that, when their signal is summed up, the common-mode linear accelerations are eliminated. Taking the difference of the two, such sums of the outputs also eliminate the rotational acceleration about the spin axis. To eliminate imbalance in the scale factor of the accelerometer pairs and their misalignment, the GGI disc is rotated at a frequencey Ω. Rotating the GGI disc at frequency Ω modulates the gradients at twice the rotation frequency since each accelerometer pair measures a particular gradient twice per revolution (Jekeli, 1988). This mechanism allows the gradient signal to modulate at a higher frequency and to eliminate low frequency noise due to individual accelerometers.

Using additional pairs of accelerometers on such a disc will increase the presicion and add some redundancy to the measurements (Jekeli, 2006). As illustrated in Figure (2.3), Bell Geospace's gravity gradiometer consists of three GGI discs that are mounted mutually orthogonally on a local-level stabilized platform such that each GGI makes an angle (35.264°) with the horizon. This structure is named the umbrella configuration.

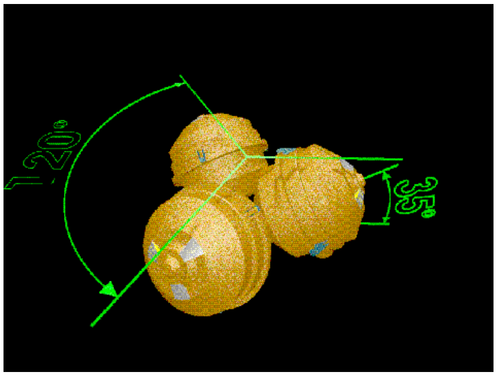


Figure 2.3: Three GGIs (Bell Geospace, 2012)

The umbrella configuration reduces the size of the platform, improves the calibration of the gradiometer and makes it easier to exchange GGIs within the triad. Each GGI senses two outputs. Therefore, three mutually orthogonal discs produce three cross gradient terms $2\Gamma_{12}$, $2\Gamma_{13}$, $2\Gamma_{23}$, and the differences of the in-line gradient terms, $(\Gamma_{11}-\Gamma_{22})$, $(\Gamma_{33}-\Gamma_{11})$, $(\Gamma_{22}-\Gamma_{33})$. The sum of these three differences is zero whether the Laplace's equation is satisfied or not (Jekeli, 1988). Thus, Bell Geospace's gradiometer measures 5 independent tensor elements, three cross and 2 in-line gradients. By utilizing Laplace's equation and the symmetry of the tensor, one can determine the rest of the tensor components (e.g., $(\Gamma_{11}-\Gamma_{22})-(\Gamma_{33}-\Gamma_{11})=2\Gamma_{11}-\Gamma_{22}-\Gamma_{33}=3\Gamma_{11}$).

The improvement of the noise level of the airborne gravity gradiometer has been incremental since most current systems are the modification of the first operational system, tested in 1986 by Bell Aerospace (Zhu and Jekeli, 2009). The noise level for the currently deployed airborne gradiometer platform by Bell Geospace is 2 to 3 E up to

200 m spatial wavelengths detectibility for Γ_{33} with the aircraft speed of 55 m/s (Murphy, 2010).

A full-tensor Gravity Gradiometer is carried on the satellite, GOCE (Gravity Field and Steady-State Ocean Circulation Explorer), launched in 2009. The GOCE gradiometer includes three pairs of opposing three-axis accelerometers which produce all 9 tensor components independently with some redundant measurements. The purpose of the GOCE mission is to measure the Earth's gravitational field with accuracy of ±1 mGal and also to determine the geoid with accuracy of ± (1-2) cm at a spatial resolution better than 100 km (ESA, 1999). Due to the much coarser spatial resolution of the data compared to airborne data, the GOCE gradients are not considered in this study, but could be used in inversion problems associated with larger and deeper subsurface structures at lithospheric scales.

Chapter 3: Markov-Chain Monte-Carlo Method

3.1 Introduction to Markov-Chain Monte-Carlo

The Monte-Carlo methods are broadly defined by a branch of mathematical statistics that deals with experiments that make use of random numbers to investigate a certain problem (Hammersley and Handscomb, 1964). The Monte-Carlo methods were developed by the research scientists S. Ulam and J. von Neumann in the Los Alamos National Laboratory in New Mexico in the late 1940s. The name "Monte Carlo" was suggested by N. Metropolis, and used later in the title of the first published Monte-Carlo paper by Ulam and Metropolis in 1949 (Metropolis, 1987; Eckhardt, 1987; Brooks et al. 2011).

The first Markov-Chain Monte-Carlo (MCMC) technique, which is known as the Metropolis algorithm today, was published by Metropolis and his coworkers in 1953. They introduced a *modified* Monte-Carlo method that could overcome the problem of sampling in high-dimensional spaces encountered by regular Monte Carlo methods (Robert and Casella, 2004). The idea of the method is to construct a Markov chain whose stationary distribution is the *target* distribution from which to sample from. Since samples are generated sequentially with the property that the drawn sample only depends on the realization of the last sample generated (Gelman et al., 2004), the sequence of points generated is not independent anymore but instead forms a stochastic process named a *Markov Chain*. If the chain is run for a sufficiently long time, drawn samples from the chain can be considered as a dependent sample from the target distribution and used to calculate important characteristics of the target distribution (Brooks, 1998). The Metropolis algorithm is one option to implement the Monte-Carlo method. Other popular methods to implement the Monte-Carlo method based on Markov chains include the Metropolis-Hastings algorithm (Hastings, 1970) and the Gibbs Sampler (Geman and Geman, 1984).

Another application of Monte-Carlo methods is the optimization in which a set of model parameters is found that best fit observed data. Simulated Annealing (SA) is such a Monte-Carlo optimization technique, developed to find a global minimum of a function which may have many local minima. It can be seen as a sequence of applying the Metropolis algorithm where a scale parameter called "*temperature*" is introduced in the target distribution and allowed to decrease slowly to zero according to a "cooling" schedule as the SA algorithm proceeds. Applying the Metropolis algorithm sequentially at decreasing values of the temperature, effectively narrows the parameter space of the target distribution, thus leading to optimal estimates. The iteration is terminated when a predefined cost function exhibits no significant change.

Since in the Metropolis algorithm, the generation of successive states can be explained by a time-homogeneous Markov chain theory, the next section presents this theory briefly and discusses the conditions under which a Markov chain converges to samples from the desired target distribution.

3.2 Markov Chains

Hastings (1970) indicates that the simulation of a Markov chain based on Markov-Chain Monte-Carlo methods using pseudo-random number generators in a computer is a finite state-space Markov chain (Robert and Casella, 2004). So, this section reviews Markov chains defined on a discrete/finite state-space and explains their properties that are needed for convergence to the desired target distribution. The theory of the discrete-time Markov chains can also be extended in continuous time state space. In this case, the properties of Markov chains that are needed to converge to the stationary distribution will be similar to the ones for the discrete-time state-space (see Tierney, 1994; 1996 and Robert and Casella, 2004). The concept of stationary distribution and detailed balance remain the same as for the discrete-time state-space (see Gilks et al., 1996).

A *Markov chain* is a special kind of discrete-time stochastic process, which involves characteristic sequences of random variables (Gamerman and Lopes, 2006). This special discrete-time *stochastic process* can be defined as a collection of random variables $\{X^{(n)} : n \geq 0\}$ take values at discrete times, n=0,1,2,3,…., from a *finite/discrete* state-space $S = \{\Xi_0, \Xi_1, ..., \Xi_k\}$. Such a discrete-time stochastic process is called a *Markov chain* if the future state only depends on the given present state of the process, and not on its past states (Trivedi, 1982),

$$\Pr(X^{(n)} = \Xi_j \big| X^{(n-1)} = \Xi_{i_{n-1}}, ..., X^{(0)} = \Xi_{i_0}) = \Pr(X^{(n)} = \Xi_j \big| X^{(n-1)} = \Xi_{i_{n-1}}), \qquad (3.1)$$

where j, $i_l \in \{0, ..., k\}$, l=0,……..,n-1. A discrete-time Markov chain is called *stationary* or *homogeneous* in time if the probability of moving from any state Ξ_i to any other state Ξ_j in one step does not depend on the time n when the step is being made (Isaacson and Madsen, 1985). For a homogeneous Markov chain, one-step transition probabilities are defined as (i, j=1,….,k)

$$p_{ij} = p_{ij}^{(1)}(\Xi_i \to \Xi_j) = \Pr(X^{(n)} = \Xi_j \big| X^{(n-1)} = \Xi_i), \qquad n \geq 1. \qquad (3.2)$$

All probability moves made between states in *S* can be recorded in a matrix, *P*. The matrix *P* includes all information about the movement made between states in *S*. It is called the *transition probability* matrix *P* and its (i,j)-th element is given by p_{ij} (Isaacson and Madsen, 1985). The one-step transition matrix *P* with $(k+1)^2$ elements is defined as

$$P = \begin{bmatrix} p_{00} & \cdots & p_{0k} \\ \vdots & & \vdots \\ p_{k0} & \cdots & p_{kk} \end{bmatrix}. \tag{3.3}$$

The transition matrix *P* is doubly-stochastic since its elements satisfy the following two properties:

$$0 \le p_{ij} \le 1 \tag{3.4a}$$

and the normalization conditions (Isaacson and Madsen, 1985),

$$\sum_i p_{ij} = 1 \text{ and } \sum_j p_{ij} = 1 \tag{3.4b}$$

Eq. (3.4) means that any change in the state of a variable must be in the state space, *S*, almost surely, i.e., with probability equal to 1. An *n*-step transition probability of a homogeneous Markov chain is the probability of moving from any state Ξ_i to any other state Ξ_j in *n* steps,

$$p_{ij}^{(n)} = p^{(n)}(\Xi_i \to \Xi_j) = \Pr(X^{(n)} = \Xi_j \big| X^{(0)} = \Xi_i), \qquad n \ge 1 \tag{3.5}$$

and the *n*-step transition probability matrix is denoted by $P^{(n)}$. For the homogeneous Markov chain, the transition probability matrix does not change with time (Stewart, 2009),

$$p_{ij} = \Pr(X^{(1)} = \Xi_j \big| X^{(0)} = \Xi_i) = \Pr(X^{(2)} = \Xi_j \big| X^{(1)} = \Xi_i) = \Pr(X^{(3)} = \Xi_j \big| X^{(2)} = \Xi_i) = ... \tag{3.6a}$$

If the two step transition probability matrix is computed from the law of total probability,

$$P^{(2)} = p_{ij}^{(2)} = \Pr(X^{(2)} = \Xi_j \big| X^{(0)} = \Xi_0) = \sum_i \Pr(X^{(2)} = \Xi_j, X^{(1)} = \Xi_i \big| X^{(0)} = \Xi_0) =$$
$$= \sum_i \Pr(X^{(2)} = \Xi_j \big| X^{(1)} = \Xi_i, X^{(0)} = \Xi_0) \Pr(X^{(1)} = \Xi_i \big| X^{(0)} = \Xi_0) = \qquad (3.6b)$$

and from the Markov chain property in eq. (3.1), the eq. (3.6b) becomes

$$\begin{aligned} P^{(2)} &= \sum_i \Pr(X^{(2)} = \Xi_j \big| X^{(1)} = \Xi_i) \Pr(X^{(1)} = \Xi_i \big| X^{(0)} = \Xi_0) \\ &= \sum_i p_{ij} p_{0i} = P \times P = P^2. \end{aligned} \qquad (3.6c)$$

The above equation can be generalized for the *n*-step transition probability matrix. Therefore, it follows that for the homogeneous Markov chain, the *n*-step transition probability matrix is the multiplication of the one-step transition probability by itself *n* times, $P^{(n)} = P \times P \times .... \times P = P^n$, for $n \geq 1$. Additionally, $P^{(0)} = P^0 = I$ is the identity matrix.

Let $p_j^{(n)}$ be the probability that a state Ξ_j occurs at time step *n* and is defined by $p_j^{(n)} = \Pr(X^{(n)} = \Xi_j)$. Then, $p_j^{(n)}$ is the component of the state probability row vector,

$$\mathbf{p}^{(n)} = \{p_0^{(n)}, p_1^{(n)}, ..., p_k^{(n)}\}. \qquad (3.7)$$

From the total probability law, the state probability vector at time step *n* can be computed as follows,

$$p_j^{(n)} = \sum_i \Pr\{X^{(n-1)} = \Xi_i\} \Pr\{X^{(n)} = \Xi_j \big| X^{(n-1)} = \Xi_i\} = \sum_i p_i^{(n-1)} p_{ij}. \qquad (3.8)$$

In matrix notation, the equation (3.8) is expressed as

$$\mathbf{p}^{(n)} = \mathbf{p}^{(n-1)} P. \qquad (3.9)$$

If $\mathbf{p}^{(0)}$ is the initial state probability vector defined by $\mathbf{p}^{(0)} = [p_1^{(0)}, p_2^{(0)}, ..., p_k^{(0)}]$, then

$$\begin{aligned} \mathrm{p}^{(1)} &= \mathrm{p}^{(0)} P , \\ \mathrm{p}^{(2)} &= \mathrm{p}^{(1)} P = (\mathrm{p}^{(0)} P) P = \mathrm{p}^{(0)} P^2, \end{aligned} \qquad (3.10)$$

and after iterating eq. (3.10), the following expression is obtained:

$$\mathbf{p}^{(n)} = \mathbf{p}^{(0)} P^{n}, \tag{3.11}$$

or in summation form,

$$p_j^{(n)} = \sum_i p_i^{(0)} p_{ij}^{(n)}, \tag{3.12}$$

which indicates that the probability distributions of a homogeneous Markov chain can be computed from one-step transition probabilities and the initial probability vector $\mathbf{p}^{(0)}$ (Trivedi, 1982).

A basic concern with Markov chains in a simulation problem is their asymptotic behavior as the number of steps or iterations gets larger, $n \to \infty$. A key concept is the *stationary* distribution, π (Gamerman and Lopes, 2006). To ensure that the generated states of a Markov chain eventually come from a stationary distribution, which is also our target distribution, the chain should satisfy three conditions: Firstly, the chain has to be *irreducible*, meaning that it must be possible to move from any state Ξ_i to any other state Ξ_j in a finite number of steps with nonzero probability. Secondly, the chain should be *aperiodic*. A Markov chain is called *aperiodic* if the maximum common divider of all possible numbers of steps it takes for the chain to come back to the starting point is equal to 1 (Liu, 2002). Finally, the chain must be *positive recurrent*. When the Markov chain is irreducible and includes only a finite number of states, then these states become positive recurrent. If a Markov chain is irreducible, finite and aperiodic, then it becomes an ergodic chain (Stewart, 2009).

An *n*-step transition probability $p_{ij}^{(n)}$ of a finite, irreducible and aperiodic Markov chain does not depend on either *n* or *i* as $n \to \infty$. In this case, the limiting state probabilities exist and are expressed by (Trivedi, 1982)

$$\pi_j = \lim_{n \to \infty} p_{ij}^{(n)}, \tag{3.13}$$

which implies that P^n converges independently of the initial starting distribution to a matrix Π with identical rows $\boldsymbol{\pi}$ as *n* gets larger. π_j is an element of the *steady-state* or *equilibrium* vector $\boldsymbol{\pi}$ for the Markov chain, satisfying the following two properties:

$$\pi_j \geq 0, \quad \sum_j \pi_j = 1. \tag{3.14}$$

Clearly, $\boldsymbol{\pi}$ satisfies the following property

$$\lim_{n\to\infty} p_j^{(n)} = \pi_j = \lim_{n\to\infty} p_j^{(n-1)}; \tag{3.15}$$

therefore, by inserting $p_j^{(n)}$ from equation (3.8) into the above equation,

$$\pi_j = \lim_{n\to\infty} \sum_i p_i^{(n-1)} p_{ij} = \sum_i (\lim_{n\to\infty} p_i^{(n-1)}) p_{ij}\ , \tag{3.16}$$

we have,

$$\pi_j = \sum_i \pi_i p_{ij}\ . \tag{3.17}$$

The above equation can be written in matrix form as,

$$\boldsymbol{\pi} = \boldsymbol{\pi} P\,, \tag{3.18}$$

which states that $\boldsymbol{\pi}$ is an eigenvector of P with eigenvalue 1. Any vector fulfilling eqs. (3.18) and (3.14) is called a *stationary* probability vector of the Markov chain (Trivedi, 1982). It should be noted that for a finite and irreducible Markov chain, there exists a unique stationary distribution. If the Markov chain is additionally aperiodic, then this stationary distribution becomes also the unique steady-state distribution of the Markov chain.

Since also $\pi_j = \sum_i \pi_i p_{ij}$ from eq. (3.17), we have with eq. (3.4b)

$$\sum_i (\pi_j p_{ji} - \pi_i p_{ij}) = 0\,. \tag{3.19}$$

A sufficient condition for this equality is the *detailed balance* (*microscopic reversibility*) condition (Wood and Parker, 1957),

$$\pi_i p_{ij} = \pi_j p_{ji}\,. \tag{3.20}$$

Therefore, the problem of creating a Markov chain with a given stationary distribution becomes equivalent to find transition probabilities p_{ij} that satisfy the detailed balance condition (Gamerman and Lopes, 2006). The detailed balance condition does not uniquely determine the transition probabilities. One way of constructing transition probabilities with desired properties is to use the Metropolis, respectively the Metropolis-Hastings algorithm, which is presented in the next sections.

3.3 Metropolis-Hastings Algorithm

The Metropolis algorithm solves the inverse problem of the usual Markov chain problem where the transition probabilities are known and the equilibrium distribution is sought. Here a particular equilibrium distribution is desired or known, and the transition probabilities need to be determined (Wood, 1968).

The transition probabilities are chosen in such a way that, as the number of generated states increases, the distribution of the states produced by this Markov chain converges towards the given target distribution. To achieve this, it is sufficient to satisfy the detailed balance condition given in eq. (3.20). The transition probabilities are chosen in the form of

$$p_{ij} = q_{ij}\alpha_{ij} \qquad \text{for} \quad i \neq j \tag{3.21}$$

where q_{ij} indicates a value of an arbitrary irreducible *proposal distribution* with $\sum_j q_{ij} = 1$ and is a conditional probability that the state Ξ_j is generated from the given current state, Ξ_i. α_{ij} denotes an *acceptance probability*. The proposal distribution values, q_{ij}, can be used as transition probabilities in eq. (3.20), but it may not satisfy the detailed balance condition. For example, if we have the following inequality,

$$\pi_i q_{ij} > \pi_j q_{ji}, \tag{3.22}$$

the acceptance probability can be used to adjust q_{ij}. The choice of the acceptance probability is determined by the following argument: The above inequality indicates that transitions from the state Ξ_i to the state Ξ_j are made more often than in the other direction. Therefore, α_{ji} should be set as large as possible, and it can be at most 1 since it is a probability. Then, since α_{ij} is determined by requiring that p_{ij} satisfies the detailed balance condition, we have

$$\begin{aligned} \pi_i q_{ij}\alpha_{ij} &= \pi_j q_{ji}\alpha_{ji} \\ &= \pi_j q_{ji}, \end{aligned} \tag{3.23}$$

which implies that $\alpha_{ij} = \pi_j q_{ji} / \pi_i q_{ij}$. Similarly, one can consider the case where the inequality in eq. (3.22) is reversed to derive α_{ji} (Chib and Greenberg, 1995). As a result, α_{ij} is set to

$$\alpha_{ij} = \alpha(\Xi_i \to \Xi_j) = \begin{cases} \min\left[1, \dfrac{\pi_j q_{ji}}{\pi_i q_{ij}}\right], & \text{if } \pi_i q_{ij} > 0 \\ 1, & \text{if } \pi_i q_{ij} = 0. \end{cases} \tag{3.24}$$

This acceptance probability is called the Metropolis-Hastings (M-H) acceptance probability (Hastings, 1970), and simulations following this scheme use the so-called M-H algorithm. As can be seen in eq. (3.24), the computation of the acceptance probability depends on the ratio, π_j / π_i, so the normalization constant of the desired distribution, π, does not need to be known (Hastings, 1970). The off-diagonal elements of the Metropolis-Hastings transition matrix are (Tierney, 1994)

$$p_{ij} = q_{ij}\alpha_{ij}, \qquad \text{if } i \neq j \tag{3.25a}$$

and from eq. (3.4), we have $p_{ii} = 1 - \sum_{j \neq i} p_{ij} = 1 - \sum_{j \neq i} q_{ij}\alpha_{ij}$, or explicitly

$$p_{ii} = q_{ii} + \sum_{j \neq i} q_{ij}(1 - \alpha_{ij}). \tag{3.25b}$$

Thus, p_{ij} can be expressed as

$$p_{ij} = q_{ij}\alpha_{ij} + \delta_i(j)\left(\sum_{l \neq i} q_{il}(1 - \alpha_{il})\right), \tag{3.26}$$

where $\delta_i(\cdot)$ denotes the *Dirac*-function on $\{0, \ldots\ldots, k\}$; it is 1 when $i = j$, otherwise it is zero. It is noted that $\alpha_{ii} = 1$, as obtained from eq. (3.24). The off-diagonal term indicates the probability of proposing a new candidate and accepting it. The diagonal term denotes the probability of remaining in the current state either due to the probability of rejecting the proposed candidate, $\sum_{l \neq i} q_{il}(1 - \alpha_{il})$, or of no move being made, q_{ii}. To show that π is the stationary distribution for the generated Markov chain, it is sufficient to show that, for any pairs of states $\Xi_j \neq \Xi_i$, the following equality holds (Green, 1995)

$$\pi_i q_{ij}\alpha_{ij} = \pi_j q_{ji}\alpha_{ji}. \tag{3.27}$$

It follows from eq. (3.27) and eq. (3.24) that

$$
\begin{aligned}
\pi_i q_{ij} \alpha_{ij} &= \pi_i q_{ij} \min\left\{1, \frac{\pi_j q_{ji}}{\pi_i q_{ij}}\right\} \\
&= \min\left\{\pi_i q_{ij}, \frac{(\pi_i q_{ij})(\pi_j q_{ji})}{\pi_i q_{ij}}\right\} \\
&= \min\left\{\pi_i q_{ij}, \pi_j q_{ji}\right\}.
\end{aligned} \tag{3.28}
$$

By denoting the ratio, $\rho_{ji} = \pi_j q_{ji} / \pi_i q_{ij}$, eq. (3.28) can be rewritten as

$$
\begin{aligned}
\pi_i q_{ij} \alpha_{ij} &= \min\{\pi_j q_{ji}, \pi_j q_{ji} / \rho_{ji}\} \\
&= \pi_j q_{ji} \min\{1, \frac{\pi_i q_{ij}}{\pi_j q_{ji}}\} = \pi_j q_{ji} \alpha_{ji},
\end{aligned} \tag{3.29}
$$

which satisfies the detailed balance condition since $1/\rho_{ji} = \rho_{ij}$ (Tierney, 1998). Therefore:

$$
\begin{aligned}
\pi_i p_{ij} &= \pi_i q_{ij} \alpha_{ij} + \delta_i(j) \left(\sum_{l \neq i} q_{il} (1 - \alpha_{il}) \right) , \\
&= \pi_j q_{ji} \alpha_{ji} + \delta_j(i) \left(\sum_{l \neq i} q_{li} (1 - \alpha_{li}) \right) = \pi_j p_{ji} \ .
\end{aligned} \tag{3.30}
$$

and thus, π is *the stationary distribution* of the Markov chain generated by the Metropolis-Hastings algorithm. When the chain reaches a stage where π is the stationary distribution for the chain, then this distribution is kept the same for all subsequent stages. Namely, if the probability distribution of the chain at any step *n* is π, then the probability distribution of the next step is $\boldsymbol{\pi} = \boldsymbol{\pi} P$ (Gamerman and Lopes, 2006).

The realization of the transition probabilities in eq. (3.25) is given in the following pseudo-code.

```
Initialize the chain, $\Xi_0$, and set the iteration number n=0.
 1. Generate a sample $\Xi_j$ from a proposal distribution, $q(\Xi_0, \Xi_j)$
 2. Compute $\alpha_{0j}$
 3. Generate a random variable $\mu$ from a uniform distribution $U(0,1)$
      if $\mu \le \alpha_{0j}$, then set $\Xi_{n+1} = \Xi_j$
         else set $\Xi_{n+1} = \Xi_n$
      Update n to n+1,
 4. Repeat step 1 through 3.
```

Figure 3.1: The Metropolis-Hastings algorithm

If the proposal distribution is chosen to be symmetric, $q_{ij} = q_{ji}$, then the acceptance probability in eq. (3.24) collapses to the *Metropolis* acceptance probability

$$\alpha_{ij} = \alpha(\Xi_i \to \Xi_j) = \min\left[1, \frac{\pi_j}{\pi_i}\right]. \qquad (3.31)$$

Simulations following the above scheme are applications of the *Metropolis algorithm*. The form of the acceptance probability given above is not unique since there may be many acceptance functions generating a Markov chain with desired properties. For example, Barker (1965) replaced the above ratio by $\pi_j / (\pi_i + \pi_j)$, which again satisfies the detailed balance condition. However, Peskun (1973) showed that the form given above is optimal among many alternatives in terms of statistical efficiency (Tierney, 1994).

3.3.1 The Choice of the Proposal Distribution

To implement the M-H algorithm, a proposal distribution needs to be specified. There are many options for q_{ij}. We here present the most frequently chosen distributions that appear in the literature. For more alternatives see Chib and Greenbeerg (1995) and Tierney (1994).

- Random Walk Metropolis algorithm: If $q_{ij}(\Xi_i \to \Xi_j) = q_1(\Xi_j - \Xi_i)$, where q_1 is a multivariate distribution, then the chain driven by this transition probability is called a *random walk* chain since the new candidate is in the form of $\Xi_j = \Xi_i + z$, where z is a random variable with its distribution independent of the chain. If q_1 is symmetric about the origin, then q_{ij} is symmetric and the simple form of the acceptance probability given in eq. (3.31) can be used. Common choices for q_1 include the uniform distribution, a multivariate normal or a t-distribution. A simulation that uses this generating scheme is called a *random walk Metropolis algorithm*.

The Metropolis algorithm can be used to generate samples from any target distribution either known explicitly or up to a normalization constant. For illustration purposes, here we present an example provided in the paper of Hastings (1970). The target distribution is the normal standard distribution $N(0,1)$ given by

$$\pi_j = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} (\Xi_j)^2 \right), \tag{3.32}$$

and the proposal distribution is the uniform distribution on $[-\delta, \delta]$, where $\delta > 0$. The new candidate is generated as $\Xi_j = \Xi_i + z$, where $\Xi_j \sim U(\Xi_i - \delta, \Xi_i + \delta)$. Due to the symmetric proposal distribution, the ratio in the acceptance probability will be

$$\rho_{ij} = \frac{\pi_j}{\pi_i} = \exp\left(\frac{1}{2} (\Xi_i - \Xi_j)^2 \right). \tag{3.33}$$

Using the scheme given in Figure 3.1, random samples are generated for the case $\delta = 1$. The Figure 3.2 illustrates the result.

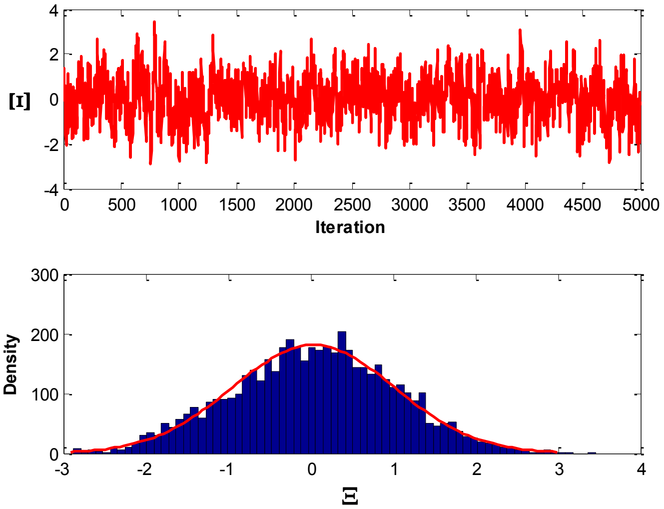


Figure 3.2: Random walk Metropolis algorithm, U(-1,1)

The top graph shows the sequence of the generated samples obtained during 5000 iterations without discarding any samples. The bottom graph shows that a density histogram of these samples fits the target distribution. With the choice of $\delta = 1$ and the initial value of the chain, $\Xi_0 = -0.1$, the chain converges to the target distribution starting from the initial value.

- Independence Sampler: If $q_{ij}(\Xi_i \to \Xi_j) = q_1(\Xi_j)$, then the new candidate is generated independently of the current state of the chain. In this case, the acceptance probability is expressed as

$$\alpha_{ij} = \min\left[1, \frac{\pi_j q_1(\Xi_i)}{\pi_i q_1(\Xi_j)}\right]. \tag{3.34}$$

The simulation that uses the above scheme is called the *independence sampler* or *independent Metropolis-Hastings algorithm*. In this algorithm, although the new candidate is generated independently of the current state, the resulting sample is not independent since the acceptance probability of the new candidate depends on the current state (Robert and Casella, 2004).

3.4 Metropolis Algorithm

The original paper by Metropolis et al. (1953) deals with the equilibrium properties of a large system of particles at a given temperature *T*. It presents the first MCMC algorithm executed on a digital computer. This algorithm has been extensively studied in statistical physics (Hammersly and Handscomb, 1964). It can be considered a special case of the random-walk Metropolis algorithm, which was later generalized by Hastings (1970). The validation and feasibility of this algorithm has been studied extensively by Wood (1968) and Fosdick (1963).

The equilibrium distribution considered by Metropolis et al. (1953) is also used in our application. It is given by the Boltzmann distribution (Binder, 1988),

$$\pi_j = \frac{1}{Z_T} \exp\left(-\frac{E(\Xi_j)}{k_B T}\right), \tag{3.35}$$

where k_B is the Boltzmann constant, *T* is the temperature, Ξ_j is the state of the system, $E(\Xi_j)$ is the energy of the state Ξ_j, and Z_T is the *normalization constant* (known as *partition function* in statistical physics).

For the Boltzmann distribution, the Metropolis acceptance probability, given in eq. (3.31), is expressed as

$$\alpha_{ij} = \alpha(\Xi_i \to \Xi_j) = \min\left[1, \frac{\pi_j}{\pi_i}\right] = \min\left[1, \exp(-\frac{\Delta E}{T})\right], \tag{3.36}$$

where $\Delta E = (E(\Xi_j) - E(\Xi_i))$ is the change in the energy. The ratio in the acceptance probability is denoted by

$$\rho_{ij} = \exp(-\frac{\Delta E}{T}). \tag{3.37}$$

The implementation of this algorithm is given in the following pseudo-code.

```
Start with state $\Xi_0$ at random with energy $E(\Xi_0)$
Do over random moves
    $\Xi_j = \Xi_{j-1} \pm \delta \cdot r$,
where $\delta$ is the maximum allowed displacement, r is a random number in the range [1,-1]
    Calculate $E(\Xi_j)$ for a new candidate
        $\Delta E = E(\Xi_j) - E(\Xi_{j-1})$
        $\rho_{j-1,j} = \exp(-\Delta E / T)$
      if $\Delta E \le 0$, then $\Xi_{j-1} = \Xi_j$.
      if $\Delta E > 0$, then accept $\Xi_j$ with probability, $\rho_{j-1,j} = \exp(-\Delta E / T)$
End Do
```

Figure 3.3: Pseudo-code for the Metropolis et al. (1953) algorithm

This algorithm is the basis for the simulated annealing. Kirkpatrick et al. (1983) developed the simulated annealing algorithm where a cost function replaces the energy. The Metropolis algorithm is applied to generate a set of parameter values that produce a given energy defined by a temperature. The temperature is reduced sequentially according to a cooling schedule until no more change in the cost function (energy) is obtained for the generated parameters.

3.5 Simulated Annealing

The Simulated Annealing (SA) algorithm can be implemented in two ways, using homogeneous and inhomogeneous algorithms. The homogeneous algorithm can be described by a sequence of homogeneous Markov chains, each generated at a fixed temperature, where the temperature is reduced between subsequent Markov chains. Since each Markov chain reaches an equilibrium condition at every temperature *T*, it is assumed to be of infinite length (van Laarhoven and Aarts, 1987). In practice, the equilibrium condition is approximately achieved by performing enough transitions at the respective temperature *T*. In our SA algorithm, the equilibrium is reached when the average cost function does not change after a number of iterations.

The inhomogeneous algorithm is described by a single inhomogeneous Markov chain. The temperature is decreased between subsequent transitions (van Laarhoven and Aarts, 1987). In this case, the Markov chain does not need to reach a stationary distribution at every temperature *T*. Only the homogeneous algorithm is considered in this study.

Choosing an initial temperature, the rate of decreasing temperature, the number of transitions generated at each temperature, and the termination criterion forms the *cooling* schedule. The initial temperature T_0 is chosen high enough so that almost all transitions are accepted, i.e., the probability of transitions occurs with probability close to 1, $(e^{-\Delta E/T_0} \simeq 1)$. In our algorithm, at each temperature the selection of new candidates for the Markov chain is controlled so that about fifty percent of the total transitions are accepted. The length of the Markov chain is the number of transition steps performed at each temperature. In our algorithm, it could be defined by length=$L \times m$, where m is the number of variables of the problem, and *L* is a fixed number of iterations performed at every temperature *T* for each variable. In this study, a geometric cooling schedule, first proposed by Kirkpatrick et al. (1983), will be considered and introduced in the next section. One typical termination criterion, also considered in this study, is to terminate the algorithm when the average cost does not change significantly for a few consecutive values of the temperature, *T*.

In the following, we present a homogeneous SA algorithm based on the Metropolis algorithm in a continuous domain as introduced by Corana et al. (1987).

3.5.1 Simulated Annealing (SA) for the Subsurface Anomaly Detection Problem

The goal of the optimization problem is to estimate geometric parameters (e.g., depth, horizontal location and shape) of subsurface structures from airborne gravitational gradients by minimizing a specified error function. For the inversion problem, a forward problem needs to be specified corresponding to the structure to be estimated. Two structures are considered; an infinite horizontal dip-slip fault and a right rectangular prism of constant density contrast with the assumption that the density contrasts in both cases are known. The cost function is defined as a sum of the squared differences between the observed and the model-predicted gravitational gradients,

$$\Phi(\Xi) = \sum_{i=1}^{n} [{}^{i}\Gamma^{obs} - {}^{i}\Gamma^{pred}(\Xi)]^2 , \tag{3.38}$$

where ${}^{i}\Gamma^{obs}$ are observed gravitational gradients at the i^{th} location, ${}^{i}\Gamma^{pred}(\Xi)$ are synthetic gravitational gradients computed using forward modeling at these observation points, $X = [X_1, X_2, ..., X_m]^T$ is the vector of size $m \times 1$ that includes the model parameters to be estimated, and n is the number of observation points along the chosen survey tracks. The goal of the inversion is to find a vector Ξ, the estimate of X, that produces the smallest objective function $\Phi(\Xi)$ among all possible solutions.

The SA algorithm proceeds as follows. At a given initial temperature, T_0, the algorithm starts with a configuration given or chosen randomly. That is, the initial realization of the random vector X is $\Xi^0 = [\Xi_1^0, \Xi_2^0, ..., \Xi_m^0]$, within predefined search domains for each parameter,

$$l_1 < \Xi_1^0 < u_1, ..., l_m < \Xi_m^0 < u_m , \tag{3.39}$$

where $(l_1, ..., l_m)$ and $(u_1, ..., u_m)$ are lower and upper boundaries, respectively. Here, any combination of $\Xi_1^0, \Xi_2^0, ..., \Xi_m^0$ will be considered a configuration. With the given initial configuration, the gravitational gradient response is computed by using the forward model. Then, the objective function for this configuration, $\Phi(\Xi^0)$, is computed. Next, a new configuration is obtained by displacing the value, Ξ_h, of one element of the model parameter vector, X, as follows;

$$\Xi_h^1 = \Xi_h^0 + r \cdot \delta_h , \tag{3.40}$$

subject to $l_h < \Xi_h^1 < u_h$. Here h indicates the element in the vector Ξ that is perturbed (h=1,……,m); δ_h is the h^{th} component of the step length vector $\delta = [\delta_1, \delta_2, ..., \delta_m]$; r is a random number drawn from a uniform distribution between [-1,1].

Then, the corresponding objective function is computed at, $\Xi^1 = [\Xi_1^0, ..., \Xi_h^1, ..., \Xi_m^0]$. If $\Phi(\Xi^1)$ is smaller than the current cost function, $\Phi(\Xi^0)$, the new configuration is accepted unconditionally. If $\Phi(\Xi^1)$ is smaller than the optimum function, $\Phi(\Xi_{opt}^{(k)})$, that is recorded so far, it is recorded as the new optimum function, ($\Phi_{opt}^{(k)}$), and the related point, Ξ^1, is also recorded as the current optimal point (Ξ_{opt}). If $\Phi(\Xi^1)$ is greater than $\Phi(\Xi^0)$, the transition to the new configuration, Ξ^1, is made by the Metropolis acceptance ratio,

$$\rho_{01} = \exp\left(\frac{(\Phi(\Xi^0) - \Phi(\Xi^1))}{T_k} \right), \tag{3.41}$$

where T_k is the "current temperature" at the k^{th} iteration (k=0,1,2, …..). The value ρ is computed and compared with a value μ, randomly drawn from the uniform distribution between [0,1]. If the value μ is smaller than or equal to ρ ($\mu \le \rho$), then the transition is accepted, and the new state Ξ^1 replaces the old state Ξ^0. If the value μ is larger than ρ ($\mu > \rho$), the transition is rejected. Then the next search starts from the previous configuration. Since each time only one component in the vector Ξ is perturbed, the above acceptance- perturbation procedure is repeated until all m components in the vector Ξ are displaced. The above steps are repeated L_δ, times for each variable, where L_δ is a predefined integer. The length of the Markov chain is $N_T \cdot L_\delta \cdot m$ where, after every $L_\delta \cdot m$ iterations, the step length is dynamically adjusted within the chain at a fixed temperature as follows: Corana et al. (1987) suggested a variation in the step length so that roughly 50% of the total transitions is accepted since too many or too few accepted transitions lead to a waste of computational effort. The computation of the h^{th} element, δ_h', of the step vector is based on its number of accepted points, M_h, from its last L_δ iterations;

$$\begin{aligned}
\delta_h' &= \delta_h \cdot \left(1 + c_h \frac{M_h / L_\delta - 0.6}{0.4} \right) && \text{if } M_h > 0.6 L_\delta \\
\delta_h' &= \frac{\delta_h}{\left(1 + c_h \dfrac{0.4 - M_h / L_\delta}{0.4} \right)} && \text{if } M_h < 0.4 L_\delta \\
\delta_h' &= \delta_h && \text{if } 0.4 L_\delta < M_h < 0.6 L_\delta .
\end{aligned} \tag{3.42}$$

Here, $(c_h)_{m\times 1}$ is the vector of size $m \times 1$ that determines the magnitude of the step length being adjusted. $(M_h)_{m\times 1}$ is the vector of size $m \times 1$ that records the number of the accepted points for each component in the vector Ξ every $L_\delta \cdot m$ iterations. After the step length is adjusted, each recorded entry in the vector M_h is set to zero. These adjustments to each component of the step length vector is repeated N_T times at the fixed temperature. After $N_T \cdot L_\delta \cdot m$ realizations of the Markov chain, equilibrium presumably is reached and then the temperature is reduced.

Before the temperature reduction, the termination criteria are checked. The SA algorithm is terminated if the difference between the final cost function Φ_k at the current temperature and the current best optimal cost function $\Phi_{opt}^{(k)}$ is smaller than a given threshold (ε), and the differences between the cost function values at the preceding E_ε temperature reductions and the final cost function Φ_k at the current temperature is smaller than the error tolerance (ε)

$$\left|\Phi_k - \Phi_{opt}^{(k)}\right| \le \varepsilon, \qquad k=0,1,\ldots.., \tag{3.43a}$$

$$\left|\Phi_k - \Phi_{k-v}^{*}\right| \le \varepsilon, \qquad v=1,\ldots., E_\varepsilon, \tag{3.43b}$$

The suggested value for E_ε is 4. At the beginning of the algorithm, Φ_{k-v}^{*} is set to $\Phi_k(\Xi^0)$. The second termination criterion means that, in addition to satisfying eq. (3.43a), if, after 4 successive temperature reductions, the difference between 4 final cost function values recorded at the end of each temperature and the optimal current function value is smaller than the error tolerance (ε), the algorithm is terminated. If the termination criteria are not met, the temperature is reduced. The temperature is reduced by a geometric cooling schedule,

$$T_{k+1} = \tau_T \cdot T_k, \tag{3.44}$$

where τ_T is the constant temperature reduction rate between $0 < \tau_T < 1$. The final optimum cost function and final optimum configuration are used to start the next search for the optimum point and cost function at the new temperature. After another $N_T \cdot L_\delta \cdot m$ cycles at the new temperature, both termination criteria are checked. If they are not met, the same procedure is repeated until the termination criteria are satisfied.

Corana et al. (1987) suggested the following values of the parameters that control the SA:

$$\begin{aligned} &L_\delta = 20, \\ &N_T = \max(100, 5\cdot m), \\ &c_h = 2, \quad h = 1,\ldots\ldots, m, \\ &E_\varepsilon = 4, \\ &\tau_T = 0.85. \end{aligned}$$

Some of the above parameters are used in our numerical examples. The choice of initial temperature T_0 and N_T depends on the problem at hand. In our case, they depend on the size of the problem and the quality of the observations that are used. They could be defined by trial and error.

A pseudo-code of this algorithm is given in Figure 3.4.

```
Define the initial parameters, $L_\delta, N_T, c_h, E_\varepsilon, \tau_T, T_0, \delta_h, \varepsilon, \Xi^0$
Set $l$=0
Compute $\Phi(\Xi^0)$
Set $\Xi_{opt} = \Xi^0$ and $\Phi_{opt} = \Phi(\Xi^0)$
SA loop
    Do until termination criteria are satisfied
    Markov Chain
        Do i=1, $N_T$
        .  Set $M_h = 0$
        .  Do j=1, $L_\delta$
        .      Do h=1, m
                   $\Xi_h^{l+1} = \Xi_h^l + r \cdot \delta_h$
        .          Compute $\Phi(\Xi^{l+1})$
        .              If $\Phi(\Xi^{l+1}) \le \Phi(\Xi^l)$ then
        .                  Set $\Xi^l = \Xi^{l+1}$ and $\Phi(\Xi^l) = \Phi(\Xi^{l+1})$,
                           $M_h$ to $M_h + 1$, $l$ to $l + 1$
        .              If $\Phi(\Xi^{l+1}) < \Phi_{opt}$ then
        .                  Set $\Phi_{opt} = \Phi(\Xi^{l+1})$ , $\Xi_{opt} = \Xi^{l+1}$
        .              If $\Phi(\Xi^{l+1}) > \Phi(\Xi^l)$ then
        .                  Compute the Metropolis ratio
        .                  If it is accepted, then
                               Set $\Xi^l = \Xi^{l+1}$, $\Phi(\Xi^l) = \Phi(\Xi^{l+1})$,
        .                      $M_h$ to $M_h + 1$, $l$ to $l + 1$
        .  End Do
        .  End Do
        .    Adjust the step length, $\delta_h$, (h=1,..,m)
        .    Set $\delta_h = \delta'_h$
        End Do
            Check the termination criteria
              If they are satisfied, then terminate the algorithm
                  Else
                    Reduce the temperature,
                    $\Xi = \Xi_{opt}$ and $\Phi(\Xi) = \Phi_{opt}$
    End Do
```

Figure 3.4: Pseudo-code for the simulated annealing algorithm

Chapter 4: Numerical Experiment

4.1 Case I: A right rectangular prism

It is assumed that there is an anomaly buried below the Earth's surface, and the gravitational gradient effects due to this anomaly are observed or simulated at some points along survey profiles on or above the surface. The task is to locate the anomaly from one or more observation profiles of the gravitational gradients. The subsurface anomaly is modeled as a right rectangular prism of constant negative density contrast which can be defined by seven parameters; width (b), length (a), height (c), depth (d) and origin coordinates of the prism (x_0, y_0), and orientation angle (α) defined by rotating the prism about the z-axis (Jekeli and Abt, 2010). Figure 4.1.a-b describes the right rectangular prism defined in a local east-north-up (ENU) coordinate system.

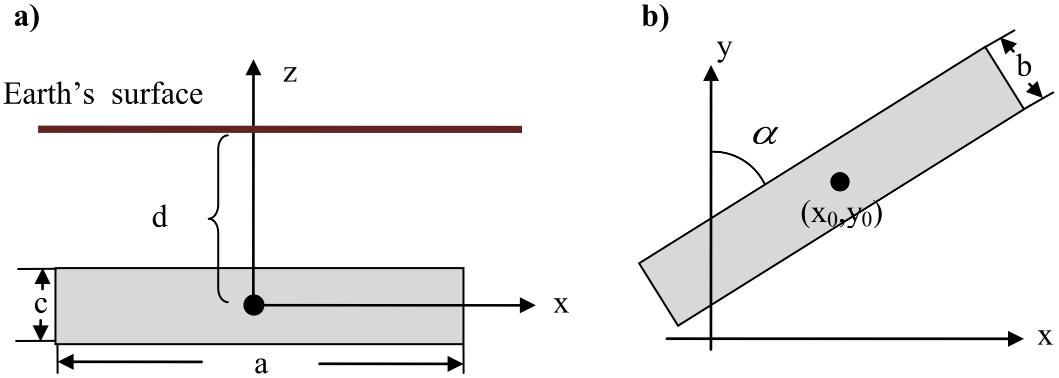


Figure 4.1: Definition of a right rectangular prism's parameters in an ENU coordinate system

In case that $\alpha = 0^o$, all the coordinate axes are aligned with the prism sides. Gravitational gradients due to such a right rectangular prism can be calculated by the following formulas (Jekeli and Abt, 2010)

$$\Gamma_{11} = \arctan \frac{(y-y')(z-z')}{(x-x')r} \Bigg|_{x'=-a/2}^{a/2} \Bigg|_{y'=-b/2}^{b/2} \Bigg|_{z'=-c/2}^{c/2}$$

$$\Gamma_{22} = \arctan \frac{(z-z')(x-x')}{(y-y')r} \Bigg|_{x'=-a/2}^{a/2} \Bigg|_{y'=-b/2}^{b/2} \Bigg|_{z'=-c/2}^{c/2}$$

$$\Gamma_{33} = \arctan \frac{(x-x')(y-y')}{(z-z')r} \Bigg|_{x'=-a/2}^{a/2} \Bigg|_{y'=-b/2}^{b/2} \Bigg|_{z'=-c/2}^{c/2} \tag{4.1}$$

$$\Gamma_{12} = -\ln(z-z'+r) \Big|_{x'=-a/2}^{a/2} \Big|_{y'=-b/2}^{b/2} \Big|_{z'=-c/2}^{c/2}$$

$$\Gamma_{23} = -\ln(x-x'+r) \Big|_{x'=-a/2}^{a/2} \Big|_{y'=-b/2}^{b/2} \Big|_{z'=-c/2}^{c/2}$$

$$\Gamma_{13} = -\ln(y-y'+r) \Big|_{x'=-a/2}^{a/2} \Big|_{y'=-b/2}^{b/2} \Big|_{z'=-c/2}^{c/2} .$$

where $r = \sqrt{(x-x')^2 + (y-y')^2 + (z-z')^2}$, (x, y, z) are the coordinates of the computation point of the gravitational gradient effect and (x', y', z') are the variables of integration over the source body.

The study in this section focuses on two different cases, where the subsurface anomaly is crossing the observation profiles orthogonally or at a certain azimuthal angle. For both cases, possible observation noise has been investigated with the further assumption that the sought anomaly is buried in a geologic background. The geologic background can be described as a local gravitational gradient field, provided by a Bell Geospace Inc. survey (Bell Geospace, 2008). The gravity gradiometer survey was performed over Vinton Dome, Louisiana, in July 2008 to test the new FTG full tensor gravity gradiometer installed in an aircraft, BT-67 C-FTGI, shown on the left and right of the Figure 4.2, respectively.

Figure 4.2: FTG gradiometer and aircraft (Bell Geospace, 2008)

The location of the Vinton Dome region is situated in south-western Louisiana near the Texas border. The survey region lies between latitudes 30.07^o and 30.23^o in the north-south direction and longitudes -93.66^o and -93.53^o in the east-west direction. The airborne gravitational gradients were collected at the ground speeds of 215 km/hr at the average altitude of 84.9 m. The data interval between points along the survey tracks is not equally spaced, but is approximately ranging from 47 to 55 m. Points along the tracks have coordinates in the WGS 84 coordinate system. The survey profiles are 16.7 km long in the north-south direction. Gravitational gradients are de-biased, de-noised, and terrain corrected using a topographic density of 1.8 gm/cm^3 (Figure 4.3).

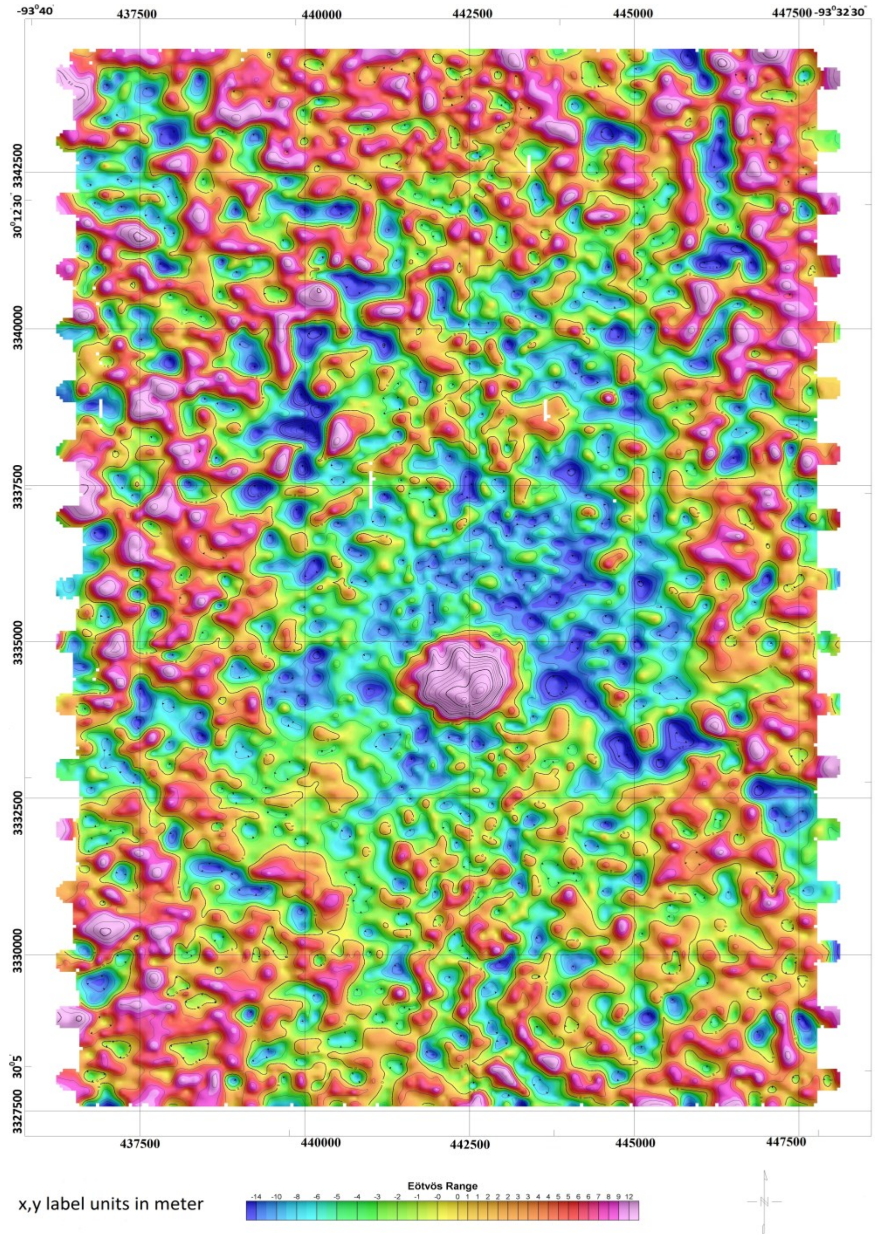


Figure 4.3: Terrain corrected Γ_{33} map with density of 1.8 gm/cm^3 (Bell Geospace, 2008)

There are 53 measured survey profiles along the north-south direction. In this study, three observation profiles (L31, L371 and L841) which are indicated with red circles in Figure 4.4 are considered.

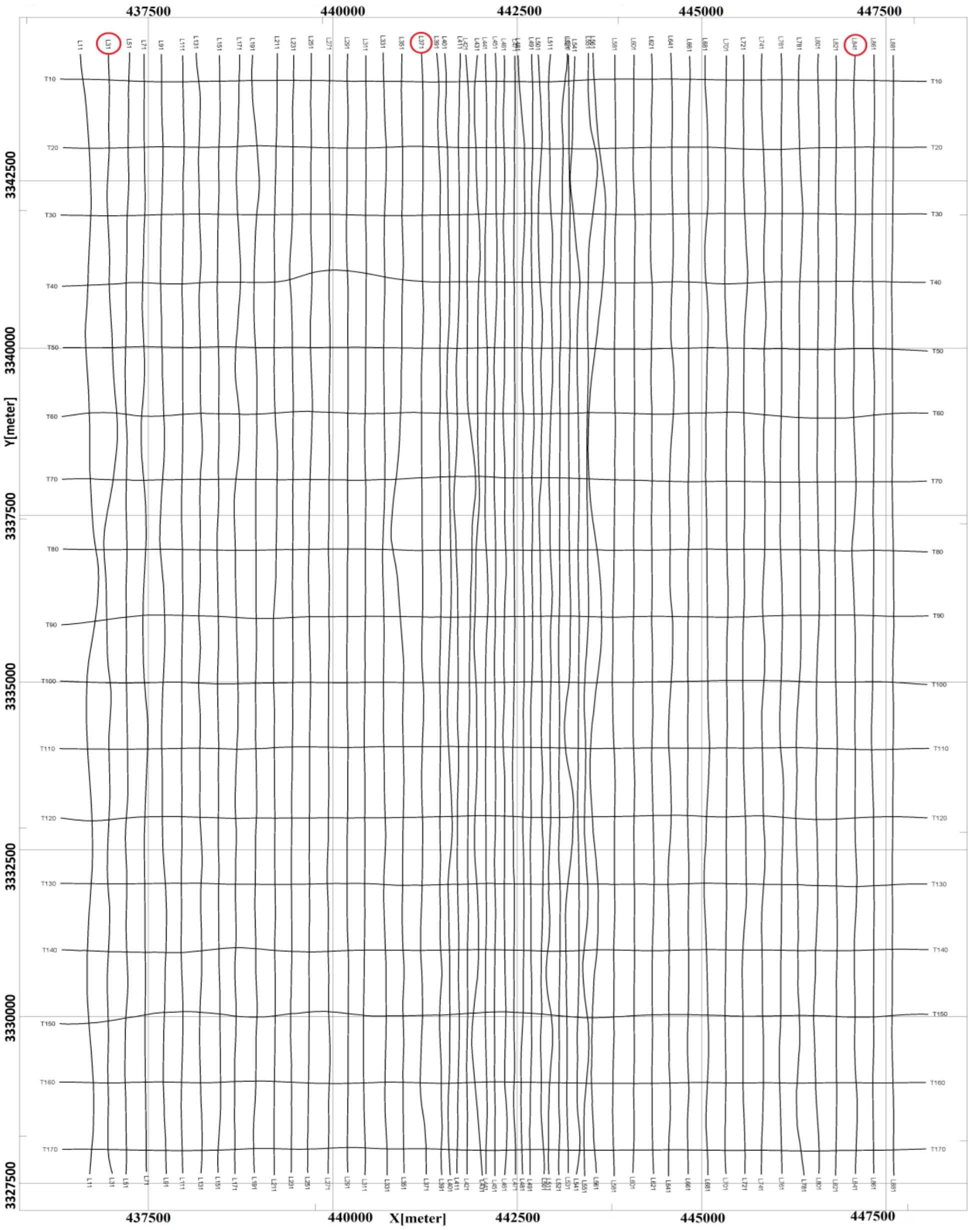


Figure 4.4: Flight lines and chosen gradiometer survey tracks (L31, L371 and L841) (Bell Geospace, 2008)

The first profile is one of the most westerly profiles of the field, the second is about 4 km away from the first one, and the third is one of the most easterly profiles of the field and about 10 km away from the first profile.

For this application, the north-south direction is set as the y-axis and the east-west direction is set as the x-axis in the system of coordinates. A prism of constant density is placed such that its length is parallel to the east-west direction in the field, as shown in Figure 4.5. It is also parallel to the horizontal plane. The origin of the coordinates is chosen to be at the center of the prism. The prism has a length of a=12 km, a width of b=100 m, and a height of c=100 m.

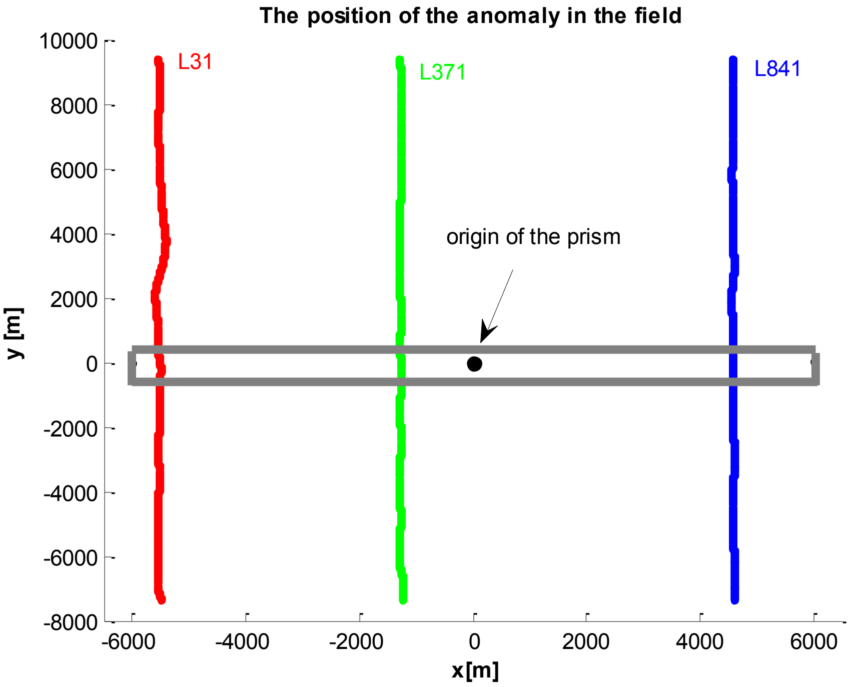


Figure 4.5: The position of the sought anomaly

The measurement points on the survey tracks are at flight altitude. The flight altitudes are GPS altitudes referenced to WGS84. They are also measured and given in the FTG data file. Therefore, z coordinates of the measurement points are computed as flight altitude plus depth (d) of the subsurface prism as illustrated in Figure 4.6.

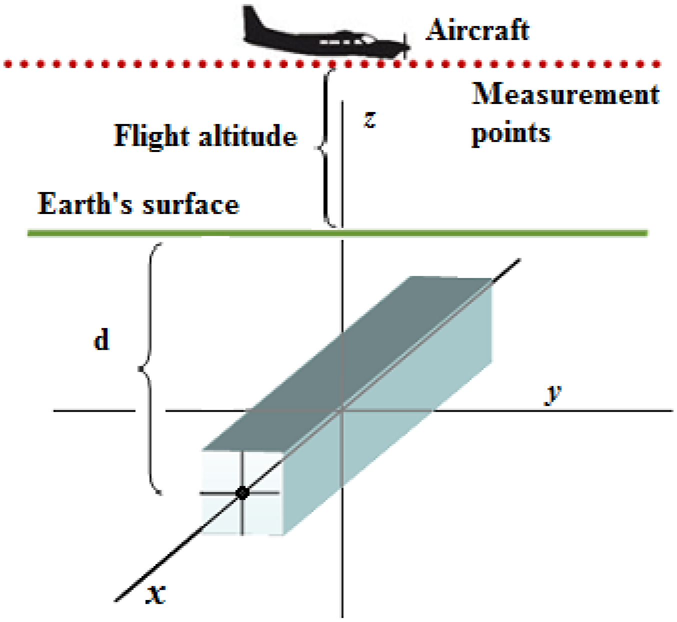


Figure 4.6: Setting up the coordinate system for the forward problem

Since we are searching for the anomaly along the profiles in the y direction, the x_0 origin coordinate of the prism is assumed to be known. Therefore, the origin coordinate y_0 is considered as the unknown parameter here. The cost function is defined as

$$\Phi_{jk}(\Xi) = \sum_{i=1}^{n} [{}^{i}\Gamma_{jk}^{obs} - {}^{i}\Gamma_{jk}^{pred}(\Xi)]^2 , \tag{4.3}$$

where j=1,2,3, k=1,2,3, i=1,….n, and n is the number of observation points along the chosen tracks. Γ_{jk}^{obs} denotes observed or simulated gravitational gradients in a vector of size $n \times 1$. Γ_{jk}^{obs} is always considered to be a vector in our computations; therefore, in case of using the second profile, it is just added to the end of the first profile. Ξ indicates the unknown parameters, which for this test is a vector of size 3×1, for the parameters $\Xi_{m\times 1} = [d, y_0, \alpha]^T$. In case that the geologic background is included, ${}^{i}\Gamma_{jk}^{obs}$ will be the combination of the signal generated by the subsurface anomaly itself plus the chosen observation profile from the gradiometer survey. Otherwise, ${}^{i}\Gamma_{jk}^{obs}$ includes only the known signal generated by the subsurface anomaly. In contrast, ${}^{i}\Gamma_{jk}^{pred}(\Xi)$ is the computed gravitational gradient using the forward model with particular parameter values according to the Monte Carlo/ Simulated Annealing algorithm. In other words, the gravitational gradient from the geologic background is like a correlated noise imposed on the signal of the anomaly.

We firstly implement the Metropolis algorithm at a fixed temperature T (i.e., no SA) to estimate the depth and location parameters (d, y_0) of the prism for the case where the anomaly crosses the survey track orthogonally. Search domains are defined by 1m<d<300 m and -7300 m<y_0<9500 m. In our application, the Metropolis algorithm can be considered as the inner loop (Metropolis cycle) of the simulated annealing (SA) in Figure 3.4, with the modification that instead of adjusting the step-length periodically for N_T times, we perform L_δ iterations for each variable at a fixed temperature, T where the step-length, δ is fixed. In addition, termination criteria are introduced and checked after every $L_\delta \cdot m$ iterations (m=2). The termination criteria are met if the differences between the previous 4 cost functions and the current one, and the difference between the current cost function and the optimum cost function are all smaller than or equal to $\varepsilon = 1\times10^{-6}$ [E^2]. If the termination criteria are not met, another $L_\delta \cdot m$ iterations are performed until the termination criteria are achieved. For this application, L_δ=20, δ is set as 0.1, and the temperature is $T = 3\times10^{-5}[E^2]$. Initial values are chosen as $y_0 = 10$ m and d=2 m. The observations are simulated vertical gravitational gradients, Γ_{33}, (j=k=3), along the survey track L31 without geologic background. The true parameters of the location and depth are y_0=0.0 m and d=150 m. Figure 4.7 illustrates a plot of the cost function for this application.

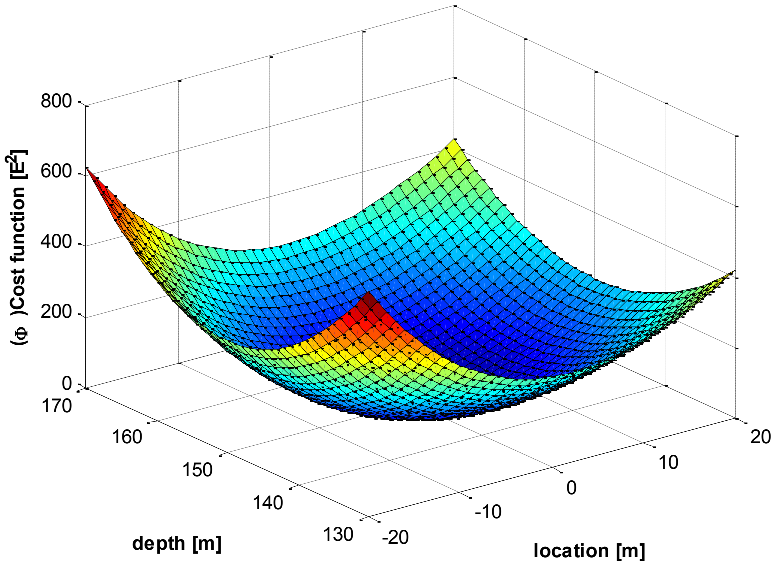


Figure 4.7: The cost function for the depth and location parameter

Figures 4.8.a-b) show profiles of the cost function taken at location y_0=0.0 m and at depth=150 m.

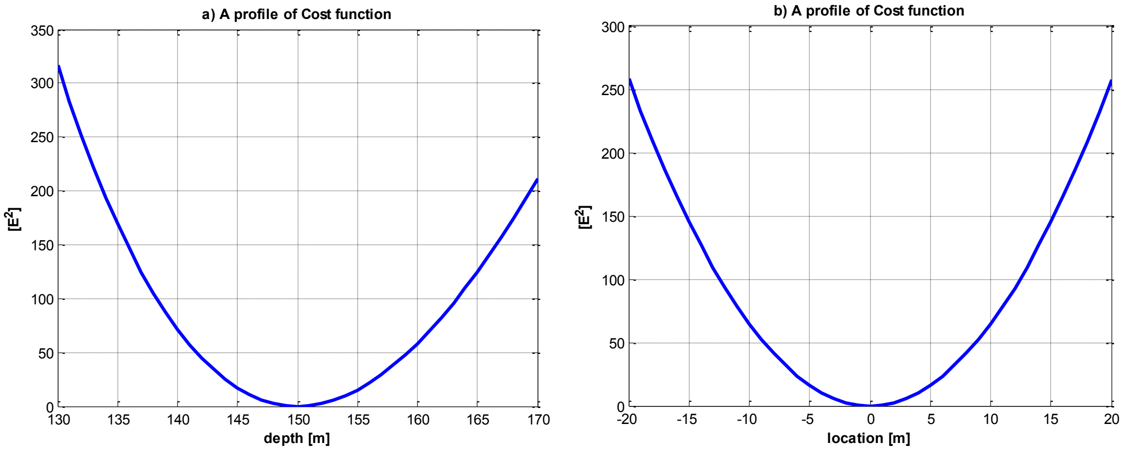


Figure 4.8.a-b): Profile of the cost function at y_0 =0.0 m (right) and at depth=150 m (left)

Figures 4.8.a-b indicate that the cost function is symmetric with respect to the location but not symmetric with respect to the depth. The estimated parameters are given in Table 4.1.

Table 4.1: Estimated parameters of the prism using the Metropolis algorithm at a single temperature, T

Location-(y_0) [m]	**Depth (d) [m]**	**(Φ) Cost Function [Eotvos^2]**
-0.00004	150.00001	0.0000000011

The results presented in Table 4.1 are obtained if the starting value for the location parameter is close to the actual value. For example, if the location parameter is chosen between [-600, 600] and any value is chosen for the depth parameter within the search domain, the algorithm finds the minimum of the cost function. When the starting value for the location parameter is chosen far away from the actual one, the Metropolis algorithm does not converge to the global minimum of the cost function. This happens because the cost function has a local minimum with respect to the location parameter as illustrated in Figure 4.9. Figure 4.9 shows a profile of the cost function taken at fixed depth=150 m for an extended domain.

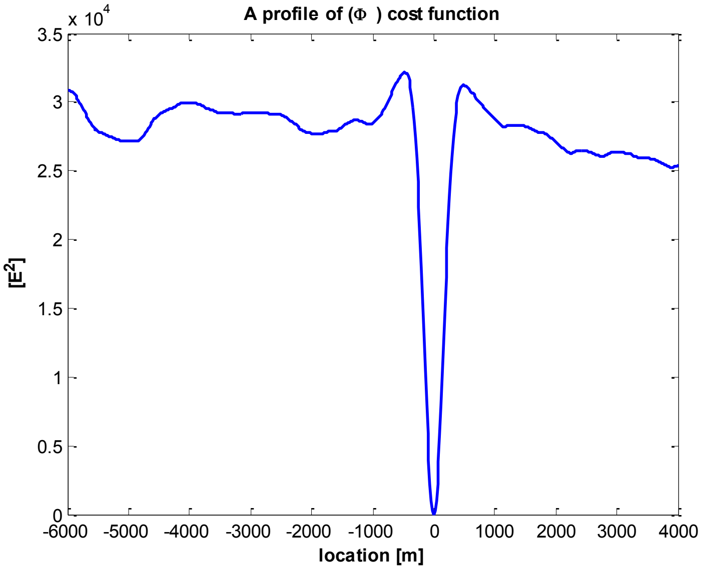


Figure 4.9: A profile of the cost function at fixed depth=150 m

Therefore, we use the simulated annealing algorithm for this application.

4.1.1 Subsurface Anomaly Detection from Vertical Gravity Gradients using Simulated Annealing (SA)

For the implementation of the SA algorithm, the following parameters are chosen: termination criterion, $\varepsilon = 1\times10^{-21}$[E^2], the parameter for the step-length adjustment, $c_{h=1,\ldots,m} = 2.0$; and the number of successive temperature reductions to check the termination criterion, E_ε=4. The other parameters to define SA will be provided in each case when needed. In the following, the size of the anomaly will be described and the plots of signals will be shown with and without geologic background for each case, respectively; afterwards, the corresponding results will be presented.

a) The anomaly is orthogonal to the survey tracks

(i) In this case, the generated signal due to the subsurface prism is known and an attempt will be made to locate it from simulated observations of Γ_{33} without including geologic background. To simulate one observation profile, the following parameters for the subsurface prism are used: width of *b*=100 m, length of a=12 km, height of *c*=100 m, depth of *d*=150 m, and density contrast of ρ= - 2670 kg/m^3. The data spacing varies around 54 m, with 310 points along the profile.

For the SA implementation, the search domains for the depth (*d*) and location (y_0) are defined as 1m<*d*<300 m, -7300 m<y_0<9500 m, respectively. The parameters that control the SA are chosen as follows. The initial temperature is defined as T_0=200 [E^2]; N_T=5, τ_T =0.85 and L_δ=20. At each temperature, $N_T \cdot L_\delta \cdot m$=5*20*2 iterations have been performed. Figure 4.10 shows the simulated signal due to the subsurface anomaly computed at 310 points at the average altitude of 84 m along the survey track L31 and Table 4.2 presents estimated parameters for that anomaly.

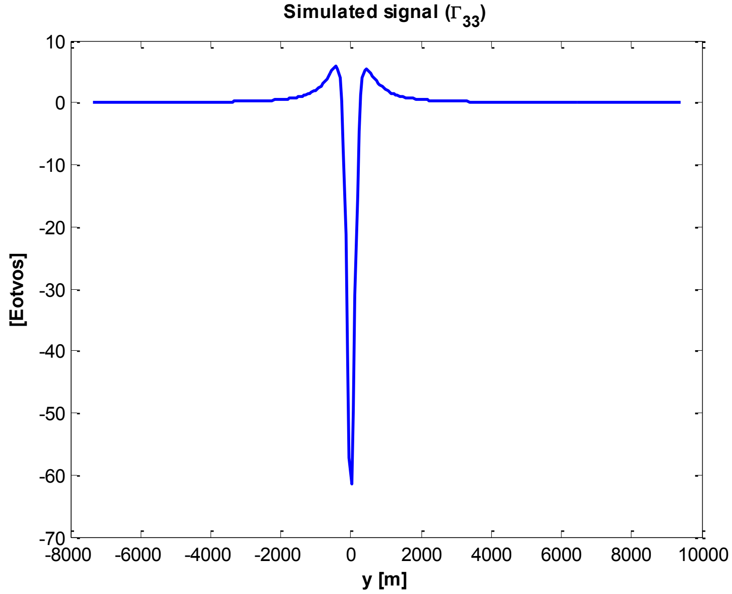


Figure 4.10: Sought signal profile

Table 4.2: Estimating parameters of the prism using no geologic background

Location-y$_0$ [m]	**Depth (d)[m]**	**(Φ) Cost Function [Eotvos^2]**
-0.7×10^{-12}	150	0.4×10^{-23}

As can be seen in Table 4.2, the estimated parameters are perfectly estimated. The algorithm is very successful in locating the prism if the generated signal does not include any geologic background.

(ii) In this case, to test the capability of the SA algorithm in estimating the prism's parameters, various possible observation noises from the geologic background have been included. Specifically, six different cases were investigated where the strength of the simulated signal gets weaker in the geologic background. This can be achieved by varying the depth of the anomaly so that it becomes less visible in the geologic background. Keeping the same size values as before, the six different depths range from 60 to 250 meters. Each simulated signal, combined with the same geologic background, is then treated as the observed gradient. Figure 4.11 shows one of the chosen observation profiles from the gradiometer survey for this application.

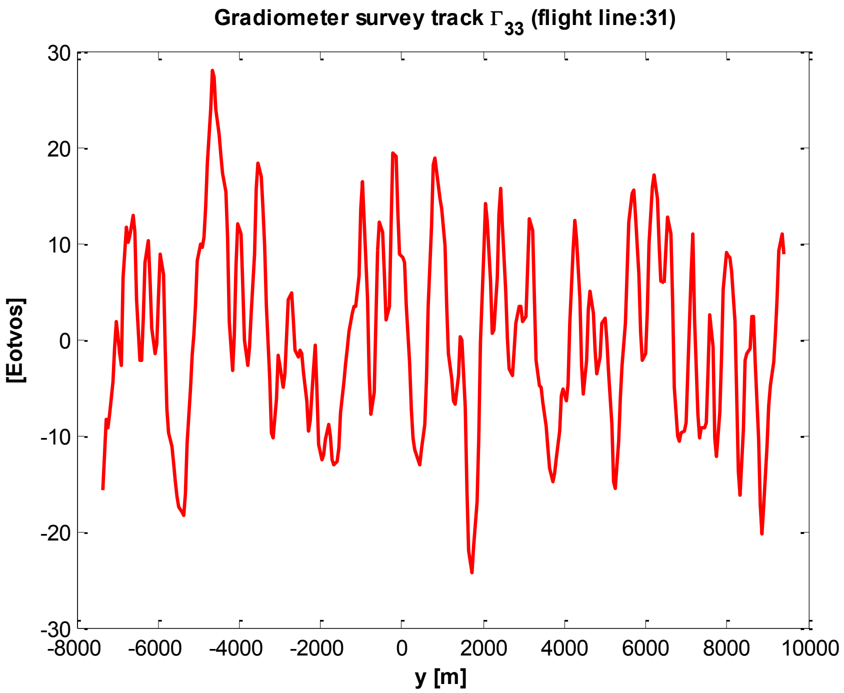


Figure 4.11: Γ_{33} gradiometer survey track (L31) data without embedded anomaly signal

The search domain for the location is defined as -7300 m<y_0<9500 m. For the first four cases, the search domain for the depth is 1m<d<300 m, and for the last two cases, it is defined as 1m<d<500 m. The parameters that control the SA algorithm are chosen as follows: initial temperature T_0=5000 [E^2], N_T=20, τ_T =0.85, L_δ=20, initial step length δ=100. At each temperature, $N_T \cdot L_\delta \cdot m$=20*20*2 iterations are performed.

The Figures 4.12.a)-f) illustrate the simulated anomaly plus gradiometer survey data for the six different cases, respectively.

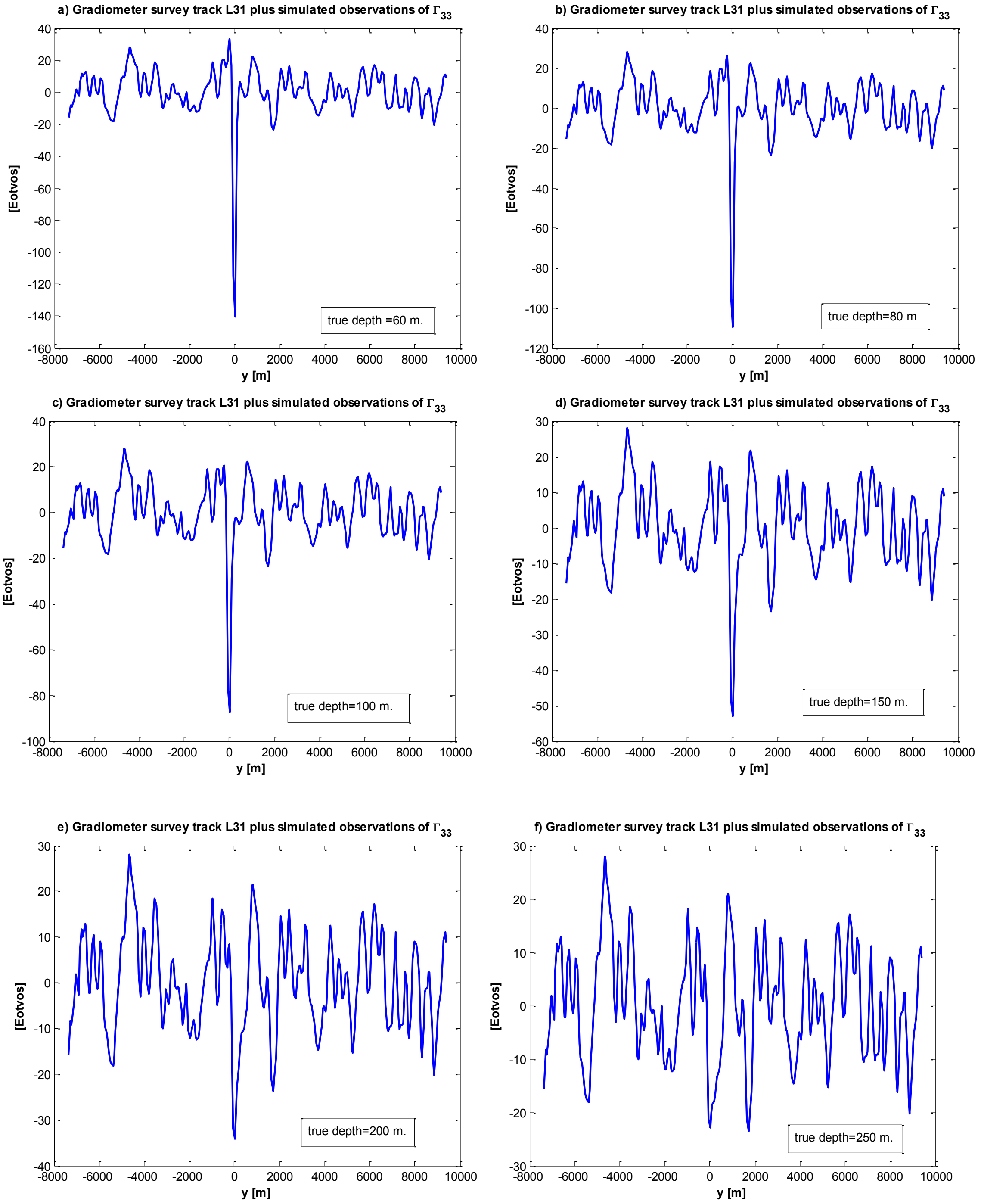


Figure 4.12.a-f): Simulated anomalies generated for six different depths of 60, 80, 100, 150, 200 and 250 m, respectively, plus gradiometer survey data on track L31

Table 4.3: Summary of estimated depth and location using one gradiometer survey track L31 plus simulated signal of Γ_{33}, true y_0 =0.0 m

True Depth d[m]	Estimated Location-y_0 [m]	Estimated Depth-d [m]
60	3.5	62
80	6.6	85
100	11.2	108
150	33.3	173
200	79.0	241
250	146.4	292

The criterion of the SA algorithm to estimate the prism parameters successfully is based on the size of the anomaly. Therefore, in this application, a criterion for a "good estimate" for the location parameter is fulfilled if the estimated location value falls in the range of true location ± 50 m. In addition, a criterion for "good estimate" for the depth is fulfilled if the estimated depth value falls in the range of the true depth ± 50 m. Thus, as can be seen from Table 4.3, the location estimate is reasonable except for the cases where the depth is 200 m or 250 m. The algorithm basically breaks down at the depths of 200 m and 250 m for the location estimate while the depth estimations are still good enough, considering the size of the anomaly. For example, the estimated locations of the anomaly for the last two cases are about 29 m, respectively 96 m away from the side of the prism facing north in the y-direction. Therefore, to locate the anomaly is not really possible although the depth estimation is adequate. This is so because the signal-to-noise ratio for these two signals is small.

To characterize the signal-to-noise ratio for the six different cases, the following formula is used,

$$\mathrm{SNR} = s^T(\phi)^{-1}s, \tag{4.4}$$

where s is the sought signal vector generated by the subsurface prism, and ϕ is the noise covariance matrix, which depends only on the horizontal coordinate differences between points along the survey track (Abt, 2011). The covariance matrix is based on a reciprocal distance model, given by (Jekeli, 2003) as

$$\phi_T(\Delta x, \Delta y; z, z') = \sum_j \frac{\sigma_j^2}{\sqrt{\left(1+\alpha_j(z+z')\right)^2 + \alpha_j^2 r_j^2}}, \tag{4.5}$$

where $\Delta y = y - y'$, $\Delta x = x - x'$ are the horizontal coordinate differences, σ_j^2 and α_j are the model parameters. Horizontal coordinate differences, Δx, Δy are computed by taking difference between the coordinates of the first and second points. Thus, the distance is defined by

$$r = \sqrt{\Delta x^2 + \Delta y^2} \; . \tag{4.6}$$

All points lie at the height of $z = z' = 84\,\text{m}$. The above covariance model is for the disturbing potential, *T*. Covariances for the gravity gradients can be derived by applying the law of error propagation to eq. (4.5). With the following notation,

$$\begin{aligned} \beta_j &= 1 + \alpha_j (z + z'), \\ M_j &= \beta_j^2 + \alpha_j^2 r^2 \; , \end{aligned} \tag{4.7}$$

the covariance function for Γ_{33} is computed in (Jekeli, 2003) as follows:

$$\phi_{\Gamma_{33}} = \sum_j \frac{3\sigma_j^2 \alpha_j^2}{M_j^{9/2}} \left(8\beta_j^2 - 24\beta_j^2 \alpha_j^2 r_j^2 + 3\alpha_j^4 r_j^4 \right), \tag{4.8}$$

where α_j and σ_j^2 are values adjusted to the geologic background for the Vinton Dome region (Abt, 2011); they are provided in Table 4.4.

Table 4.4: Covariance model parameters

$\sigma_1^2 = 1\times10^5$	$\alpha_1 = 3\times10^{-7}$	$\sigma_9^2 = 3\times10^{-4}$	$\alpha_9 = 4.8\times10^{-4}$
$\sigma_2^2 = 3500$	$\alpha_2 = 7.7\times10^{-7}$	$\sigma_{10}^2 = 9\times10^{-6}$	$\alpha_{10} = 1.3\times10^{-3}$
$\sigma_3^2 = 778$	$\alpha_3 = 3\times10^{-6}$	$\sigma_{11}^2 = 4\times10^{-7}$	$\alpha_{11} = 3\times10^{-3}$
$\sigma_4^2 = 300$	$\alpha_4 = 8.5\times10^{-6}$	$\sigma_{12}^2 = 4\times10^{-8}$	$\alpha_{12} = 5\times10^{-3}$
$\sigma_5^2 = 20$	$\alpha_5 = 2\times10^{-5}$	$\sigma_{13}^2 = 5\times10^{-9}$	$\alpha_{13} = 1.1\times10^{-2}$
$\sigma_6^2 = 0.2$	$\alpha_6 = 6\times10^{-5}$	$\sigma_{14}^2 = 6\times10^{-11}$	$\alpha_{14} = 3\times10^{-2}$
$\sigma_7^2 = 0.02$	$\alpha_7 = 1\times10^{-4}$	$\sigma_{15}^2 = 5\times10^{-12}$	$\alpha_{15} = 5\times10^{-2}$
$\sigma_8^2 = 0.08$	$\alpha_8 = 2\times10^{-4}$	$\sigma_{16}^2 = 5\times10^{-13}$	$\alpha_{16} = 1.2\times10^{-1}$

To compute covariances along the gradiometer survey track L31, $\Delta x = 0$ is set and, since the data interval is not equally spaced between points, the average of coordinate differences $\Delta y = 54.198\,\mathrm{m}$ is used. Table 4.5 summarizes the computed signal-to-noise ratios for the six different cases.

Table 4.5: Signal-to-noise ratio for different depths of the anomaly

True Depths d [m]	SNR
60	602.2
80	238.4
100	108.8
150	24.2
200	8.2
250	3.6

From Table 4.5, the signal-to-noise ratio is relatively small for the anomaly depths of 200 m and 250 m. It can be concluded from these tests that the SA algorithm estimates both the location and the depth from one observation profile of vertical gravitational gradients if the signal-to-noise ratio is 24 or greater for this case according to the criterion of "good estimate" that was established before.

Weights may also be considered in the cost function,

$$\Phi_{jk}(\Xi) = \left({}^{i}\Gamma_{jk}^{obs} - {}^{i}\Gamma_{jk}^{pred}(\Xi) \right)^{T} P \left({}^{i}\Gamma_{jk}^{obs} - {}^{i}\Gamma_{jk}^{pred}(\Xi) \right), \tag{4.9}$$

where P is an $n \times n$ symmetric positive-definite weight matrix, j=k=3 and i=1,…,n (n=310). It is obtained from inverting the covariance matrix Σ, $P = \Sigma^{-1}$. The covariance matrix is defined by

$$\Sigma = \phi_{\Gamma_{33}} + \phi_{\mathrm{inst}}, \tag{4.10}$$

where $\phi_{\Gamma_{33}}$ is the computed covariance matrix for the geologic background gradients, Γ_{33}, and ϕ_{inst} is the covariance matrix for the instrument noise. ϕ_{inst} is a diagonal matrix where the diagonal elements show the variances σ_{inst}^2 of the instrument noise for the gradiometer, and the off-diagonal elements are zero. The standard deviation, σ_{inst} of the instrument noise for the gradiometer is ± 3 E. Thus, ϕ_{inst} consists of diagonal elements with variances of $\sigma_{inst}^2 = 9\,\mathrm{E}^2$. There is no correlation between the geologic background gradient, Γ_{33}, and the observation noise. The Table 4.6 presents the estimated results from the SA algorithm using these weights in the cost function.

Table 4.6: The depth and location estimations using one gradiometer survey profile L31 plus simulated signal of Γ_{33}, using weights in the cost function, true y_0 =0.0 m

True Depth d [m]	Estimated Location-y_0 [m]	Estimated Depth-d [m]
60	-2.8	59
80	-4.2	78
100	-5.5	96
150	-7.5	142
200	-3.0	192
250	23.8	271

Whenever a weight matrix is considered in the cost function, the estimated location and depth parameters are much better in comparison to the results provided in Table 4.3. It can be concluded that, if weights are available and used in the cost function, it gives better estimated results. In this case, one observation profile is adequate to locate the anomaly.

To investigate whether the anomaly can be better located by having additional observations, a second profile from the gradiometer survey is used, as illustrated in Figure 4.13.

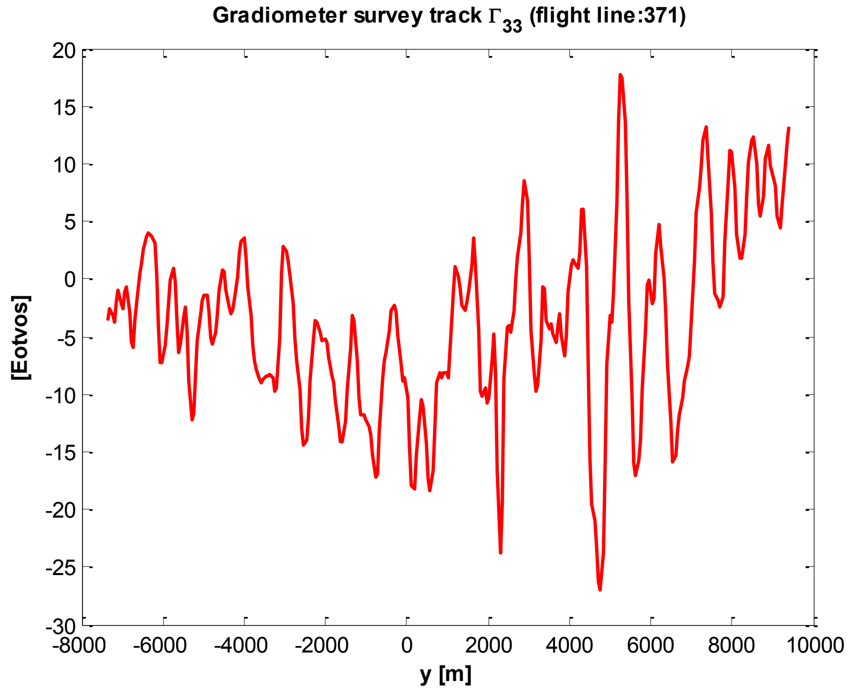


Figure 4.13: Gradiometer survey data of the second chosen track (L371)

The data interval for this profile is about 55 m. There are 305 points along the profile. Using additional observations improves the estimates, particularly for the cases where the depth is 200 m and 250 m.

Table 4.7: Summary of depth and location estimation using two gradiometer survey tracks (L31 & L371) plus simulated signal of Γ_{33}, true y_0 =0.0 m

True Depth d [m]	**Estimated Location-y_0 [m]**	**Estimated Depth-d [m]**
60	3.2	61
80	5.5	82
100	8.8	103
150	22.6	158
200	46.5	213
250	81.1	266

In this case, the anomaly can be located except for the last case where the location estimation is not good enough since the estimated location of the anomaly is about 31m away from the side of the prism facing north in the y-direction. Therefore, the anomaly may not be located accurately in this case, although the depth estimation is adequate. Another gradiometer survey track which is further away from the first one is finally used as shown in Figure 4.14. There are 325 points along this profile, and its data interval is about 51 m.

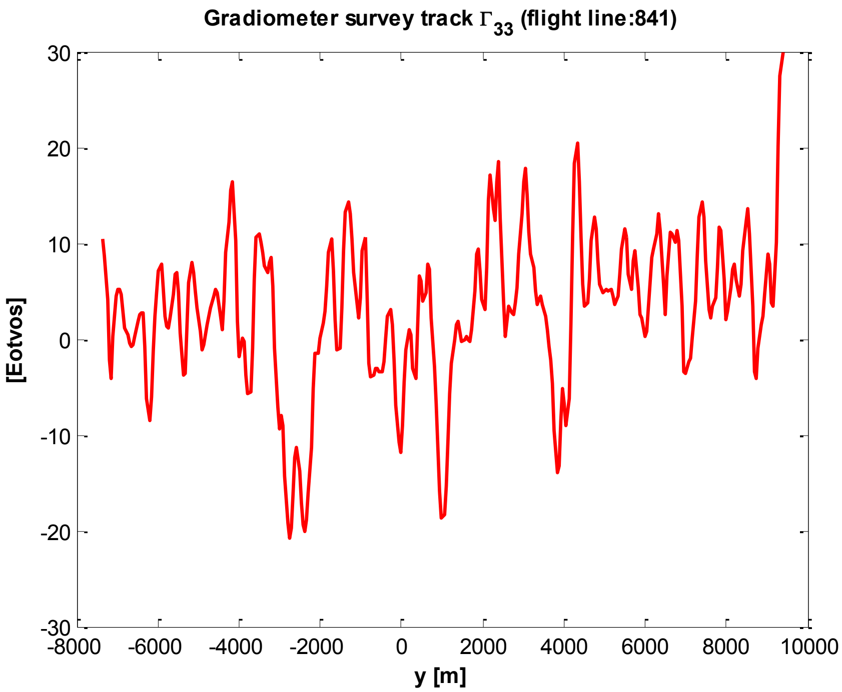


Figure 4.14: Gradiometer survey data along the third chosen track (L841)

The Table 4.8 provides the estimated location and depth for the six different depths. In these cases, both the depth and location estimates are very good.

Table 4.8: Summary of depth and location estimation using two gradiometer survey tracks (L31 & L841) plus simulated signal of Γ_{33}, true y_0=0.0 m

True Depth d [m]	**Estimated Location-y_0 [m]**	**Estimated Depth-d [m]**
60	-0.1	58
80	0.4	78
100	1.2	97
150	5.6	147
200	14.1	197
250	27.5	248

In addition, computations were performed for the case where the three tracks are chosen close to each other. For this, L761, L801 and L841 were used, where L761 and L801 are approximately 1 km, resp, 0.5 km away from L841.

Table 4.9: Summary of depth and location estimation using three gradiometer survey tracks (L761, L801 & L841) plus simulated signal of Γ_{33}, true y_0=0.0 m

True Depth d [m]	Estimated Location-y_0 [m]	Estimated Depth-d [m]
60	-1.02	59
80	-1.45	79
100	-1.91	98
150	-2.83	147
200	-2.84	196
250	-1.36	246

Comparing the results of the Tables 4.7, 4.8 and 4.9, the location estimation is improved, especially for the last two depths, when three survey tracks were used to estimate the prism location and depth. This indicates that, if more than two tracks are used, or when the number of observation points is increased, these parameters for all six different depth cases can be estimated sufficiently well.

4.1.2 Least-Squares Solution (LESS) within the Gauss-Helmert Model

The *nonlinear* Gauss-Helmert Model (GHM) is represented by

$$b(\underset{n\times 1}{Y} - \underset{n\times 1}{e}, \underset{m\times 1}{\Xi}) = 0, \qquad e \sim (0, \sigma_o^2 \underset{n\times n}{P^{-1}}), \tag{4.11}$$

where $b: R^{n+m} \longrightarrow R^{r+m}$ is a multivariate *nonlinear* function, Y is the $n\times 1$ vector of observations, e is the $n\times 1$ vector of random errors, Ξ is the $m\times 1$ unknown parameter vector, P is the symmetric positive-definite $n\times n$ *weight matrix*, $Q = P^{-1}$ is the $n\times n$ *cofactor matrix*, σ_o^2 is the unknown variance component, $\Sigma = \sigma_o^2 Q$ is the *covariance matrix* for the errors, n is the number of observations, m is the number of parameters, and r is the redundancy number in the model ($r = n - m$). By denoting μ as the $n\times 1$ expected observation vector (Schaffrin and Snow, 2010),

$$\mu = Y - e = E\{Y\}, \tag{4.12}$$

the Least-Squares objective function for the model in eq. (4.11) is minimized according to

$$e^T P e = (Y-\mu)^T P (Y-\mu) = \text{min.} \quad \text{subject to} \qquad b(Y-e,\Xi) = 0. \tag{4.13}$$

The Taylor series expansion of $b(Y-e,\Xi)=0$ about Ξ_o and μ_o, by neglecting the higher order terms, leads to

$$\left.\frac{\partial b(\mu,\Xi)}{d\mu^T}\right|_{\mu_o,\Xi_o}\cdot(\mu-\mu_o)+\left.\frac{\partial b(\mu,\Xi)}{d\Xi^T}\right|_{\mu_o,\Xi_o}\cdot(\Xi-\Xi_o)+b(\mu_o,\Xi_o)=0\,. \tag{4.14}$$

By replacing μ_o and Ξ_o with μ_j and Ξ_j (j=0,1,…), and by substituting $\mu = Y - e$ in the term $\mu - \mu_j$ in eq. (4.14),

$$\mu-\mu_j=-e+(Y-\mu_j)\,, \tag{4.15}$$

and after inserting this term into the expansion given in eq. (4.14) , the identity

$$\left.\frac{\partial b(\mu,\Xi)}{d\mu^T}\right|_{\mu_j,\Xi_j}\cdot(-e)+\left.\frac{\partial b(\mu,\Xi)}{d\Xi^T}\right|_{\mu_j,\Xi_j}\cdot(\Xi-\Xi_j)+\left.\frac{\partial b(\mu,\Xi)}{d\mu^T}\right|_{\mu_j,\Xi_j}\cdot(Y-\mu_j)+b(\mu_j,\Xi_j)=0. \tag{4.16}$$

is obtained. After defining

$$\xi_{(j+1)}:=\Xi-\Xi_j \tag{4.16a}$$

$$\underset{(r+m)\times m}{A^{(j)}}:=-\left.\frac{\partial b(\mu\ ,\Xi)}{\partial\Xi^T}\right|_{\mu_j,\Xi_j} \tag{4.16b}$$

$$\underset{(r+m)\times n}{B^{(j)}}:=\left.\frac{\partial b(\mu,\Xi)}{\partial\mu^T}\right|_{\mu_j,\Xi_j} \tag{4.16c}$$

$$\underset{(r+m)\times 1}{w_j}:=b(\mu_j,\Xi_j)+B^{(j)}(Y-\mu_j)\,. \tag{4.16d}$$

the linearized GHM is expressed as

$$w_j=A^{(j)}\xi_{(j+1)}+B^{(j)}e,\qquad e\ \sim\ (0,\sigma_o^2P^{-1})\ . \tag{4.17}$$

The solution is performed iteratively as follows (Snow, 2012):

$$\hat{\xi}_{(j+1)}=\left[(A^{(j)})^T\left(B^{(j)}P^{-1}(B^{(j)})^T\right)^{-1}A^{(j)}\right]^{-1}(A^{(j)})^T\left(B^{(j)}P^{-1}(B^{(j)})^T\right)^{-1}w_j \tag{4.18}$$

with the residual vector

$$\tilde{e}^{(j+1)}=P^{-1}(B^{(j)})^T\left(B^{(j)}P^{-1}(B^{(j)})^T\right)^{-1}[w_j-A^{(j)}\hat{\xi}_{(j+1)}] \tag{4.19}$$

Pope (1972) indicates some pitfalls on updating w_j $A^{(j)}$, $B^{(j)}$, $\hat{\Xi}_{j+1}$ and $\hat{\mu}_{j+1}$ during the execution of the algorithm. For example, $\hat{\mu}_{j+1}$ is updated by subtracting the most recent predicted error vector $\tilde{e}^{(j+1)}$ from the observation vector *Y*, not from the adjusted observation vector obtained from the previous iteration, while $\hat{\Xi}_{j+1}$ is updated by adding the estimated incremental parameter vector $\hat{\xi}_{(j+1)}$ to the vector $\hat{\Xi}_j$, which is obtained from the previous iteration. One should notice that, at the first iteration for *j*=0, $B^{(o)}(Y-\mu_o)$ in eq. (4.16d) drops out, due to $\mu_o \mathrel{\hat{=}} Y$. However, in subsequent iterations, the points of expansion (Ξ_j, μ_j) are set equal to the solution from the previous iteration. Therefore, for subsequent iterations the initial values are numerically defined by

$$\hat{\Xi}_{j+1} = \Xi_j + \hat{\xi}_{(j+1)}, \tag{4.20}$$

$$\hat{\mu}_{j+1} = Y - \tilde{e}^{(j+1)}, \tag{4.21}$$

and one should consider the extra term in the w_j vector which amounts to

$$w_j = b(\mu_j, \Xi_j) + B^{(j)}(Y - \mu_j) = b(Y, \Xi_j)\,. \tag{4.22}$$

This procedure is iterated until the change in $\hat{\Xi}_{j+1}$ and $\tilde{e}^{(j+1)}$ is smaller than the given termination criteria, δ and ε:

$$\left\| \hat{\xi}_{(j+1)} \right\| < \delta \quad \text{and} \quad \left\| \tilde{e}^{(j+1)} - \tilde{e}^{(j)} \right\| < \varepsilon \quad . \tag{4.23}$$

Linearization

The aim is to estimate the location, y_0, and the depth, *d*, of the prism using the LESS as explained above. The observations are the vertical gravitational gradients, Γ_{33}, observed at some points above the earth's surface. By denoting the observables as μ, the observation equation can be written as

$$\Gamma_{33} = \mu + e, \qquad\qquad e \sim (0, \sigma_o^2 I)\,. \tag{4.24}$$

The nonlinear model according to eq. (4.11) reads

$$b(\mu, \Xi) = -\mu - G\rho \arctan \frac{(x-x')(y-y_0-y')}{(z+d-z')r} \Bigg|_{x'=-a/2}^{a/2} \Bigg|_{y'=-b/2}^{b/2} \Bigg|_{z'=-c/2}^{c/2} = 0\,. \tag{4.25}$$

where

$$r=\sqrt{\left(x-x'\right)^2+\left(y-y_0-y'\right)^2+\left(z+d-z'\right)^2}\,.$$

The equation (4.25) is linearized as in eq. (4.16) as follows: The derivative with respect to the location, y_0, is

$$\left(\frac{\partial\Gamma_{33}}{\partial y_0}\right)=\left(-G\rho\left(-\frac{(x-x')}{r(z+d_0-z')}+\frac{(y-y_0-y')^2(x-x')}{r^3(z+d_0-z')}\right)\cdot\frac{1}{1+u^2}\right)\Bigg|_{x'=-a/2}^{a/2}\Bigg|_{y'=-b/2}^{b/2}\Bigg|_{z'=-c/2}^{c/2} \tag{4.26}$$

where

$$u=\frac{\left(x-x'\right)\left(y-y_0-y'\right)}{\left(z+d-z'\right)\sqrt{\left(x-x'\right)^2+\left(y-y_0-y'\right)^2+\left(z+d-z'\right)^2}}$$

which equals to

$$=-G\rho\left(\begin{array}{l}-\dfrac{(x-x')(z+d_0-z')r}{(z+d_0-z')^2r^2+(y-y_0-y')^2(x-x')^2}+\\ \qquad+\dfrac{(y-y_0-y')^2(x-x')(z+d_0-z')}{(z+d_0-z')^2r^3+r(y-y_0-y')^2(x-x')^2}\end{array}\right)\Bigg|_{x'=-a/2}^{a/2}\Bigg|_{y'=-b/2}^{b/2}\Bigg|_{z'=-c/2}^{c/2} \tag{4.27}$$

Rearranging the terms inside the parentheses in eq. (4.27) leads to

$$\left(\frac{\partial\Gamma_{33}}{\partial y_0}\right)=-G\rho\left(-\frac{(x-x')(z+d_0-z')r^2+(y-y_0-y')^2(x-x')(z+d_0-z')}{(z+d_0-z')^2r^3+r(y-y_0-y')^2(x-x')^2}\right)\Bigg|_{x'=-a/2}^{a/2}\Bigg|_{y'=-b/2}^{b/2}\Bigg|_{z'=-c/2}^{c/2} \tag{4.28}$$

The derivative with respect to the depth, d , is

$$\left(\frac{\partial\Gamma_{33}}{\partial d}\right)=\left(-G\rho\left(-\frac{(x-x')(y-y_0-y')}{r(z+d_0-z')^2}-\frac{(x-x')(y-y_0-y')}{r^3}\right)\cdot\frac{1}{1+u^2}\right)\Bigg|_{x'=-a/2}^{a/2}\Bigg|_{y'=-b/2}^{b/2}\Bigg|_{z'=-c/2}^{c/2} \tag{4.29}$$

where

$$u=\frac{(x-x')(y-y_0-y')}{(z+d-z')\sqrt{(x-x')^2+(y-y_0-y')^2+(z+d-z')^2}}$$

which equals to

$$=-G\rho\left(\begin{array}{c}\dfrac{-(x-x')(y-y_0-y')(z+d_0-z')^2}{(z+d_0-z')^2r^3+r(y-y_0-y')^2(x-x')^2}- \\ -\dfrac{r(x-x')(y-y_0-y')}{(z+d_0-z')^2r^2+(y-y_0-y')^2(x-x')^2}\end{array}\right)\Bigg|_{x'=-a/2}^{a/2}\Bigg|_{y'=-b/2}^{b/2}\Bigg|_{z'=-c/2}^{c/2} \tag{4.30}$$

Rearranging the terms inside the parentheses in eq. (4.30) leads to

$$\left(\frac{\partial\Gamma_{33}}{\partial d}\right)=-G\rho\left(\frac{-(x-x')(y-y_0-y')(z+d_0-z')^2-r^2(x-x')(y-y_0-y')}{(z+d_0-z')^2r^3+r(y-y_0-y')^2(x-x')^2}\right)\Bigg|_{x'=-a/2}^{a/2}\Bigg|_{y'=-b/2}^{b/2}\Bigg|_{z'=-c/2}^{c/2} \tag{4.31}$$

According to the linearized model given in eq. (4.16), the vector $(w_j)_{310\times 1}$ is given as

$$b(\mu_j,\Xi_j)=w_j=-\mu_j^0-G\rho\arctan\frac{(x-x')(y-(y_0)_0-y')}{(z+d_0-z')r}\Bigg|_{x'=-a/2}^{a/2}\Bigg|_{y'=-b/2}^{b/2}\Bigg|_{z'=-c/2}^{c/2}+B^{(j)}(\Gamma_{33}-\mu_j^0),$$

In addition, there is

$A^{(j)}=-\begin{bmatrix}\dfrac{\partial\Gamma_{33}}{\partial(y_0)_0} & \dfrac{\partial\Gamma_{33}}{\partial d_0}\end{bmatrix}_{310\times 2}$ unit in [E/m] ; $B^{(j)}=-I_{310\times 310}$; $P=I_{310\times 310}$;

$\hat{\xi}_{j+1} = \Xi - \Xi_j$ with $\Xi := \begin{bmatrix} y_0 \\ d \end{bmatrix}$ unit in m.

Firstly, the depth parameter is estimated for the six different depth cases where $A^{(j)} = -\left[\frac{\partial \Gamma_{33}}{\partial d_0}\right]_{310\times 1}$; $B^{(j)} = -I_{310\times 310}$; $P = I_{310\times 310}$ and $\Xi := d$. The estimated parameters are provided in Table 4.10.

Table 4. 10: LESS within the Gauss-Helmert Model in case $P=I$, termination criterion $\delta = 10^{-4}$. True location y_0=0.0 m

True depth d [m]	**Initial Depth d_0 [m]**	**No. of iterations**	**Estimated Depth-d [m]**
60	140	8	62
80	170	8	84
100	200	8	108
150	2	11	172
200	500	11	246
250	600	9	325

From the Table 4.10, it can be concluded that the depth parameter is estimated even if the initial values for the depth parameter is chosen far away from the actual value. Secondly, the location and depth parameters are estimated for the six different depth cases. It is possible to estimate the location and depth parameters when the initial values for the parameters are chosen appropriately. It is found that the least-squares solution is highly dependent on the starting values in this case. If the initial value for the location parameter is chosen far away from the true value, the LESS does not converge to the global minimum of the cost function. This is due the fact that the cost function has local minima with respect to the location parameter as illustrated in Figure 4.9. If the initial value for the location parameter is chosen within the large well of the cost function, the LESS gives the same results that are computed from the SA algorithm (see Table 4.3).

Table 4.11: LESS within the Gauss-Helmert Model in case *P=I*, termination criterion $\delta = 10^{-4}$. True location y_0=0.0 m

True depth [m]	Initial Location $(y_0)_0$[m]	Initial Depth d_0 [m]	No. of iterations	Estimated Location-y_0 [m]	Estimated Depth-d [m]
60	10	150	8	3.5	62
80	15	190	9	6.6	85
100	30	190	9	11.2	108
150	35	210	10	33.3	173
200	40	250	11	79.0	241
250	50	300	10	146.4	292

Finally, the weighted LESS is performed to estimate the location and depth parameters. The weights are computed by inverting the covariance matrix given in eq. (4.10). Similarly, the LESS gives the same results obtained from the SA algorithm (see Table 4.6) if the initial values for the parameters are chosen appropriately. Table 4.12 presents the weighted LESS for the location and depth parameter.

Table 4.12: LESS within the Gauss-Helmert Model in case $P = \Sigma^{-1}$, termination criterion $\delta = 10^{-4}$. True location y_0=0.0 m

True depth [m]	Initial Location $(y_0)_0$[m]	Initial Depth d_0 [m]	No. of iterations	Estimated Location-y_0 [m]	Estimated Depth-d [m]
60	20	100	6	-2.8	59
80	30	150	9	-4.2	78
100	15	180	9	-5.5	96
150	35	200	9	-7.5	142
200	40	300	12	-3.0	192
250	45	320	23	23.8	271

b) The anomaly crosses the survey tracks at a certain orientation angle

(i) In this case, the simulated observations of Γ_{33} are used to estimate the depth, the location, and additionally the orientation angle of the prism with the assumption that there is no geologic background**.** The prism is rotated around the z-axis with an angle α as illustrated in Figure 4.15.

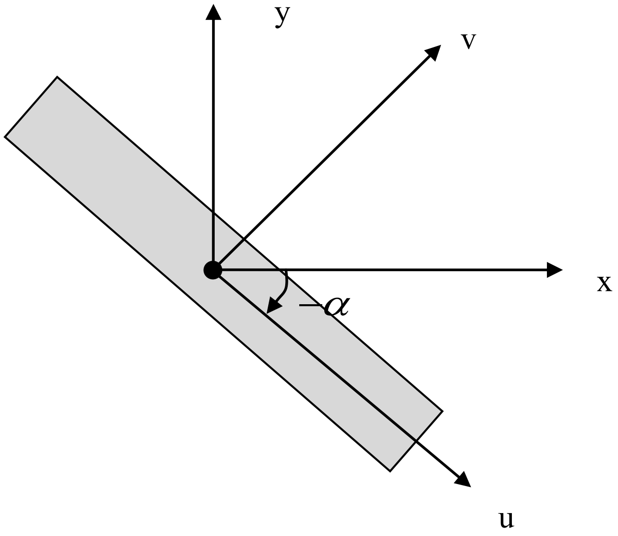


Figure 4.15: Rotated rectangular prism

This orientation is described by the following rotation matrix,

$$R_3 = \begin{bmatrix} \cos(-\alpha) & \sin(-\alpha) & 0 \\ -\sin(-\alpha) & \cos(-\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.32}$$

To compute the gravitational gradients, firstly the coordinates of the measurement points are transformed into the u-system by

$$\underline{u} = R_3(-\alpha)\underline{x}\,, \tag{4.33}$$

where, $\underline{u} = (u, v, w)$ is the system obtained by rotating the x- system so that the axes of the u system are parallel to the prism's sides. Replacing $\underline{x} = (x, y, z)$ by $\underline{u} = (u, v, w)$ and referring the limits of integration to the u-system, $u' = [-a/2, a/2], v' = [-b/2, b/2]$, $w' = [-c/2, c/2]$, the same formulas as in eq. (4.1) are used to compute the gravitational gradient tensor, Γ^u, in the u-system. To compute the gradient tensor in the original x-system, the following transformation is applied:

$$\Gamma^x = R_3(-\alpha)^T \Gamma^u R_3(-\alpha). \tag{4.34}$$

To simulate one observation profile, a prism is chosen with the following parameters: width b=100 m, length a=16 km, height c=100 m, depth d=150 m, orientation angle $\alpha = 40^{\circ}$, and density contrast $\rho = -2670$ kg/m^3.

For this application, the search domains for depth (d), location (y_0) and orientation angle (α) are defined as 1m<d<300 m, -7300 m<y_0<9500 m, and $0^{\circ} < \alpha < 90^{\circ}$, respectively. In this case, the SA algorithm's parameters are chosen as: initial temperature T_0=500 [E^2], N_T=50, τ_T =0.85, and L_δ=20. At each temperature $N_T \cdot L_\delta \cdot m$=50*20*3 iterations have been performed. Figure 4.16 shows the gravitational gradients due to the prism as simulated at 310 points at an average altitude of 84 m along the survey track L31; and Table 4.13 presents the estimated parameters for that anomaly.

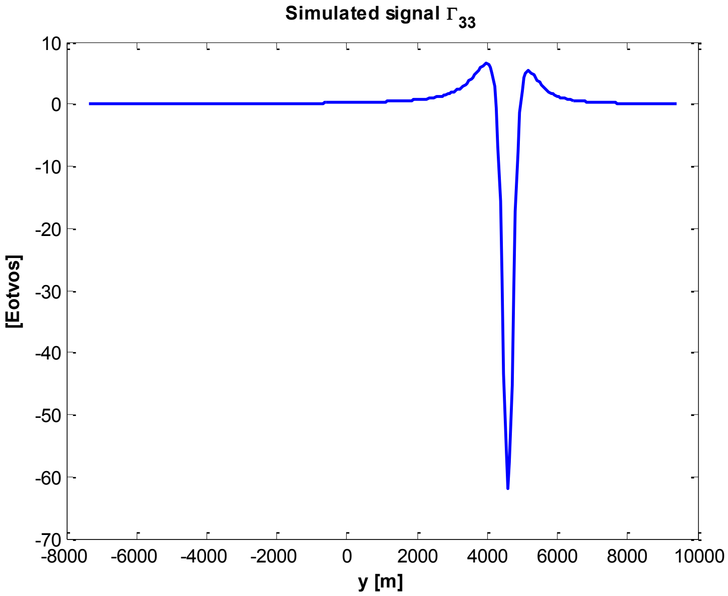


Figure 4.16: Sought signal profile

Table 4. 13: Estimated parameters of the prism using no geologic background

Location-y_0 [m]	Depth (d) [m]	Orientation Angle (α) [$^{\circ}$]	(Φ) Cost Function [Eotvos^2]
-0.9×10^{-11}	150	40.0	0.2×10^{-23}

Again, the prism parameters are perfectly estimated. It should be noted that the location parameter refers to the origin of the prism (that is, its center). However, Figure 4.16 shows that the peak of the simulated signal is where the anomaly crosses the survey track L31 as illustrated in Figure 4.17.

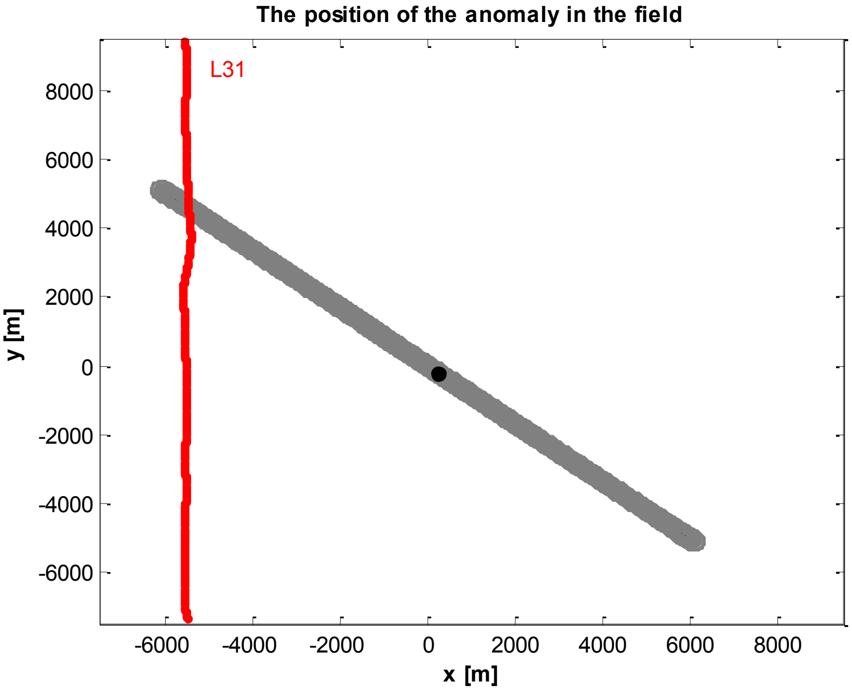


Figure 4.17: The position of the oriented anomaly in the field

Also, Figures 4.18 and 4.19 illustrate the cost function with respect to its parameters for this simulation.

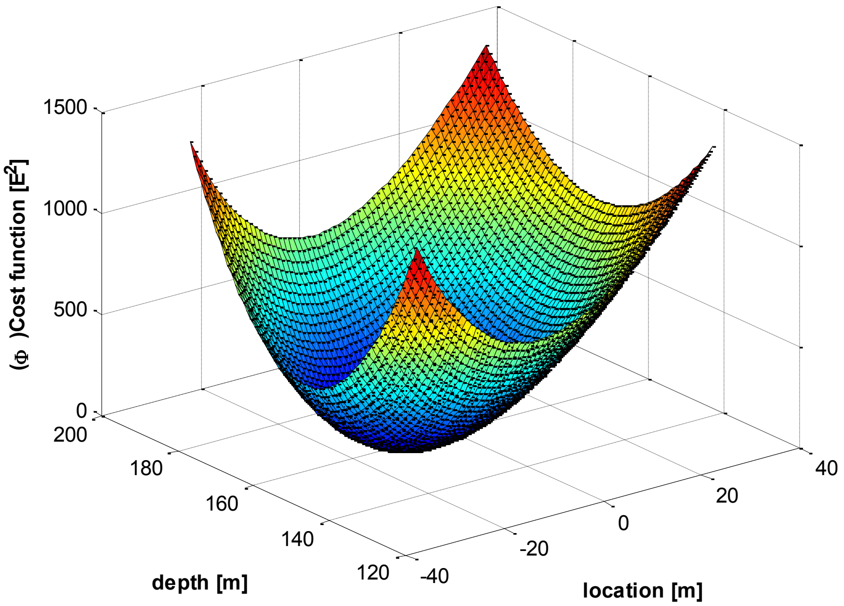


Figure 4.18: The plot of the cost function for depth *d*, and location y_0, and $\alpha = 40^{o}$

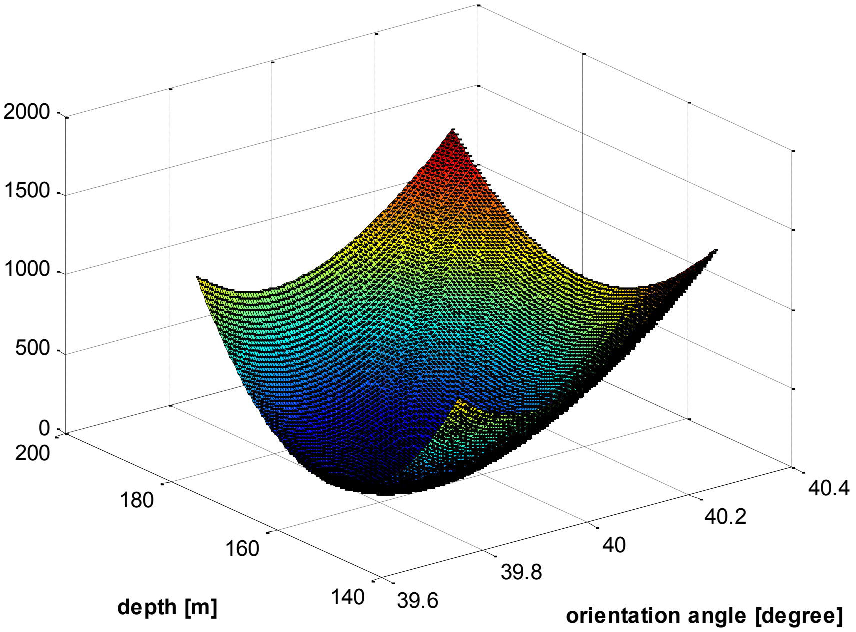


Figure 4.19: The plot of the cost function for depth *d*, and orientation angle α, and y_0=0.0 m

Plots of the profiles of the cost functions in Figures 4.18 and 4.19 at the fixed depth of 150 m over an extended domain are illustrated in Figures 4.20 and 4.21. The cost function has local minimum with respect to both the location and orientation angle parameters in this case.

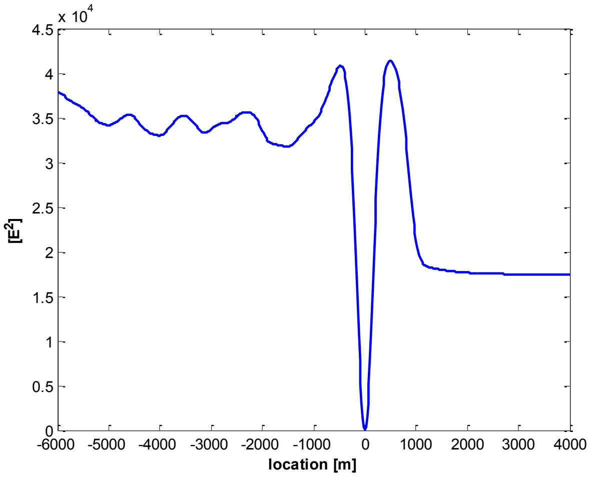


Figure 4.20: A profile of the cost function as function of location parameter at fixed depth 150 m, and orientation angle, $\alpha = 40°$

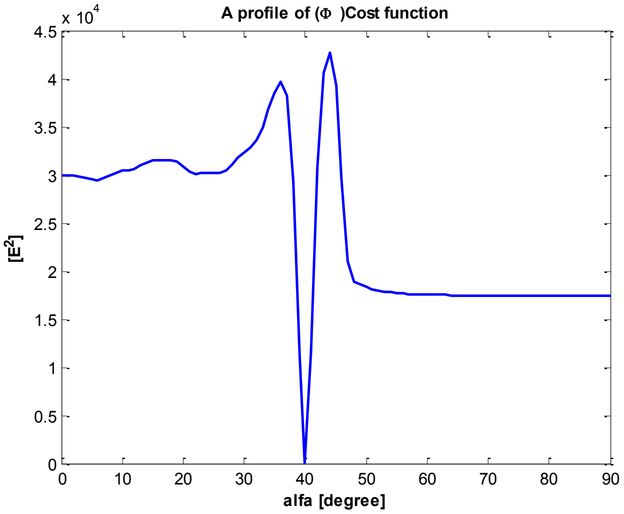


Figure 4.21: A profile of the cost function as function of orientation angle at fixed depth 150 m and location $y_0 = 0.0$ m

(ii) In these cases, each of the six simulated signals plus the gradiometer survey data that were used are shown in Figure 4.22.

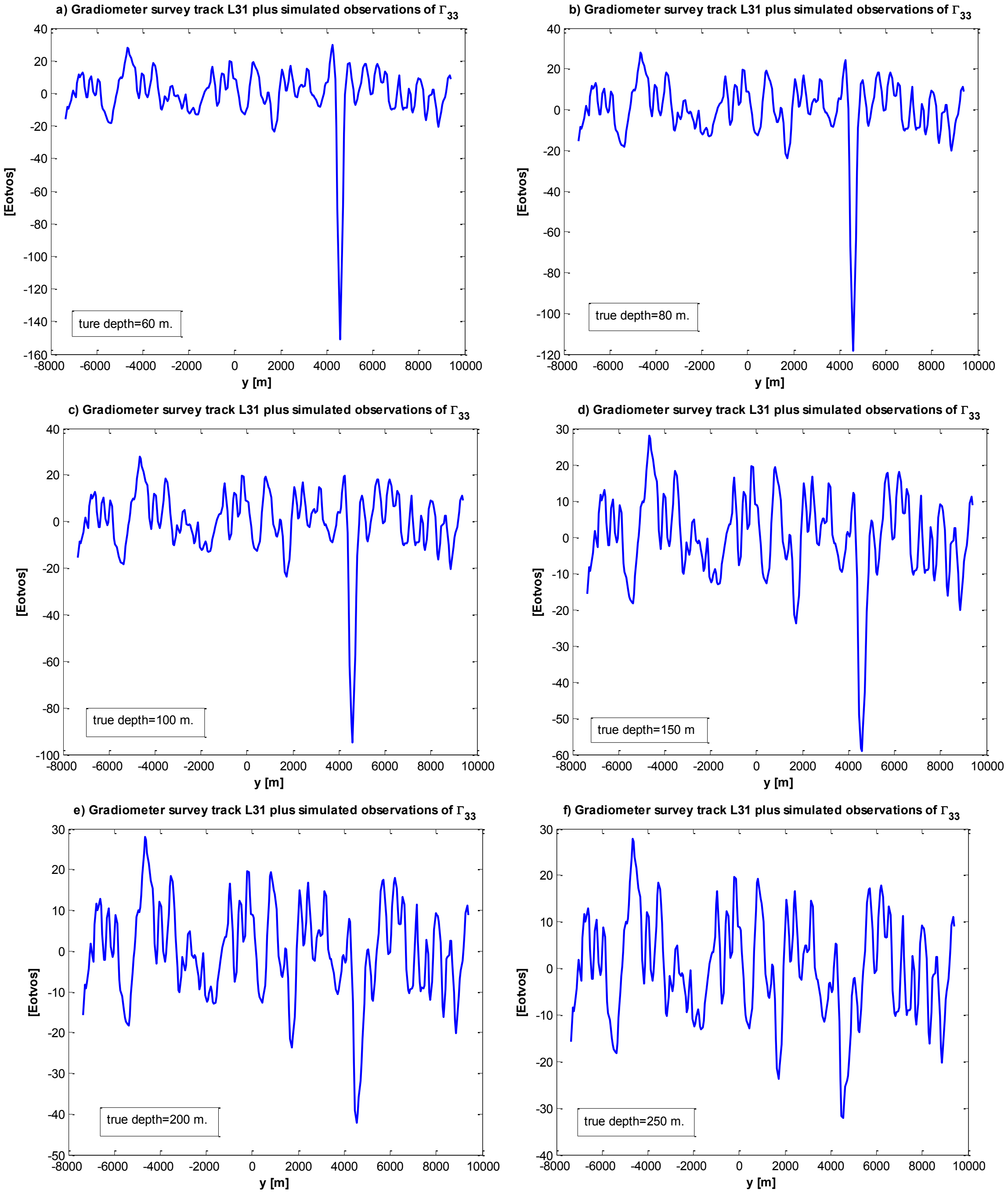


Figure 4.22: Simulated signals for six different depths 60, 80, 100, 150, 200 and 250 m, respectively, plus survey track L31, α=40°

It is not possible to obtain good estimates of the location parameter from one observation profile because of adding the observation noises to the simulated signal and estimating the orientation angle. Table 4.14 presents the estimated parameters of the prism in these cases. The search domains for the location (y_0) and orientation angle (α) are set as -7300 m<y_0<9500 m, $0^o < \alpha < 90^o$, respectively. For the first four cases, the depth search interval is 1 m<d<300 m and for the last two cases, it is 1m<d<500 m. The initial temperature is T_0=5000 [E^2], N_T=100, τ_T=0.85, and L_δ=20. At each temperature $N_T \cdot L_\delta \cdot m$=100*20*3 iterations were performed.

Table 4.14: Summary of estimated depth, location and orientation angle, using one gradiometer survey track L31 plus simulated signal of Γ_{33}, true y_0 =0.0 m and true $\alpha = 40^o$

True Depth d [m]	Estimated Location-y$_0$ [m]	Estimated Depth-d [m]	Estimated Orient. Angle (α) [^{o}]
60	144.6	61	38.8
80	225.5	81	38.2
100	309.6	101	37.5
150	458.5	153	36.3
200	346.0	210	37.3
250	-929.6	282	47.7

As can be seen from Table 4.14, while the depth estimations are good, the location estimations are not good, when considering the size of the anomaly. For example, for the first case, the estimated location is about 94 m away from the side of the prism facing north in the y-direction. Therefore, a second observation profile (Figure 4.13) in combination with the first is used to estimate the same prism parameters. When simulating the gravitational gradients along this second observation profile for the depth of 150 m, the plot of the simulated anomaly is as illustrated in Figure 4.23.

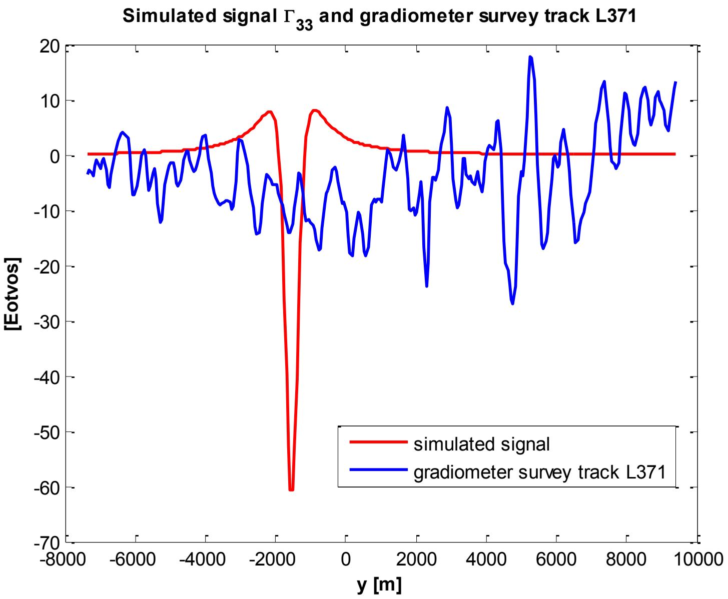


Figure 4.23: Simulated signal Γ_{33} (true depth =150 m) and survey track L371

The search domains for the location (y_0) and orientation angle (α) are set as -7300 m<y_0<9500 m, $0^o<\alpha<90^o$, respectively. For the first four cases, the depth interval is 1 m<d<300 m and, for the last two cases, the depth interval is 1m<d<500 m. The initial temperature is T_0=5000 [E^2], and N_T=100, τ_T=0.85, and L_δ=20. At each temperature $N_T \cdot L_\delta \cdot m$=100*20*3 iterations were performed. The Table 4.15 provides the estimated parameters of the prism when using two observation profiles.

Table 4.15: Summary of estimated depth, location and orientation angle, using two gradiometer survey tracks (L31 & L371) plus simulated signal of Γ_{33}, true y_0=0.0 m and true α=40^o

True Depth d [m]	**Estimated Location-y_0 [m]**	**Estimated Depth-d [m]**	**Estimated Orient. Angle (α) [^{o}]**
60	-2.7	60.2	40.007
80	-4.4	80.5	40.022
100	-6.7	101	40.045
150	-16.7	154	40.154
200	-36.1	211	40.370
250	-71.1	274	40.748

According to these results, one can conclude that, when the anomaly is oriented with an arbitrary angle, then at least two observation profiles of vertical gradients are needed to locate the anomaly in the presence of background geologic signal. For the first five cases, the SA algorithm provides enough information to locate the anomaly. Although the depth and orientation angle estimates for the depth=250m are good, the location estimation is not good enough if the size of the anomaly is considered.

If a third gradiometer survey track L841 is used to estimate the same parameters of the prism (Figure 4.24), then the location estimation for the last case is improved. Table 4.16 presents the estimated parameters of the prism using two tracks, L31 & L841. As can be seen, the estimated parameters are good enough to locate the anomaly in all six cases.

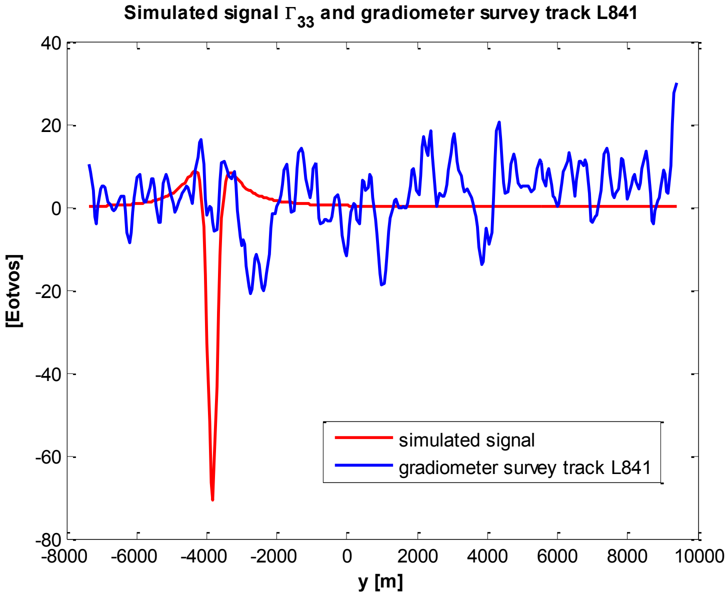


Figure 4.24: Simulated signal Γ_{33} (true depth =150 m) and survey track L841

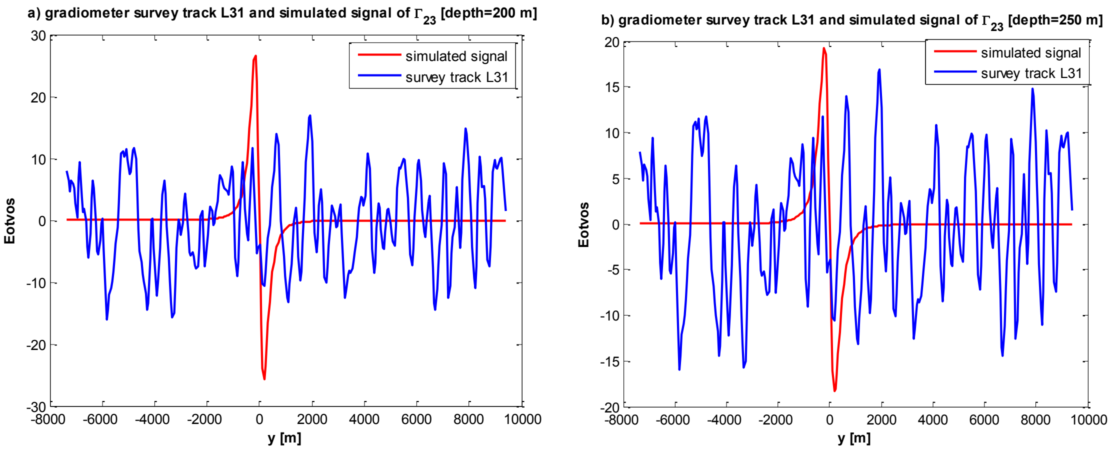
Table 4. 16: Summary of estimated depth, location and orientation angle, using two gradiometer survey tracks (L31 & L841) plus simulated signal of Γ_{33}, true y_0=0.0 m and true α=40°

True Depths d [m]	Estimated Location-y_0 [m]	Estimated Depth-d [m]	Estimated Orient. Angle (α) [°]
60	-0.2	58	39.988
80	-0.05	77	39.987
100	0.3	96	39.987
150	2.05	141	39.996
200	5.2	187	40.017
250	10.01	236	40.053

4.1.3 Subsurface Anomaly Detection from Gravity Cross-Gradients using Simulated Annealing (SA)

The aim of this section is to investigate whether the anomaly can be better located by using gradients other than the vertical-vertical gravitational gradient. In the previous section, the depth, location, and orientation angle of the prism were estimated using one, two or three observation profiles. Here, the same computations were performed but using gravitational cross-gradients, $\Gamma_{23}, \Gamma_{13}, \Gamma_{12}$. The focus is on the last two cases, presented in the previous section. These are the cases where the depths of the anomaly are 200 and 250 meters, respectively. In this case, the geologic background will also be represented by the measured gravitational cross-gradients. For implementation of the SA, the same search domains, termination criteria, annealing schedule, and prism parameters are used. Also, the same gradiometer survey profiles L31, L371 and L841 are used. In the following, the plots of the simulated signals and gradiometer survey tracks are presented for both cases where the anomaly crosses the survey tracks orthogonally or at a certain azimuthal angle; afterwards, the corresponding results will be presented.

a) The anomaly crosses to the survey tracks orthogonally: Figures 4.25. a)-b) show the plots of the gradiometer survey data and the simulated signal of Γ_{23} generated along the survey track, L31.


a) gradiometer survey track L31 and simulated signal of Γ_{23} [depth=200 m]

b) gradiometer survey track L31 and simulated signal of Γ_{23} [depth=250 m]

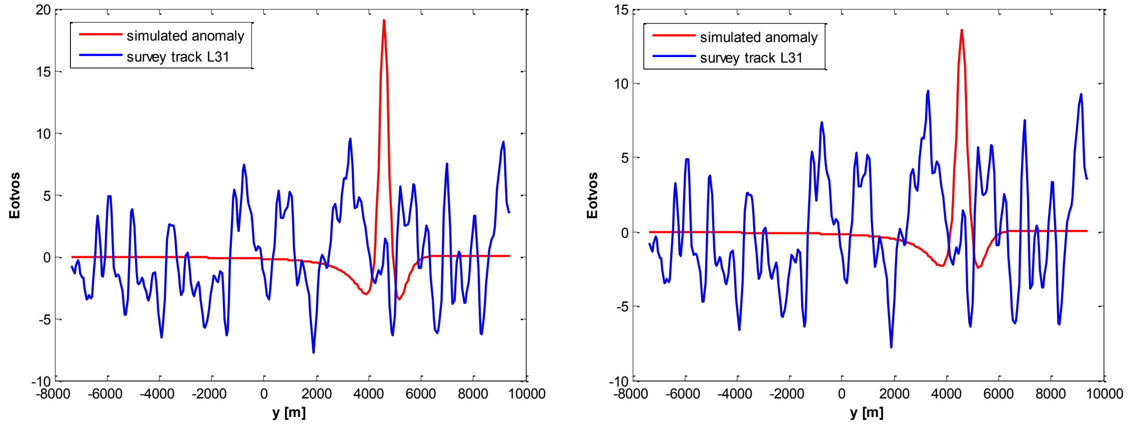
Figure 4.25: The plots of simulated signals of Γ_{23} for depths 200 m (left) and 250 m (right), and gradiometer survey track (L31)

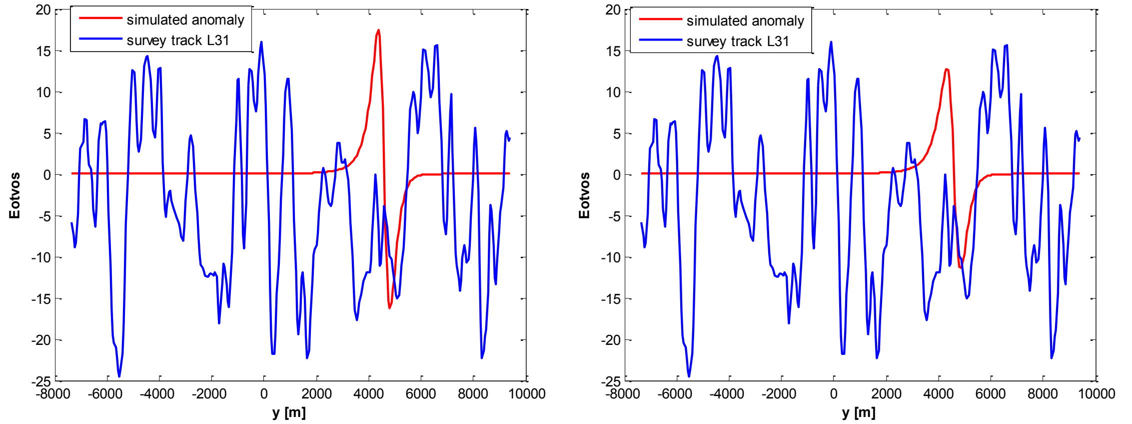
The magnitude of the gradients Γ_{13} and Γ_{12} of the simulated signal in comparison to the geologic background is small and is not shown here. Therefore, the SA algorithm is not able to locate the anomaly either from one or two observation profiles of these gradients. On the other hand, the anomaly can be located from one observation profile of Γ_{23} successfully since in this case the strength of the simulated signal is high in comparison to the geologic background. Especially, the location estimation is very good compared to the results obtained from one observation profile of Γ_{33}. Table 4.17 shows the estimated parameters of the prism using one observation profile, Γ_{23}, from gradiometer survey plus the corresponding anomaly signal.

Table 4.17: Summary of depth and location estimation using one gradiometer survey track (L31) plus simulated signal of Γ_{23}, true y_0=0.0 m

True Depth d [m]	Estimated Location-y_0 [m]	Estimated Depth-d [m]
200	-23.1	175
250	-35.0	210

b) The anomaly crosses the survey tracks at a certain azimuthal angle: The Figures 4.26. a-f) illustrate the plots of the gradiometer survey data along track L31 and simulated signals of Γ_{13}, Γ_{12}, Γ_{23}, respectively. As can be seen from Figures 4.26.a,b,c,d,e,f, the strength of the simulated signals is high in comparison to the geologic background. In this case, the location parameter is not estimated accurately from one gradiometer survey track. Therefore, the anomaly can be located from two observation profiles of gradients Γ_{12}, Γ_{13} and Γ_{23}. In this case, the estimates for location and depth are much better than the ones estimated from Γ_{33}. The location is estimated especially well from Γ_{13} gradients for the case that the depth is 250 m. However, the depth estimate is worse than the ones estimated from gradients Γ_{12} and Γ_{23}. The Table 4.18 present the estimated parameters of the prism from gradients Γ_{12}, Γ_{23} and Γ_{13}.


a) gradiometer survey track L31 and simulated observations of Γ_{12} [depth=200 m]   b) gradiometer survey track L31 and simulated observations of Γ_{12} [depth=250 m]


c) gradiometer survey track L31 and simulated observations of Γ_{13} [depth=200 m]   d) gradiometer survey track L31 and simulated observations of Γ_{13} [depth=250 m]

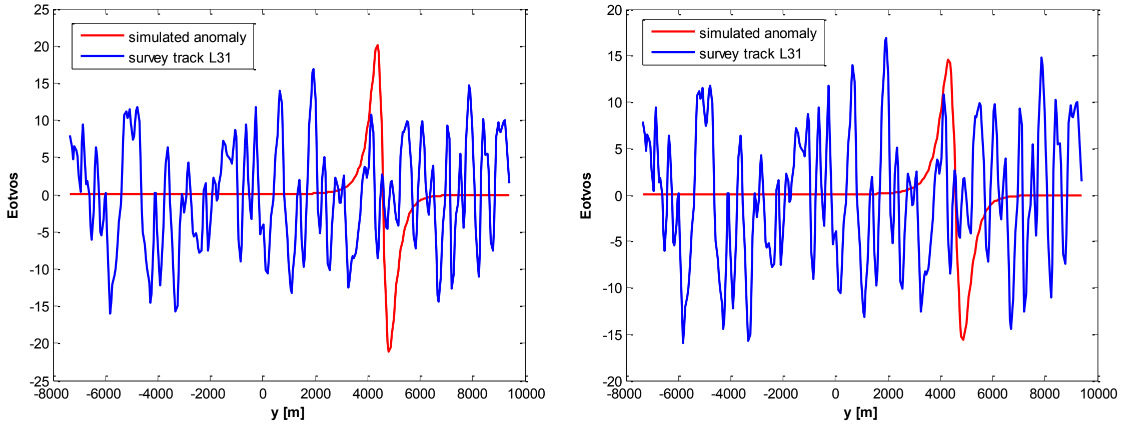

e) gradiometer survey track L31 and simulated observations of Γ_{23} [depth=200 m]   f) gradiometer survey track L31 and simulated observations of Γ_{23} [depth=250 m]

Figure 4.26: The plots of simulated signals of Γ_{12}, Γ_{13}, Γ_{23} for depths 200 m (left) and 250 m (right), and gradiometer survey track L31, α =40^{o}

Table 4.18: Summary of depth, location and orientation angle estimation, using two gradiometer survey tracks (L31 & L371) plus simulated signal of Γ_{12}, Γ_{23}, Γ_{13}, true y_0=0.0 m and true α=40°

True Depth d [m]	**Estimated Location-y_0 [m]**	**Estimated Depth-d [m]**	**Estimated Orient. Angle (α)[°]**
Γ_{12}			
200	11.1	197	39.864
250	26.2	251	39.703
Γ_{23}			
200	19.1	198	39.732
250	33.3	245	39.518
Γ_{13}			
200	10.4	208	39.983
250	7.1	262	39.995

According to the above results, one can conclude that, in comparison to the vertical gravitational gradients, the location of the prism is estimated much better from the three gravitational cross-gradients, separately. If the three gravitational cross-gradients are combined to estimate the prism parameters, the estimated results, especially for the location parameter, are much better (Table 4.19).

Table 4.19: The depth, location and orientation angle estimation, using three gradiometer survey tracks (L31, L371, L841) plus simulated signal of Γ_{12}, Γ_{23}, Γ_{13}, true y_0 =0.0 m and true α=40°

True Depths d [m]	**Estimated Location-y_0 [m]**	**Estimated Depth-d [m]**	**Estimated Orient. Angle (α)[°]**
$\Gamma_{12}, \Gamma_{13}, \Gamma_{23}$			
200	-2.3	204	40.110
250	-4.0	249	40.144

4.2 Case 2: Infinite horizontal dip-slip fault

The task is to estimate the parameters of a real fault in the Wichita uplift region from gravitational gradients. The Wichita province consists of the uplifted basement portion of Southern Oklahoma. It is bounded by major fault zones in the north and the south as illustrated in Figure 4.27.

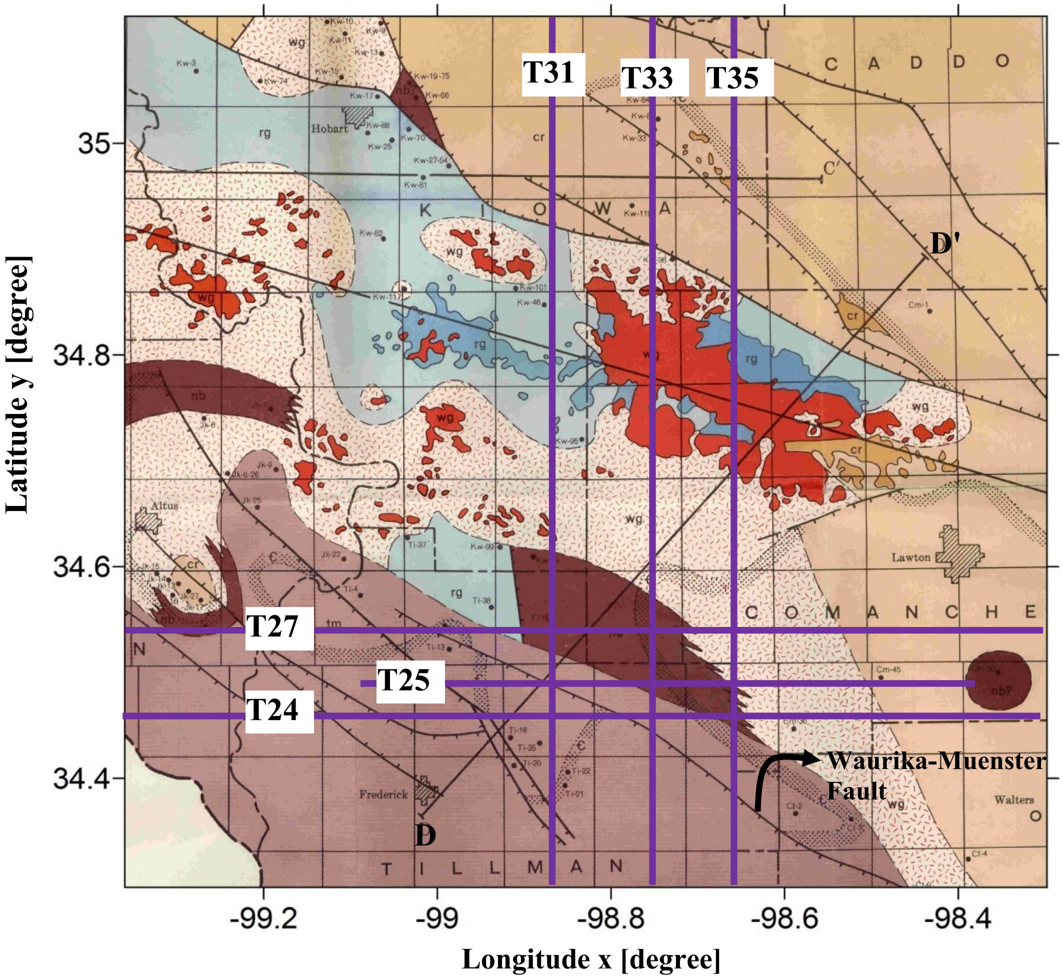


Figure 4.27: Basement geology of the Wichita uplift region (for explanation of rock types, see Ham et al., 1964, plate 1 from which this figure was extracted). The purple lines indicate gradiometer survey tracks

Gradiometer Survey System (GGSS) was flown over a large part of the Texas/Oklahoma border in 1987, including the Wichita uplift region (Jekeli, 1993). The survey was flown at approximately 1000 m above the ground with an aircraft speed of 400 km/hr. There are 19 survey tracks available, considered the best for gravity field modeling and therefore of geophysical interest. These tracks are in both the north-south and the east-west directions and data along some of these tracks are illustrated in Figures 4.28 and 4.29.

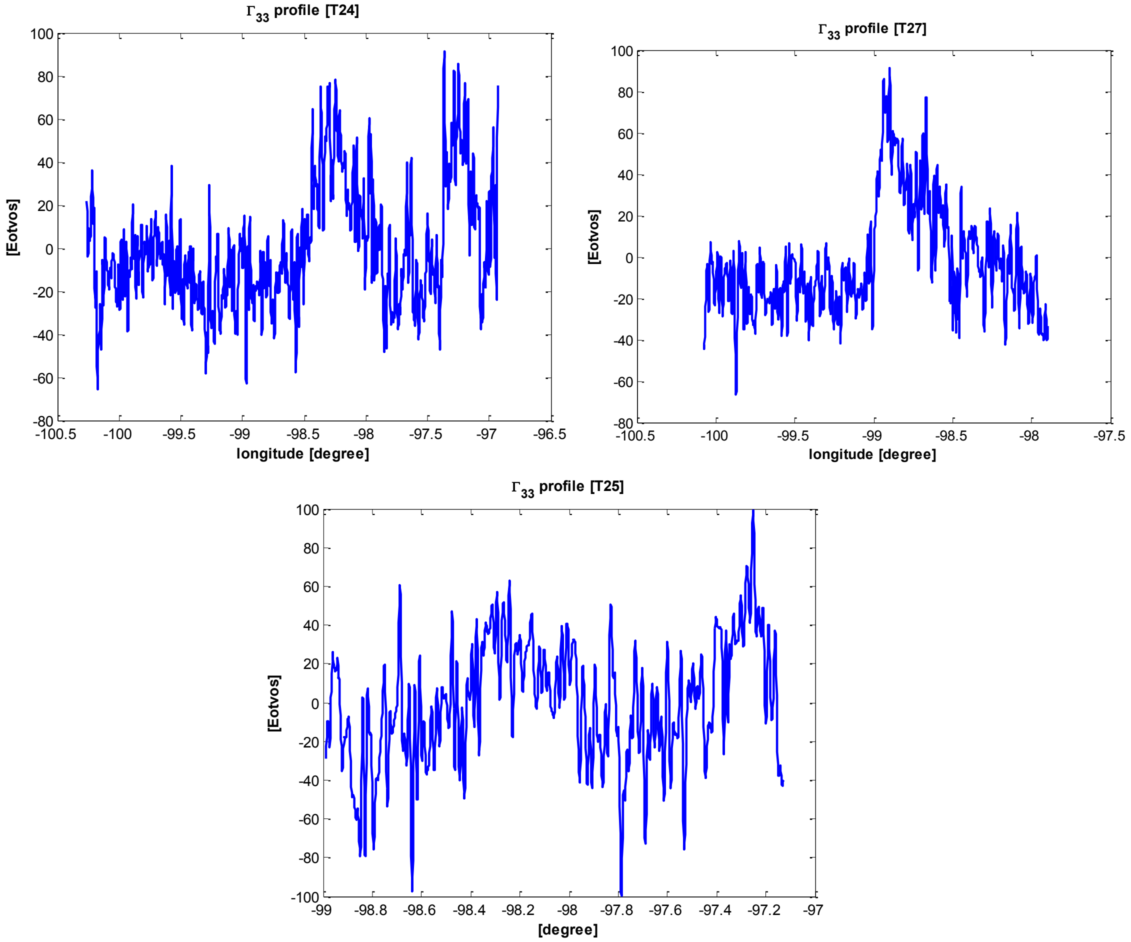


Figure 4.28: Γ_{33} survey tracks, T27, T24 and T25 in the east-west direction

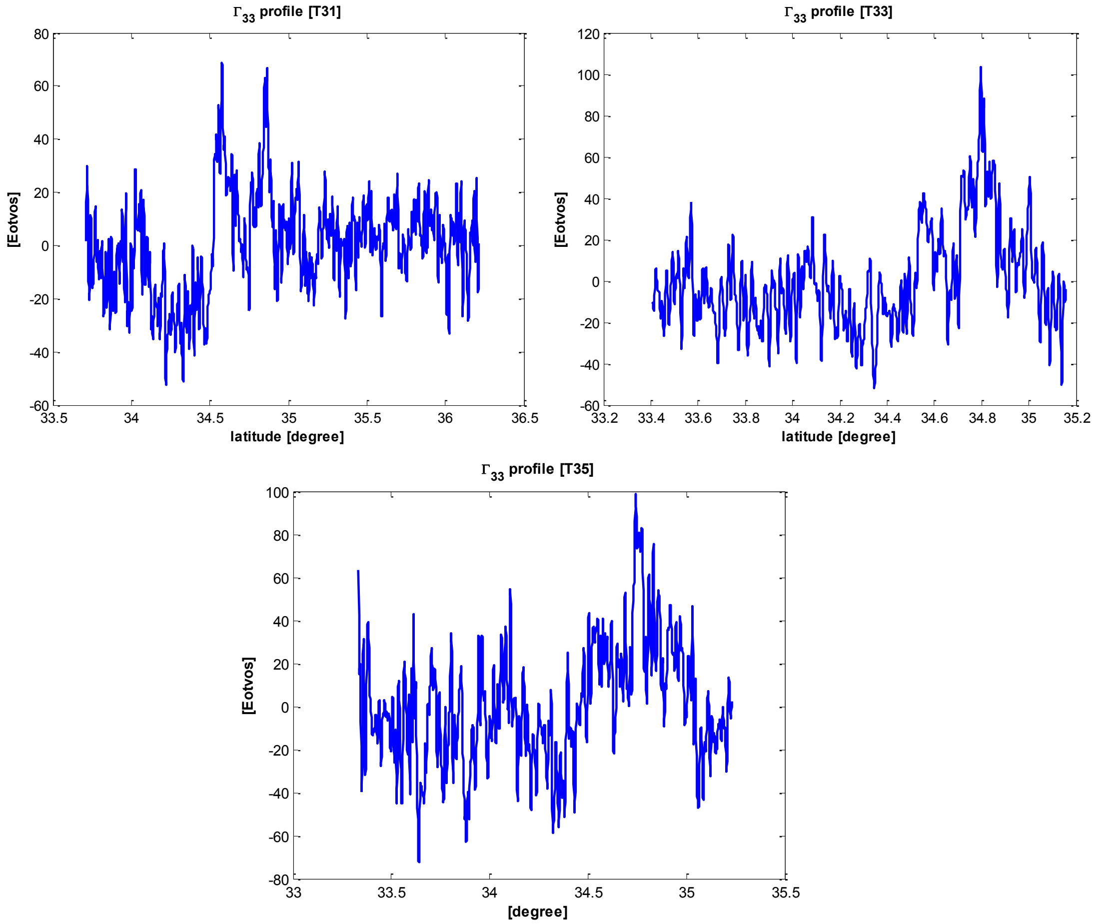


Figure 4.29: Γ_{33} survey tracks, T31, T33 and T35 in the north-south direction

The survey profiles T24, T27 and T25 include 3000, 1850 and 1575 observation points, and T31, T33 and T35 include 2533, 1775 and 1945 observation points, respectively. Basement rocks of the region were investigated in outcrops and in samples obtained from boreholes (Ham et al., 1964). Figure 4.30 shows the D-D' cross section of the Waurika-Muenster fault over the Wichita uplift region. The location of the fault, associated with the blue rectangular blocks in Figure 4.30, is assumed to be known, but its parameters should be estimated. These might include dip angle, ξ, the lower and upper plane of the western horizontal slab, z_{L1}, z_{L2}, and the lower and upper plane of the eastern horizontal slab, z_{R1}, z_{R2}.

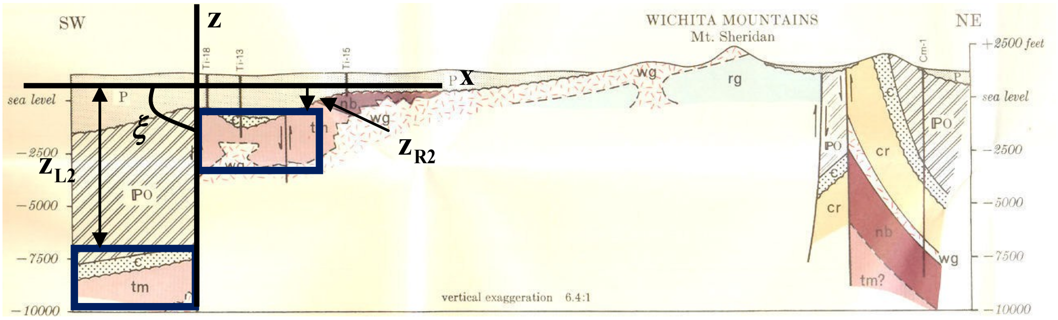


Figure 4.30: D-D' cross-section in Figure 4.27. For explanation of rock types, see Ham et al., 1964, plate 1 from which this figure was extracted

The north-south direction defines the *y*-coordinate axis, and the east-west direction defines the *x*-coordinate axis. z-coordinates denote the altitude of points which consists of ellipsoidal terrain elevation plus flight altitude. The average altitude of the measurement points is set as 1300 m, for both the north-south and the east-west survey tracks. The altitude of the GGSS data was not well determined. However, accuracy of the altitude is not considered critical for gravity gradient measurements.

The forward model is defined as an infinite horizontal dip-slip fault of constant density contrast with the assumption that, in the strike direction, the fault is extended to infinity. Figure 4.31 shows the fault model and its parameters, defined in an East-North-Up coordinate system. The fault parameters are defined as follows: ξ is the dip angle, z_{L1}, z_{L2} define the lower and upper plane of the western horizontal slab, and z_{R1}, z_{R2} define the lower and upper plane of the eastern horizontal slab.

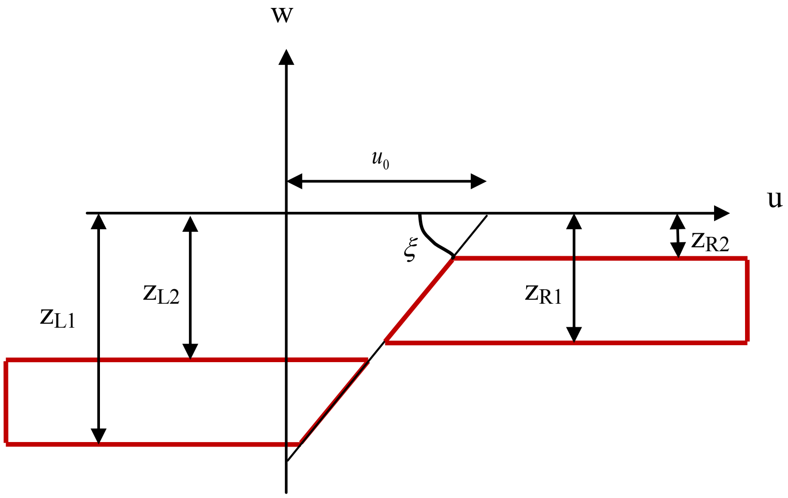


Figure 4.31: Definition of the dip-slip fault plane and associated parameters

The formulas to compute gravitational gradient effects due to such a fault are given in Appendix A:

$$
\begin{aligned}
\Gamma_{11}^{(L)} &= -2G\rho\left(\begin{array}{l}\sin\beta\cos\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln\dfrac{\sin\psi_2}{\sin\psi_1}\right)- \\ \left(\left(u-u_0\right)\cos\beta-w\sin\beta\right)\cos\beta\left(\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial w}-\theta_2\csc^2\psi_2\dfrac{\partial\theta_2}{\partial w}\right)\end{array}\right) \\
\Gamma_{13}^{(L)} &= -2G\rho\left(\begin{array}{l}\cos^2\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln\dfrac{\sin\psi_2}{\sin\psi_1}\right)+ \\ \left(\left(u-u_0\right)\cos\beta-w\sin\beta\right)\cos\beta\left(\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial u}-\theta_2\csc^2\psi_2\dfrac{\partial\theta_2}{\partial u}\right)\end{array}\right) \\
\Gamma_{33}^{(L)} &= 2G\rho\left(\begin{array}{l}\sin\beta\cos\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln\dfrac{\sin\psi_2}{\sin\psi_1}\right)- \\ \left(\left(u-u_0\right)\cos\beta-w\sin\beta\right)\cos\beta\left(\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial w}-\theta_2\csc^2\psi_2\dfrac{\partial\theta_2}{\partial w}\right)\end{array}\right) \\
\Gamma_{11}^{(R)} &= 2G\rho\left(\begin{array}{l}\sin\beta\cos\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln\dfrac{\sin\psi_2}{\sin\psi_1}\right)- \\ \left(\left(u-u_0\right)\cos\beta-w\sin\beta\right)\cos\beta\left(\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial w}-\theta_2\csc^2\psi_2\dfrac{\partial\theta_2}{\partial w}\right)\end{array}\right) \\
\Gamma_{13}^{(R)} &= 2G\rho\left(\begin{array}{l}\cos^2\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln\dfrac{\sin\psi_2}{\sin\psi_1}\right)+ \\ \left(\left(u-u_0\right)\cos\beta-w\sin\beta\right)\cos\beta\left(\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial u}-\theta_2\csc^2\psi_2\dfrac{\partial\theta_2}{\partial u}\right)\end{array}\right) \\
\Gamma_{33}^{(R)} &= -2G\rho\left(\begin{array}{l}\sin\beta\cos\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln\dfrac{\sin\psi_2}{\sin\psi_1}\right)- \\ \left(\left(u-u_0\right)\cos\beta-w\sin\beta\right)\cos\beta\left(\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial w}-\theta_2\csc^2\psi_2\dfrac{\partial\theta_2}{\partial w}\right)\end{array}\right)
\end{aligned}
\tag{4.35}
$$

where $\beta = 90^o - \xi$, $\psi_{1,2} = \theta_{1,2} - \beta$,

$$\theta_{1,2} = \arctan\frac{u-u_0-w_{1,2}\tan\beta}{w-z_{1,2}},$$

$$\frac{\partial\theta_{1,2}}{\partial u} = \frac{\cos^2\theta_{1,2}}{w-z_{1,2}},$$

$$\frac{\partial\theta_{1,2}}{\partial w} = -\frac{\sin\theta_{1,2}\cos\theta_{1,2}}{w-z_{1,2}}.$$

Since the fault is oriented with a certain angle, an orientation angle is introduced to the forward model. As shown in Figure 4.32, the orientation angle is defined with respect to the y-axis. In this case, the (x,y,z) coordinate system is rotated into (u,v,w) system by azimuth α. The fault intersects the u-axis at $-x_0\cos(\pi-\alpha)$ and the coordinates of the computation points on the x-axis in the (u,v,w) system are $u=-x\cos(\pi-\alpha)$.

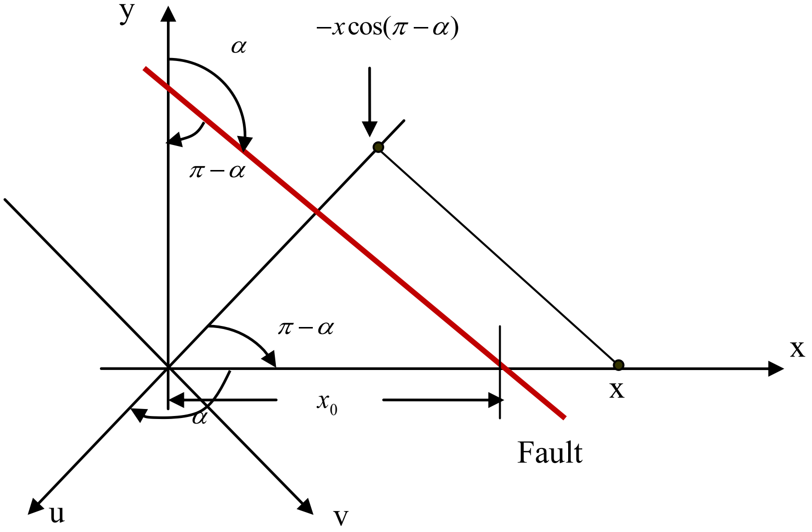


Figure 4.32: Geometry for an rotated fault crossing the x-axis

The gravitational gradient due to this fault is obtained by first calculating the gravitational gradient tensor in the (u,v,w)-system,

$$\Gamma^{(u,v,w)}=\begin{pmatrix}\Gamma_{11} & 0 & \Gamma_{13}\\ 0 & 0 & 0\\ \Gamma_{13} & 0 & \Gamma_{33}\end{pmatrix}, \tag{4.36}$$

using $-x_0\cos(\pi-\alpha)$ and $-x\cos(\pi-\alpha)$ in place of u_0 and u, respectively, in the formulas (4.35). To compute the gravitational gradient tensor in the (x,y,z)-system, the tensor is rotated,

$$\mathit{\Gamma}^{(x,y,z)}=R_3(\alpha)\mathit{\Gamma}^{(u,v,w)}R_3^T(\alpha), \tag{4.37}$$

where the rotation matrix is defined as

$$R_3(\alpha) = \begin{pmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{4.38}$$

SA is now implemented using the east-west survey tracks, after defining appropriate search domains and approximate values for the parameters to be estimated. From the D-D' cross section, it is seen that the dip angle is around 90°. However, not all the depth parameters of the fault are likely well known. Depending on how to define these parameters affects the accuracy of the estimated parameters. This can be seen from the following simple simulations of the fault model. From the cross-section, the following fault parameters are chosen with fixed values: $\xi = 85^o$; $x_0 = 0$ [m]; $z_{L2} = -3500$[m]; $z_{R2} = -1200$[m]; orientation angle $\alpha = 0^o$; the elevation of measurement points, z=0[m] and the density contrast is $\rho = 0.30\ \mathrm{g/cm}^3$. As can be seen from Figures 4.33 and 4.34, as the lower planes of the right and left horizontal slabs z_{R1}, z_{L1} change, namely the thicknesses of the right and left slab increase, the magnitude of the simulated signal becomes larger. This shows the sensitivity of the simulated signal with respect to these depth parameters. Therefore, it is difficult to estimate these depth parameters using only information taken from the D-D' cross section.

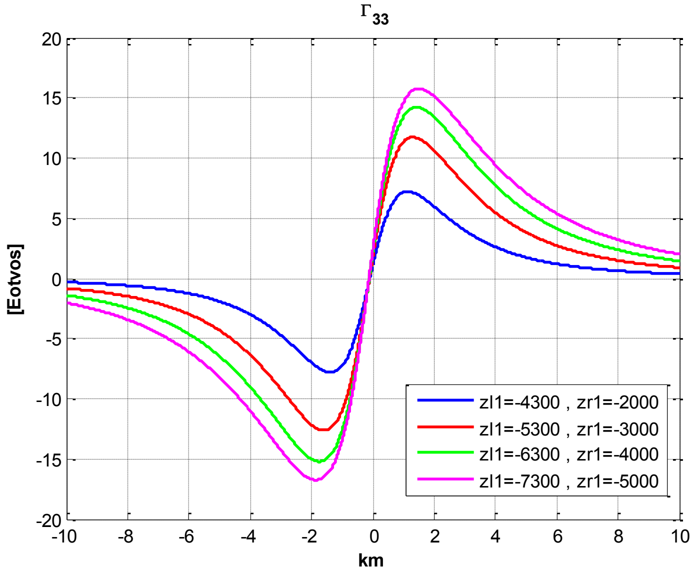


Figure 4.33: Γ_{33} gradients due to dip-slip fault by varying the lower plane of the slab

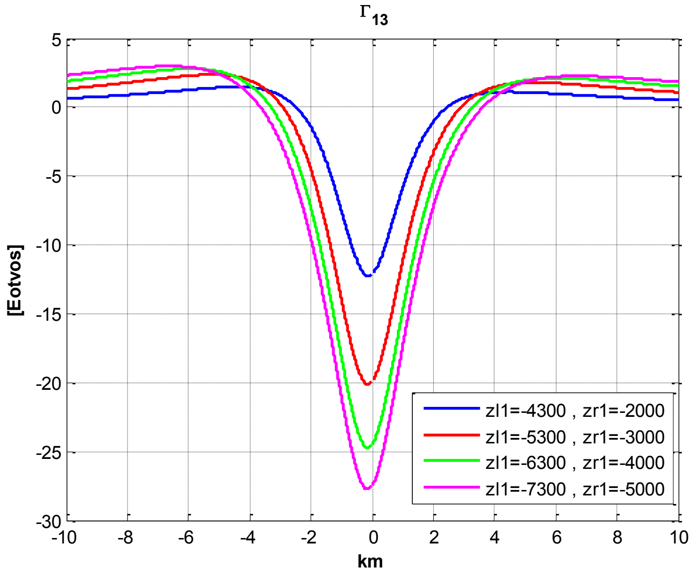


Figure 4.34: $\mathit{\Gamma}_{13}$ gradients due to dip-slip fault by varying the lower plane of the slab

From the Figures 4.27 and 4.30, approximate values for the parameters may be inferred as follows: $\xi = 90^{o}$; $x_0 = -11008.30\,[\mathrm{m}]$; $z_{R2} = -150[\mathrm{m}]$; orientation angle $\alpha = 120^{\circ}$; and according to the simulations in Figures 4.33 and 4.34, the other parameters are set as $z_{L1} = -9500\,[\mathrm{m}]$; $z_{L2} = -4500[\mathrm{m}]$; and $z_{R1} = -3500[\mathrm{m}]$. The density contrast is assumed to be $\rho = 0.5\,\mathrm{g/cm}^3$ (Robbins and Keller, 1992; Coffman, et al., 1986). The search domains for the parameters are, therefore, defined by -1 m $< z_{R2} <$ -4000 m, -3500 m$< z_{L2} <$-9500 m, $30 < \alpha < 180^{\circ}$, and $45 < \xi < 100^{\circ}$.

In the cost function as defined in eq. (4.3), ${}^{i}\Gamma_{33}^{obs}$ represents the vertical gravitational gradient from the GGSS survey, and ${}^{i}\Gamma_{33}^{pred}$ is the computed gravitational gradient using the forward model. The SA parameters are $N_T = 50$, $L_\delta = 20$, $\tau_T = 0.5$; the initial temperature is $T_0 = 9000\,[\mathrm{E}^2]$, the termination criterion is $\varepsilon =$ $1\times10^{-21}\,[\mathrm{E}^2]$, and the number of successive temperature reductions to check the termination criterion is E_ε=4.

The Table 4.20 presents the results of estimating some combinations of three fault parameters from three gradiometer survey tracks.

Table 4.20: Estimated fault parameters using T24, T27 and T25 survey tracks of Γ_{33}. The fault parameters inferred from the geologic maps are approximately ξ=90^{o} ; z_{R2}= -150; α=120^{o}; z_{L2}= - 4500

ξ **[degree]**	z_{R2} **[m]**	α **[degree]**	z_{L2} **[m]**
71	-2104	99	-
99	-3459	-	-6006

In the present SA algorithm, it is assumed that the gravitational gradient observations are due to a single anomaly. However, there are many different additional anomalies surrounding the fault whose parameters need to be estimated. Therefore, the gravitational gradient effects of different anomalies are also included in the observation profiles. In addition, the true values of the fault parameters are unknown, especially the z_{R1}, z_{L1} parameters. These parameters are used to construct the forward model, for which many simplifications were made. For example, z_{R2}, z_{L2} are chosen with the assumption that the real fault consists of the right and the left horizontal rectangular blocks. In fact, the real fault does not possess any defined mathematical shape. Therefore, no conclusion can be reached about the success of the SA algorithm on estimating these depth-related parameters. However, some comments on the dip angle estimation are in order since it can be inferred from Figure 4.30 that the dip angle is close to 90^{o}. From Table 4.20, it can be concluded that the SA algorithm is able to estimate the dip angle. The dip angle is not estimated correctly when estimated together with two depths parameters. On the other hand, the dip angle is estimated more accurately together with one depth parameter.

If the north-south direction survey tracks are used, then the observations are along the y-direction. In this case, the geometry of the observations relative to the fault are such that the fault intersects the u-axis at $-y_0 \sin(\pi-\alpha)$ and the coordinates of the observation points in the (u,v,w) system are $u=-y\sin(\pi-\alpha)$ as illustrated in Figure 4.35.

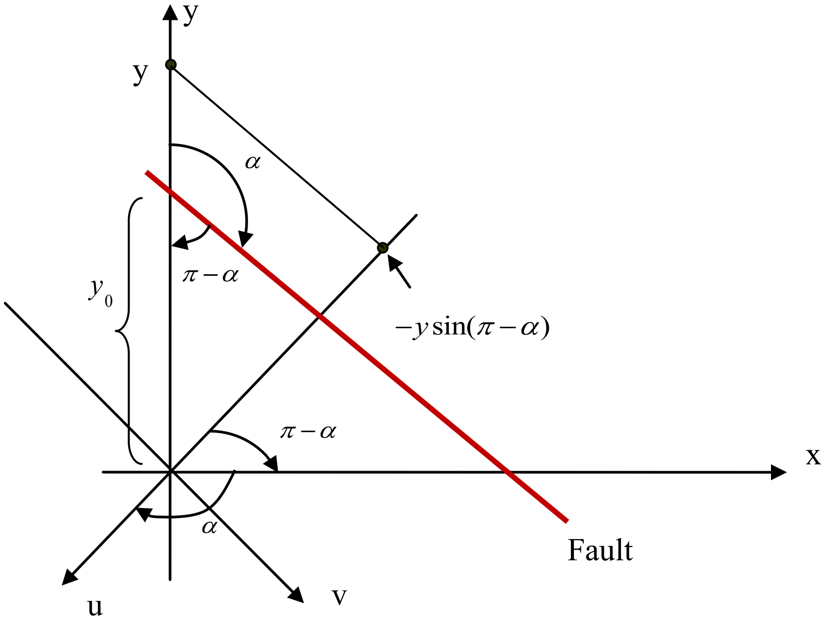


Figure 4.35: Geometry for an rotated fault crossing the y-axis

Replacing u_0 and u with $-y_0\sin(\pi-\alpha)$ and $-y\sin(\pi-\alpha)$ in the formulas (4.35), the gravitational gradient tensor is computed in the (u,v,w)-system,

$$\Gamma^{(u,v,w)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \Gamma_{22} & \Gamma_{23} \\ 0 & \Gamma_{32} & \Gamma_{33} \end{pmatrix}. \tag{4.39}$$

Then, these are rotated back to the (x,y,z)-system using the same rotation matrix as in eq. (4.38).

For the implementation of the SA method, the same approximate fault parameters are used as before, except that the location is given by $y_0 = 3836.22$ [m]. Therefore, also the same search domains are used. The Table 4.21 shows several cases of the estimated parameters of the fault.

Table 4.21: Estimated fault parameters using T31, T33 and T35 survey tracks of Γ_{33}. The approximate fault parameters; ξ=90^o ; z_{R2}= -150; α=120^o; z_{L2}=-4500

No. of survey tracks	ξ [degree]	z_{R2} [m]	α [degree]	z_{L2} [m]
T31, T33,T35	91	-4000	-	-3632
	93	-4000	144	-
T33	100	-1515	-	-9500
	100	-669	176	-

In this case, from Table 4.21, it is seen that the SA algorithm is successful in estimating the dip angle from the three survey tracks. The orientation angle is also reasonably well estimated from the three survey tracks of Γ_{33} gradients. It seems that the north-south survey profiles are most suitable for estimating this parameter. This is so because the gradients in the east-west direction have lower amplitudes. This can be explained by the fact that the geologic structure is elongated roughly in the east-west direction. The dip angle is not estimated well from one survey track of Γ_{33} gradient. In the following applications the north-south gradiometer survey tracks were used.

The other gradients are now considered to estimate the fault parameters. In this case, the cross gradients, Γ_{12} and Γ_{13} are not zero, but their magnitudes are not as significant as that of the gradients Γ_{22} and Γ_{23}. Therefore, Γ_{22} and Γ_{23} gradients are considered as observations to estimate the fault parameters. Table 4.22 presents the estimated parameters of the fault from the Γ_{22} and Γ_{23} gradients.

Table 4.22: Estimated fault parameters using T31, T33 and T35 survey tracks of Γ_{22} and Γ_{23} gradients. ξ=90^o ; z_{R2}= -150; α=120^o; z_{L2}= -4500

Gradient	ξ [degree]	z_{R2} [m]	α [degree]	z_{L2} [m]
Γ_{22}	81	-4000	141	-
	80	-4000		-3500
Γ_{23}	81	-188	170	-
	88	-4000	-	-5870

It can be concluded that the dip angle can be estimated reasonably well from both the Γ_{23} and the Γ_{22} gradients. Also, the orientation angle can be estimated from the Γ_{22} gradients.

Weights in the cost function were also considered when the Γ_{33} gradients constitute the observations.

$$\Phi_{jk}(\Xi) = \left({}^{i}\Gamma_{jk}^{obs} - {}^{i}\Gamma_{jk}^{pred}(\Xi)\right)^{T} P\left({}^{i}\Gamma_{jk}^{obs} - {}^{i}\Gamma_{jk}^{pred}(\Xi)\right) \tag{4.40}$$

where P is a $n \times n$ symmetric positive-definite weight matrix, usually defined by inverting the covariance matrix Σ, $P = \Sigma^{-1}$, where *j*=*k*=3 and *i*=1,…,*n*. For this purpose, the same covariance function as given in eq. (4.8) is used, but with different model parameters, α_j and σ_j^2. These parameters are adjusted values to the gravitational field for Oklahoma/Texas region as presented in Table 4.23.

Table 4.23: Covariance model parameters

$\sigma_1^2 = 2\times10^{-5}$	$\alpha_1 = 1\times10^{-3}$
$\sigma_2^2 = 1\times10^{-4}$	$\alpha_2 = 6\times10^{-4}$
$\sigma_3^2 = 7.5\times10^{-4}$	$\alpha_3 = 3\times10^{-4}$
$\sigma_4^2 = 1.42\times10^{-2}$	$\alpha_4 = 1.51\times10^{-4}$
$\sigma_5^2 = 7.5\times10^{-1}$	$\alpha_5 = 4.5\times10^{-5}$
$\sigma_6^2 = 36$	$\alpha_6 = 1.45\times10^{-5}$
$\sigma_7^2 = 778$	$\alpha_7 = 4.9\times10^{-6}$
$\sigma_8^2 = 3500$	$\alpha_8 = 7.7\times10^{-7}$
$\sigma_9^2 = 110000$	$\alpha_9 = 4.16\times10^{-7}$

The single track T33 is used for this application. The distance between consecutive points along this track is approximately 110 m. All points lie at the height of $z = z' = 1000$ m. The small interval of 110 m for the measurement points causes covariance matrix to be very ill-conditioned. Therefore, instead of using all 1775 points along the track, only 355 points were used, namely every 5th point along the track. So the point spacing along the survey track T33 is set as, $\Delta y = 440\,\text{m}$ and $\Delta x = 0\,\text{m}$. The covariance matrix, Σ, is used as defined in eq. (4.10). Table 4.24 presents the estimated fault parameters.

Table 4.24: Estimated fault parameters using T33survey track of Γ_{33} gradients using weights in the cost function. The approximate fault parameters; ξ=90°; z_{R2}= -150; α=120°; z_{L2}=-4500

ξ **[degree]**	z_{R2} **[m]**	α **[degree]**	z_{L2} **[m]**
68	4000	90	-
73	-3237	-	-3500

As can be seen by comparing Tables 4.21 and 4.24, using appropriate weights in the cost function improves the dip angle estimation.

Chapter 5: Conclusions

In this study, the Simulated Annealing (SA) algorithm is used to estimate various parameters of subsurface structures from airborne gravitational gradients. The method is tested for two different applications where parameters are estimated from observations of a simulated signal due to a rectangular prism buried in a real field, and from observations of a real signal due to a fault buried in a real field. In the first application, two cases were considered where one or more observation profiles are orthogonal to the anomaly and, then where the anomaly crosses these profiles at a certain azimuthal angle. The capability of the SA algorithm is tested in the presence of gravitational gradients due to a geologic background for both cases, which also include 6 different anomaly depths that determine the strength of the simulated signal. The depth, location, and orientation angle from vertical gravitational gradients, $\mathit{\Gamma}_{33}$, are estimated for these various cases. Furthermore, the least-squares solution (LESS) for the depth and location parameters of the prism are compared with the results from the SA algorithm. Moreover, the weights for the observational noise (geologic background and instrument noise) are taken into account in the cost function for the implementation of the SA algorithm. Finally gravitational cross gradients ($\Gamma_{12}, \Gamma_{13}, \Gamma_{23}$) are used to estimate the same parameters. In this case, special attention is given to the cases where the generated signal due to the prism is small relative to the geologic background signal.

It is shown that the SA algorithm is capable of estimating the depth, location and orientation angle of the subsurface prism from gravitational gradients for the 6 different depth cases. In case that the profiles are orthogonal to the anomaly, one observation profile of the vertical gravitational gradients is enough to estimate parameters according to the established criterion. It can be concluded that if the signal-to-noise ratio is close to 1, the SA algorithm clearly fails. The LESS within the Gauss-Helmert model is not uniformly successful since it diverges if the initial values for the parameters are not chosen appropriately. This is the case when the location parameter is estimated together with depth parameter. Since the cost function has local minima with respect to the location parameter, the performance of LESS depends very much on the initial values for this parameter. If it is not chosen within the global minimum well of the cost function, the solution does not converge to the correct solution. It is thus concluded that the simulated annealing algorithm is a stable technique, but that it also starts to break down when the signal-to-noise ratio is small. When using the weights for the observational noise in the cost function, it is shown that the results, especially the location estimation, turn out much better. In case that the subsurface anomaly crosses the survey tracks at a non-orthogonal angle, thus displacing the orthogonal projection of the prism centroid away from the crossing point, the SA algorithm is not able to estimate both the location of the centroid and the orientation angle from one observation profile of vertical gravitational gradients. Rather, at least two observation profiles are required to estimate these parameters of the prism.

When using gravitational cross gradients in the case that the anomaly crosses the survey tracks orthogonally, estimating the location of the prism from one observation profile of Γ_{23} gradients, compared to the vertical gravitational gradients, is much better. This is because the signal-to-noise ratio is higher for this particular case of the corresponding geologic background gradient. The other two gravitational cross gradients (Γ_{12}, Γ_{13}) are not suitable to detect the anomaly in this particular simulation since the signal-to-noise ratio is very small for these gradients. The choice of suitable gradient tensor component for estimation depends on how the coordinate system is established and how the anomaly is positioned in the field. If the anomaly (a long prism) is positioned in the field in such a way that its length is parallel to the east-west direction in an East-North-Up coordinate system, then there is not much change in the east-west direction and the gradients, Γ_{12}, Γ_{13} are close to zero.

In case that the anomaly crosses the survey tracks non-orthogonally, the parameters of the anomaly including the orientation can be estimated from three gravitational cross gradients ($\Gamma_{12}, \Gamma_{13}, \Gamma_{23}$) provided that two profiles are used since the location of the prism centroid cannot be estimated accurately from one observation profile. In comparison to the results obtained for the vertical gravitational gradients, especially the location estimation, is much better. It can be concluded that in the first case where the east-west anomaly is orthogonal to the north-south survey track, any of two gravitational gradients, Γ_{23}, Γ_{33}, and in the second case where the anomaly is oriented arbitrarily, any of four gravitational gradients Γ_{12}, Γ_{23}, Γ_{13}, Γ_{33} can be used to estimate the location, depth and orientation of the subsurface prism.

To test the SA algorithm on an actual anomaly, the parameters of a real fault in the Wichita uplift region of south-western Oklahoma are estimated from three observation profiles of airborne gravity gradients that were provided in both the north-south and the east-west directions. It is concluded that the dip angle is especially well estimated from the given gradients crossing the fault in the north-south direction. No conclusion could be reached on the depth parameters since the estimation problem particularly ill-posed for these parameters as formulated. Using appropriate weights in the cost function does not appear to improve the estimation of any of the parameters except for the dip angle.

In summary, the presented SA algorithm locates the anomaly if there is a known anomaly in a field, but with unknown parameters, i.e., location and depth. The algorithm is not able to detect an anomaly in the field if no information is available about it. It is shown that the SA algorithm estimates the parameters of a single anomaly according to the degree of noise level of the observations that are used. One of the strengths of the SA algorithm is that it is an inherently stable algorithm. Since it is based on a Monte Carlo method applied to a forward model, numerical instabilities due to inversion of ill-conditioned matrices are eliminated. Moreover, derivatives of highly nonlinear and complex functions need not be computed. That is, nonlinear models are inverted without linearization. In addition, the algorithm is designed to achieve a solution that corresponds to the global minimum of the cost function. As such it is superior to LESS which may yield a solution for a local minimum or simply diverge if the initial parameter values are not close to the true values. On the other hand, it is difficult to get any accuracy or precision information for the estimated results, in contrast to the LESS. The successful implementation of the SA algorithm also depends critically on the annealing schedule as well as initial parameters for the Markov chain. For example, choosing the initial temperature requires some experience with the problem to be solved. Since the algorithm searches the parameter space with some randomness, the results may be different for different runs of the SA algorithm (i.e., for different starting values, and for different random seed numbers) and a reasonable termination criterion must also be implemented.

For future research, the SA algorithm may be tested to estimate the parameters of two or more different anomalies. Further investigations need to be performed to obtain accuracy information of the estimated results from the SA.

References

Abt, T.L. (2011): Detection of a Local Mass Anomaly in the Shallow Subsurface by Applying a Matched Filter, Ph.D. dissertation, Div. of Geodetic Science, School of Earth Sciences, The Ohio State University, Columbus/OH, USA.

Barker, A.A. (1965): Monte Carlo calculations of the radial distribution functions for a proton-electron plasma, *Aust., J. Phys*., Vol.18, pp.119-133.

Bell Geospace (2008): Final report of processing and acquisition of Air-FTG data in Vinton Dome, Louisiana, Bell Geospace, Inc., Houston/TX.

Bell Geospace (2012): Bell Geospace Inc., Three GGIs and Bell Geospace's GGI. Retrieved 16 April 2012 from <www.bellgeospace.com/tech/technology_system.html>.

Baselga, S. (2007): Global optimization solution of robust estimation, *Journal of Surveying Engineering*, Vol.133, No.3, pp.123-128.

Baselga, S. and M. Asce (2011): Second-order design of geodetic networks by the simulated annealing, *Journal of Surveying Engineering*, Vol.137, No.4, pp.167-173.

Berne, J.L. and S. Baselga (2004): First-order design of geodetic networks using the simulated annealing method, *Journal of Geodesy*, 78:47-54.

Binder, K. (1988): Monte Carlo Simulation in Statistical Physics, Springer: Berlin/ Heidelberg /New York.

Bohachevsky, I.O, M.E. Johnson and M.L. Stein (1986): Generalized Simulated Annealing for Function Optimization, *Technometrics*, Vol. 28, No.3, pp.209-217.

Brooks, S.P. (1998): Markov Chain Monte Carlo Method and Its Application, *The Stastician*, Vol.47, No.1, pp. 69-100.

Brooks, S., A. Gelman, G.L. Jones, X. L. Meng (2011): Handbook of Markov Chain Monte Carlo, Chapman and Hall: Boca Raton/FL, USA.

Burington, R.S. (1948): Handbook of Mathematical Tables and Formulas, McGraw-Hill, Inc., fifth edition, New York.

Chib, S. and E. Greenberg (1995): Understanding the Metropolis-Hastings Algorithm, *The American Statistician*, Vol.49, No.4, pp. 327-335.

Chunduru, R.K., M.K. Sen and P.L.Stoffa (1996): 2-D resistivity inversion using spline parametrization and simulated annealing, *Geophysics*, Vol.61, No.1, pp. 151-161.

Coffman, J.D., M.C. Gilbert and D.A. McConnell (1986): An Interpretation of the Crustal Structure of the Southern Oklahoma Aulacogen Satisfying Gravity Data, Oklahoma Geological Survey Guide book 23, pp.1-10.

Corana, A., M. Marchesi, C. Martini and S. Ridella (1987): Minimizing Multimodal Functions of Continuous Variables with the "Simulated Annealing*", ACM Transactions on Mathematical Software* 13, 262-280.

Dittmer, J.K. and J.E. Szymanski (1995): The stochastic inversion of magnetics and resistivity data using the simulated annealing algorithm, *Geophysical Prospecting*, Vol. 43, pp. 397-416.

Eckhardt, R. (1987): Stan Ulam, John von Neumann and the Monte Carlo Method, Los Alamos Science, Special Issue 15, pp. 131-137.

ESA (1999): The Four Candidate Earth Explorer Core Missions, Gravity Field and Steady-State Ocean Circulation Mission, European Space Agency Publications Division, Noordwijk, The Netherlands.

Fosdick, L.D. (1963): Monte Carlo Computations on the Ising lattice, Methods in Computational Physics, Academic Press Inc., New York, Vol.1, pp. 245-280.

Gamerman, D. and H. F. Lopes (2006): Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference, second edition, Chapman &Hall: Boca Raton/FL.

Gelman, A., J.B. Carlin, H.S. Stern and D.B. Rubin (2004): Bayesian Data Analysis, 2nd edition, Chapman & Hall: Boca Raton/FL.

Geman, S. and D. Geman (1984): Stochastic Relaxation, Gibbs Distributions, and the Bayesian restoration of images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, No.6, pp.721-741.

Gilks, W. R., S. Richardson and D. J. Spiegelhalter (1996): Introducing Markov chain Monte Carlo. Markov Chain Monte Carlo in Practice, Chapman and Hall: London, pp. 1-19.

Goffe, W., G. D. Ferrier and J. Rogers (1994): Global optimization of statistical functions with simulated annealing, *Journal of Econometrics,* Vol. 60, pp. 65-99.

Green P.J. (1995): Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination, *Biometrika*, Vol. 82, No.4, pp. 711-732.

Ham W. E., R. E. Denison and C. A. Merritt (1964): Basement Rocks and Structural Evolution of Southern Oklahoma, plate 1, Oklahoma Geological Survey, Bulletin No. 95, The University of Oklahoma, Norman.

Hammersly, J. M., and D.C. Handscomb (1964): Monte Carlo Methods, Methuen & Co., London, and Jon Wiley & Sons Inc., New York, pp. 113-126.

Hastings, W. K. (1970): Monte Carlo Sampling Methods Using Markov Chains and Their Applications, *Biometrika*, Vol. 57, No.1, pp.97-109.

Isaacson, D. L., and R. W. Madsen (1985): Markov Chains Theory and Applications, Krieger Publishing Company: Malabar/Florida, USA.

Jekeli, C. (1988): Gravity Gradiometer Survey System, *EOS*, vol.69, No.8, pp.105-117.

Jekeli, C. (1993): A review of gravity gradiometer survey system data analyses, *Geophysics*, Vol.58, No.4, pp. 508-514.

Jekeli, C. (2001): Inertial Navigation Systems with Geodetic Applications, de Gruyter: Berlin/New York, 2001.

Jekeli, C. (2003): Statistical Analysis of Moving-Base Gravimetry and Gravity Gradiometry, OSU Report No.466, Department of Civil and Environmental Engineering and Geodetic Science, The Ohio State University, Columbus, OH, USA.

Jekeli, C. (2006): Airborne Gradiometry Error Analysis, *Surveys in Geophysics*, 27, pp.257–275.

Jekeli, C. and T. L. Abt (2010): The Statistical Performance of the Matched Filter for Anomaly Detection Using Gravity Gradients, OSU Report No. 494, Div. of Geodetic Science, School of Earth Sciences, The Ohio State University, Columbus, OH, USA.

Kirkpatrick, S., C. D. Gelatt, Jr., and M.P. Vecchi (1983): Optimization by simulated Annealing, *Science*, Vol. 220, No. 4598, pp. 671-680.

Laarhoven, P.J.M. van and E.H.L. Aarts (1986): Simulated Annealing: Theory and Applications, Mathematics and Its Applications, Reidel Publishing Company: Dordrecht, The Netherlands.

Liu, J. S. (2002): Monte Carlo Strategies in Scientific Computing, Springer: Department of Statistics, Harvard University.

Metropolis, N., A.W. Rosenbluth, M.N. Rosenbluth and A. H. Teller (1953): Equation of State Calculations by Fast Computing Machines, *The Journal of Chemical Physics*, Vol. 21, No.6, pp. 1087-1092.

Metropolis, N. and S. Ulam (1949): The Monte Carlo Method, *Journal of the American Statistical Association*, Vol. 44, No. 247, pp. 335-341.

Metropolis, N. (1987): The beginning of the Monte Carlo Method, Los Alamos Science Special Issue 15, pp.125-30.

Montesinos, F. G., J. Arnoso and R. Vieira (2005): Using a genetic algorithm for 3-D inversion of gravity data in Fuerteventura (Canary Islands), *Int. J. Earth Sci*., 94: 301-316.

Moritz, H. (2008): Advanced Physical Geodesy (Reprint), Division of Geodesy and Geospatial Science, School of Earth Sciences, Ohio State University, Columbus, Ohio, USA.

Mosegaard, K. and P.D. Vestergaard (1991): A Simulated Annealing approach to seismic model optimization with sparse prior information, *Geophysical Prospecting*, 39, pp.599-611.

Mosegaard, K. and M. Sambridge (2002): Monte Carlo methods in geophysical inverse problems, *Reviews of Geophysics*, 40 (3), pp. 1-29.

Murphy, C. A. (2010): Recent developments with Air-FTG, Airborne Gravity 2004-Abstracts from the ASEG-PESA Airborne Gravity 2010 workshop.

Nagihara, S. and S. Hall (2001): Three-dimensional gravity inversion using simulated annealing: Constraints on the diapiric roots of allochthonous salt structures, *Geophysics*, Vol. 66, No.5, pp. 1483-1449.

Peskun, P.H. (1973): Optimum Monte-Carlo Sampling Using Markov Chains, *Biometrika*, Vol. 60, no.3, pp.607-612.

Pope, J. A. (1972): Some Pitfalls to be avoided in the iterative adjustment of nonlinear problem, in: Proceedings of the 38th Annual Meeting of the ASPRS, Falls Church, VA, pp. 449-447.

Roberts, G.O. (1996): Markov Chain Concepts Related to Sampling Algorithms. Markov chain Monte Carlo in Practice, Chapman and Hall: London.

Robert, C.P. and G. Casella (2004): Monte Carlo Statistical Methods, 2nd edition, Springer: New York.

Robbins, S.L. and G.R. Keller (1992): Complete Bouguer and Isostatic Residual Gravity Maps of the Anadarko Basin, Wichita Mountains, and Surrounding Areas, Oklahoma, Kansas, Texas, and Colorado, U.S. Geological Survey Bulletin 1866G, pp. G1-G11.

Rothman, D.H. (1985): Nonlinear inversion, statistical mechanics, and residual statics estimation, *Geophysics*, Vol.50, No.12, pp. 2784-2796.

Roy, K.K. (2008): *Potential Theory in Applied Geophysics*, Springer: Berlin/ Heidelberg.

Roy, L., M.K. Sen, D.D. Blankenship, P.L. Stoffa and T.G. Richter (2005): Inversion and uncertainty estimation of gravity data using simulated annealing: An application over Lake Vostok, East Antarctica, *Geophysics*, Vol.70, No.1, pp.J1-J12.

Ryden, N. and C.B. Park (2006): Fast Simulated Annealing inversion of surface waves on pavement using phase-velocity spectra, *Geophysics*, Vol.71, No.4, pp. R49-R58.b

Sambridge, M. and K. Mosegaard (2002): Monte Carlo Methods in Geophysical Inverse Problems, *Reviews of Geophysics*, Vol.40, No. 3, pp.1-29.

Schaffrin, B. and K. Snow (2010): Total Least-Squares regularization of Tykhonov type and an ancient racetrack in Corinth, *Linear Algebra and its Applications*, Vol. 432, pp. 2061-2076.

Sen, M.K. and P.L. Stoffa (1991): Nonlinear one-dimensional seismic waveform inversion using simulated annealing, *Geophysics*, Vol. 56, No. 10, pp.1624-1638.

Sen, M.K. and P.L. Stoffa (1995): *Global Optimization Methods in Geophysical Inversion*, Elsevier: Amsterdam, The Netherlands.

Silva, J.B.C., W.E. Medeiros and V.C.F. Barbosa (2001): Potential-field inversion: Choosing the appropriate technique to solve a geologic problem, *Geophysics*, Vol.66, No.2, pp. 511-520.

Smith, W. E., R.G. Paxman and H.H. Barret (1985): Image reconstruction from coded data: I. Reconstruction algorithms and experimental results, *J. Opt. Soc. Am.*, Vol. 2, No.4, pp. 491-500.

Snow, K. (2012): Topics in Total Least-Squares Adjustment within the Errors-In-Variables Model: Singular Cofactor Matrices and Prior Information, OSU Report No.502, Div. of Geodetic Science, School of Earth Sciences, The Ohio State University, Columbus, OH, USA.

Stewart, W.J. (2009): Probability, Markov Chains, Queues and Simulation, Princeton University Press: Princeton and Oxford.

Szabó, Z. (1998): Three fundamental papers of Roland Eötvös, Eötvös Roland Geophysical Institute of Hungary, Budapest, 1998.

Tarantola, A. (2005): *Inverse Problem Theory and Methods for Model Parameter Estimation*, Society for Industrial and Applied Mathematics: Philadelphia/PA, USA.

Telford, W.M., L.P. Geldart and R.E. Sheriff (1990): Applied Geophysics, 2nd edition, Cambridge University Press: Cambridge/New York.

Tierney, L. (1994): Markov Chains for Exploring Posterior Distribution, *The Annals of Statistics*, Vol.22, No.4, pp. 1701-1762.

Tierney, L. (1996): Introduction to general state-space Markov chain theory. Markov Chain Monte Carlo in Practice, Chapman and Hall: London.

Tierney, L. (1998): A note on Metropolis-Hastings Kernels for General State Spaces, *The Annals of Applied Probability*, Vol. 8, No.1, pp. 1-9.

Trivedi, K.S. (1982): Probability and Statistics with Reliability, Queuing, and Computer Science Applications, Prentice-Hall: New York.

Wood, W.W. (1968): Monte Carlo Studies of Simple Liquid Models. Physics of Simple Liquids, North-Holland Publ. Co: Amsterdam, pp.114-230.

Wood, W. W. and Parker (1957): Monte Carlo Equation of State of Molecules Interacting with the Lennard Jones Potential. I. A. Supercritical Isotherm at about Twice the Critical Temperature, *J. Chem. Phys*. Vol.27, pp. 720-733.

Yin, C. and G. Hodges (2007): Simulated annealing for airborne EM inversion, *Geophysics*, Vol. 72, No.4, pp. F189-F195.

Zhu, L. and C. Jekeli (2009): Gravity gradient modeling using gravity and DEM, *J. of Geodesy*, 83, pp. 557-567.

Appendix A: Gravitational Gradients due to infinite horizontal dip-slip fault

The logarithmic potential of an infinitely extended 3-D body with constant cross-section in the *u*-*w* plane is given by (Telford et al., 1990, pp.8)

$$V(u,w) = 2G\rho \int_{u'}\int_{w'} \ln\left(\frac{1}{r}\right) du'dw' \tag{A-1}$$

where $r = \sqrt{(u-u')^2 + (w-w')^2}$, *u* and *w* are the coordinates of the computation points defined in an East-North-Up coordinate system and u' and w' are the variables of the integration over the elements of the source body. The computation points are on or above the earth surface, $w - w' > 0$. The gravitational acceleration, is the first derivative of the gravitational potential with respect to *u* and *w*,

$$\frac{\partial V}{\partial u} = 2G\rho \frac{\partial V}{\partial u}\left(\ln\frac{1}{r}\right) = 2G\rho \cdot \frac{-2(u-u')}{2r^3} \cdot r = -2G\rho \frac{(u-u')}{r^2} \tag{A-2}$$

$$\frac{\partial V}{\partial w} = 2G\rho \frac{\partial V}{\partial w}\left(\ln\frac{1}{r}\right) = 2G\rho \cdot \frac{-2(w-w')}{2r^3} \cdot r = -2G\rho \frac{(w-w')}{r^2} \tag{A-3}$$

and it is expressed in vector form as

$$\underline{g}(u,w) = \begin{bmatrix} \partial V / \partial u \\ 0 \\ \partial V / \partial w \end{bmatrix} = \begin{bmatrix} -2G\rho \int_{u'}\int_{w'} \frac{u-u'}{r^2} du'dw' \\ 0 \\ -2G\rho \int_{u'}\int_{w'} \frac{w-w'}{r^2} du'dw' \end{bmatrix} \tag{A-4}$$

Gravitational gradients are then expressed by

$$\Gamma(u,w)=\begin{bmatrix}\partial^2V/\partial u^2 & 0 & \partial^2V/\partial u\partial w\\ 0 & 0 & 0\\ \partial^2V/\partial w\partial u & 0 & \partial^2V/\partial^2 w\end{bmatrix}=\begin{bmatrix}\Gamma_{11} & 0 & \Gamma_{13}\\ 0 & 0 & 0\\ \Gamma_{31} & 0 & \Gamma_{33}\end{bmatrix} \tag{A-5}$$

where its elements are given by

$$\begin{aligned}\frac{\partial V}{\partial u}\left(\frac{\partial V}{\partial u}\right)&=-2G\rho\frac{\partial V}{\partial u}\left(\frac{u-u'}{r^2}\right)=-2G\rho\left(\frac{(u-u')^2+(w-w')^2-2(u-u')(w-w')}{r^4}\right)\\ &=-2G\rho\left(\frac{-(u-u')^2+(w-w')^2}{r^4}\right)\end{aligned}$$

$$\frac{\partial V}{\partial u}\left(\frac{\partial V}{\partial w}\right)=-2G\rho\frac{\partial V}{\partial u}\left(\frac{w-w'}{r^2}\right)=-2G\rho\left(\frac{-2(u-u')(w-w')}{r^4}\right)$$

$$\begin{aligned}\frac{\partial V}{\partial w}\left(\frac{\partial V}{\partial w}\right)&=-2G\rho\frac{\partial V}{\partial w}\left(\frac{w-w'}{r^2}\right)=-2G\rho\left(\frac{(u-u')^2+(w-w')^2-2(u-u')(w-w')}{r^4}\right)=\\ &=-2G\rho\left(\frac{(u-u')^2-(w-w')^2}{r^4}\right)\end{aligned}$$

Therefore, the gravitational gradient matrix can be written as

$$\Gamma(u,w)=\begin{bmatrix}2G\rho\int\limits_{u'}\int\limits_{w'}\frac{(u-w')^2-(w-w')^2}{r^4}du'dw' & 0 & 4G\rho\int\limits_{u'}\int\limits_{w'}\frac{(u-u')(w-w')}{r^4}du'dw'\\ 0 & 0 & 0\\ 4G\rho\int\limits_{u'}\int\limits_{w'}\frac{(u-u')(w-w')}{r^4}du'dw' & 0 & 2G\rho\int\limits_{u'}\int\limits_{w'}\frac{(w-w')^2-(u-u')^2}{r^4}du'dw'\end{bmatrix} \tag{A-6}$$

Consider a semi-infinite horizontal slab with finite thickness that ends on an inclined fault plane as illustrated in Figure A.1.

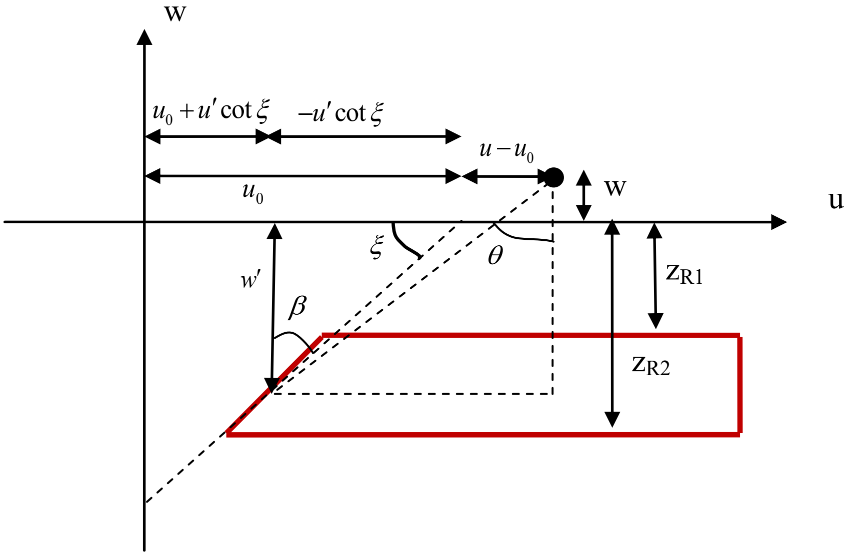


Figure A.1: Definition of right horizontal slab

To find the gravitational gradients, from eq. (A-4), the following equation needs to be integrated

$$\frac{\partial V}{\partial w} = -2G\rho \int_{u'}\int_{w'} \frac{u-u'}{r^2} du'dw' \tag{A-7}$$

with respect to w' as a function of u'. The integration limit for u' starts at $u_0 + w' \cot\xi$ and ends in infinity, ∞. The limits of the integration, $w' = z_{R1}$, $w' = z_{R2}$, are the right lower and upper planes of the slab, respectively. To integration with respect to u',

$$\int_{u_0 + w'\cot\xi}^{\infty} \frac{1}{r^2} du' \tag{A-8}$$

obtained by utilizing the Table of integrals (Burington, 1948, pp. 84)

$$\int \frac{1}{ax^2 + bx + c} dx' = \frac{2}{\sqrt{4ac - b^2}} \tan^{-1} \frac{2ax + b}{\sqrt{4ac - b^2}}, \tag{A-9}$$

Equation (A-8) can be expressed as in eq. (A-9), as follows,

$$\int_{u_0 + w'\cot\xi}^{\infty} \frac{1}{(u-u')^2 + (w-w')^2} du' = \int_{u_0 + w'\cot\xi}^{\infty} \frac{1}{u^2 - 2uu' + u'^2 + (w-w')^2} du'$$

$$= \frac{2}{\sqrt{4(u^2+(w-w')^2)-4u^2}} \tan^{-1} \frac{2u'-2u}{\sqrt{4(u^2+(w-w')^2)-4u^2}} \Bigg|_{u_0+w'\cot\xi}^{\infty}$$

$$= \frac{2}{2(z-z')} \tan^{-1} \frac{2u'-2u}{2(w-w')} \Bigg|_{u_0+w'\cot\xi}^{\infty} = \frac{1}{(w-w')}\frac{\pi}{2} - \frac{1}{(w-w')} \tan^{-1} \frac{u_0+w'\cot\xi-u}{(w-w')}$$

Therefore, the integration in eq. (A-8) is expressed as

$$\int_{u_0+w'\cot\xi}^{\infty} \frac{1}{r^2} du' = \frac{1}{w-w'} \left(\frac{\pi}{2} + \tan^{-1} \frac{u-u_0-w'\cot\xi}{w-w'} \right) \tag{A-10}$$

Then, the integration in eq. (A-7) becomes

$$\begin{aligned} \frac{\partial V}{\partial w} &= -2G\rho \int_{z_{R1}}^{z_{R2}} \left(\frac{\pi}{2} + \tan^{-1} \frac{u-u_0-w'\cot\xi}{w-w'} \right) dw' \\ &= -2G\rho \left(\frac{\pi}{2}(z_{R2}-z_{R1}) + \int_{z_{R1}}^{z_{R2}} \tan^{-1} \frac{u-u_0-w'\cot\xi}{w-w'} dw' \right) \end{aligned} \tag{A-11}$$

To compute the integration term in eq. (A-11), we need to find an expression for dw'. From Figure A.1, the following relationship is defined for θ,

$$\tan\theta = \frac{u-u_0-w'\cot\xi}{w-w'} \tag{A-12}$$

If w' is left alone in the left hand side of eq. (A-12), we have

$$w' = \frac{w\tan\theta-u-u_0}{\tan\theta-\cot\xi} = \frac{w\tan\theta-u-u_0}{\tan\theta-\cot(\pi/2-\beta)} = \frac{w\tan\theta-u-u_0}{\tan\theta-\tan\beta} \tag{A-13}$$

The derivative of eq. (A-13) is computed with respect to θ

$$dw' = \frac{\dfrac{w(\tan\theta-\tan\beta)}{\cos^2\theta} - \dfrac{(w\tan\theta-u-u_0)}{\cos^2\theta}}{(\tan\theta-\tan\beta)^2} d\theta = \frac{-w\tan\beta+u-u_0}{\cos^2\theta(\tan\theta-\tan\beta)^2} d\theta \tag{A-14}$$

The denominator of eq. (A-14) can be rewritten as

$$\begin{aligned}\cos^2\theta(\tan\theta-\tan\beta)^2 &= \cos^2\theta(\tan^2\theta-2\tan\theta\tan\beta+\tan^2\beta)\\ &=\cos^2\theta\left(\frac{\sin^2\theta}{\cos^2\theta}-2\frac{\sin\theta}{\cos\theta}\frac{\sin\beta}{\cos\beta}+\frac{\sin^2\beta}{\cos^2\beta}\right)=\\ &=\sin^2\theta-2\frac{\sin\theta\sin\beta\cos\theta}{\cos\beta}+\cos^2\theta\frac{\sin^2\beta}{\cos^2\beta}\\ &=\frac{\sin^2\theta\cos^2\beta-2\sin\theta\sin\beta\cos\theta\cos\beta+\cos^2\theta\sin^2\beta}{\cos^2\beta}\\ &=\frac{(\sin\theta\cos\beta-\cos\theta\sin\beta)^2}{\cos^2\beta}=\frac{\sin^2(\theta-\beta)}{\cos^2\beta}\end{aligned} \tag{A-15}$$

If the term in eq. (A-15) is inserted into the denominator of eq. (A-14), then

$$dw'=\frac{(-w\dfrac{\sin\beta}{\cos\beta}+u-u_0)\cos^2\beta}{\sin^2(\theta-\beta)}d\theta=\frac{-w\sin\beta\cos\beta+(u-u_0)\cos^2\beta}{\sin^2(\theta-\beta)}d\theta \tag{A-16}$$

From eq. (A-12), the angle θ is obtained as

$$\theta=\arctan\frac{u-u_0-w'\cot\xi}{w-w'} \tag{A-17}$$

and from Figure A.1, we can see that the angle, β is

$$\beta=\pi/2-\xi\,. \tag{A-18}$$

Defining another angle, ψ as

$$\psi=\theta-\beta\ , \tag{A-19}$$

the equation (A-16) is expressed as

$$dw'=\frac{((u-u_0)\cos\beta-w\sin\beta)\cos\beta}{\sin^2\psi}d\psi \tag{A-20}$$

Then, the integral in eq. (A-11) becomes

$$\int_{z_{R1}}^{z_{R2}} \tan^{-1}\frac{u-u_0-w'\cot\xi}{w-w'}dw' = \int_{z_{R1}}^{z_{R2}} \theta dw' = \int_{\psi_1}^{\psi_2} (\psi+\beta)\frac{((u-u_0)\cos\beta - w\sin\beta)\cos\beta}{\sin^2\psi}d\psi$$

$$= ((u-u_0)\cos\beta - w\sin\beta)\cos\beta\int_{\psi_1}^{\psi_2}\frac{(\psi+\beta)}{\sin^2\psi}d\psi \qquad \text{(A-21)}$$

The limit of the integration in eq. (A-17) is then defined by

$$\psi_1 = \psi_2 = \theta_{1,2} - \beta = \tan^{-1}\frac{u-u_0-w'\cot\xi}{w-w'} - \left(\frac{\pi}{2} - \xi\right) \qquad \text{(A-22)}$$

Using the following information,

$$\int\frac{1}{\sin^2\psi}d\psi = -\cot\psi \quad \text{since } \frac{\partial}{\partial\psi}\cot\psi = -\csc^2(\psi) = -\frac{1}{\sin^2\psi}$$

The integration in eq. (A-17) is computed as integration by part (Burington, 1948, pp. 61)

$$\int u dv = uv - \int v du$$

where $u = (\psi+\beta)$, $v = -\cot\psi$, $du = d\psi$, $dv = 1/\sin^2\psi$. Therefore,

$$\int_{\psi_1}^{\psi_2}\frac{\psi+\beta}{\sin^2\psi}d\psi = -(\psi+\beta)\cot\psi + \int_{\psi_1}^{\psi_2}\cot\psi d\psi$$

$$= -(\psi+\beta)\cot\psi + \ln(\sin\psi)\Big|_{\psi_1}^{\psi_2} =$$

$$= -(\psi_2+\beta)\cot\psi_2 + \ln(\sin\psi_2) + (\psi_1+\beta)\cot\psi_1 - \ln(\sin\psi_1)$$

$$= -(\psi_2+\beta)\cot\psi_2 + (\psi_1+\beta)\cot\psi_1 + \ln\left(\frac{\sin\psi_2}{\sin\psi_1}\right) \qquad \text{(A-23)}$$

By inserting eq. (A-23) into the integral in eq. (A-21) leads to

$$\int_{z_{R1}}^{z_{R2}} \tan^{-1}\frac{u-u_0-w'\cot\xi}{w-w'}dw' = ((u-u_0)\cos\beta - w\sin\beta)\cos\beta\int_{\psi_1}^{\psi_2}\frac{(\psi+\beta)}{\sin^2\psi}d\psi$$

$$= -((u-u_0)\cos\beta - w\sin\beta)\cos\beta\left((\psi_2+\beta)\cot\psi_2 + (\psi_1+\beta)\cot\psi_1 + \ln\left(\frac{\sin\psi_2}{\sin\psi 1}\right)\right)$$

(A-24)

Therefore,

$$\frac{\partial V}{\partial w}=-2G\rho\left(\begin{array}{l}\frac{\pi}{2}(z_{R2}-z_{R1})+((u-u_0)\cos\beta-w\sin\beta)\cos\beta\cdot\\ \qquad\left(-(\psi_2+\beta)\cot\psi_2+(\psi_1+\beta)\cot\psi_1+\ln(\frac{\sin\psi_2}{\sin\psi_1})\right)\end{array}\right)$$

since $\theta=(\psi+\beta)$. Then,

$$\frac{\partial V}{\partial w}=-2G\rho\left(\begin{array}{l}\frac{\pi}{2}(z_{R2}-z_{R1})-((u-u_0)\cos\beta-w\sin\beta)\cos\beta\cdot\\ \qquad\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln(\frac{\sin\psi_2}{\sin\psi_1})\right)\end{array}\right) \tag{A-25}$$

To find the vertical gravitational gradient $\Gamma_{33}^{(R)}$, the derivative of eq. (A-25) is computed with respect to *w*. Then,

$$\frac{\partial^2 V}{\partial w^2}=-2G\rho\left(\begin{array}{l}\sin\beta\cos\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln(\frac{\sin\psi_2}{\sin\psi_1})\right)-((u-u_0)\cos\beta-w\sin\beta)\cos\beta\cdot\\ \left(\frac{\partial\theta_2}{\partial w}\cot\psi_2-\theta_2\csc^2\psi_{1,2}\frac{\partial\theta_2}{\partial w}-(\frac{\partial\theta_1}{\partial w}\cot\psi_1+\theta_1\csc^2\psi_1\frac{\partial\theta_1}{\partial w})\right.\\ \left.-\cot\psi_2\frac{\partial\theta_2}{\partial w}+\cot\psi_1\frac{\partial\theta_1}{\partial w}\right)\end{array}\right)$$

$$\Gamma_{33}^{(R)}=-2G\rho\left(\begin{array}{l}\sin\beta\cos\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln(\frac{\sin\psi_2}{\sin\psi_1})\right)-\\ ((u-u_0)\cos\beta-w\sin\beta)\cos\beta\left(-\theta_2\csc^2\psi_{1,2}\frac{\partial\theta_2}{\partial w}+\theta_1\csc^2\psi_1\frac{\partial\theta_1}{\partial w}\right)\end{array}\right) \tag{A-26}$$

where the terms $\frac{\partial\theta_2}{\partial w}$, $\frac{\partial\theta_1}{\partial w}$ are computed by taking the derivatives of eq. (A-17)

$$\frac{\partial\theta_{1,2}}{\partial w}=-\frac{(u-u_0-w'\cot\xi)}{(w-z_{R1,2})^2}\cdot\frac{1}{1+\left(\dfrac{u-u_0-z_{R1,2}\cot\xi}{w-z_{R1,2}}\right)^2}=-\frac{(u-u_0-z_{R1,2}\cot\xi)}{(w-z_{R1,2})^2+(u-u_0-z_{R1,2}\cot\xi)^2}$$

(A-27)

From Figure A.1, the following relationships are established

$$\sin\theta=\frac{u-u_0-z_{R1,2}\cot\xi}{\sqrt{(w-z_{R1,2})^2+(u-u_0-z_{R1,2}\cot\xi)^2}} \tag{A-28}$$

$$\cos\theta=\frac{(w-z_{R1,2})}{\sqrt{(w-z_{R1,2})^2+(u-u_0-z_{R1,2}\cot\xi)^2}} \tag{A-29}$$

Therefore, from eq. (A-28) and eq. (A-29), eq. (A-27) is further rearranged as

$$\frac{\partial\theta_{1,2}}{\partial w}=-\frac{(u-u_0-z_{R1,2}\cot\xi)}{(w-z_{R1,2})^2+(u-u_0-z_{R1,2}\cot\xi)^2}=-\frac{\sin\theta\cos\theta}{w-z_{R1,2}}\ . \tag{A-30}$$

Similarly, $\Gamma_{13}^{(R)}$ is obtained taking derivative of eq. (A-25) with respect to *u* as follows;

$$\frac{\partial}{\partial u}\left(\frac{\partial V}{\partial w}\right)=-2G\rho\left(\begin{array}{l}-\cos^2\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln(\dfrac{\sin\psi_2}{\sin\psi_1})\right)-((u-u_0)\cos\beta-w\sin\beta)\cos\beta\cdot\\ \left(\dfrac{\partial\theta_2}{\partial u}\cot\psi_2-\theta_2\csc^2\psi_{1,2}\dfrac{\partial\theta_2}{\partial u}-\dfrac{\partial\theta_1}{\partial u}\cot\psi_1+\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial u}\right.\\ \left.-\cot\psi_2\dfrac{\partial\theta_2}{\partial u}+\cot\psi_1\dfrac{\partial\theta_1}{\partial u}\right)\end{array}\right)$$

$$\Gamma_{13}^{(R)}=2G\rho\left(\begin{array}{l}\cos^2\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln(\dfrac{\sin\psi_2}{\sin\psi_1})\right)+\\ ((u-u_0)\cos\beta-w\sin\beta)\cos\beta\cdot\left(-\theta_2\csc^2\psi_{1,2}\dfrac{\partial\theta_2}{\partial u}+\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial u}\right)\end{array}\right) \tag{A-31}$$

where the terms $\dfrac{\partial\theta_2}{\partial u},\dfrac{\partial\theta_1}{\partial u}$ are given by

$$\frac{\partial \theta_{1,2}}{\partial u} = \frac{(w - z_{R1,2})}{(w - z_{R1,2})^2} \cdot \frac{1}{1 + \left(\frac{u - u_0 - z_{R1,2} \cot \xi}{w - z_{R1,2}} \right)^2} = \frac{1}{(w - z_{R1,2})} \frac{(u - u_0 - z_{R1,2} \cot \xi)}{(w - z_{R1,2})^2 + (u - u_0 - z_{R1,2} \cot \xi)^2} \tag{A-32}$$

which can be further simplified using eq. (A-29) as

$$\frac{\partial \theta_{1,2}}{\partial u} = \frac{1}{(w - z_{R1,2})} \frac{(u - u_0 - z_{R1,2} \cot \xi)}{(w - z_{R1,2})^2 + (u - u_0 - z_{R1,2} \cot \xi)^2} = \frac{\cos^2 \theta}{(w - z_{R1,2})} \tag{A-33}$$

As can be seen from eq. (A-6), the diagonal elements of the gravitational gradient matrix satisfy Laplace's equation. Thus, the Γ_{11} gradients are

$$\Gamma_{11}^{(R)} = -\Gamma_{33}^{(R)} \tag{A-34}$$

Similar derivations can be computed for the left horizontal slab. Figure A.2 shows the geometry of the left horizontal slab.

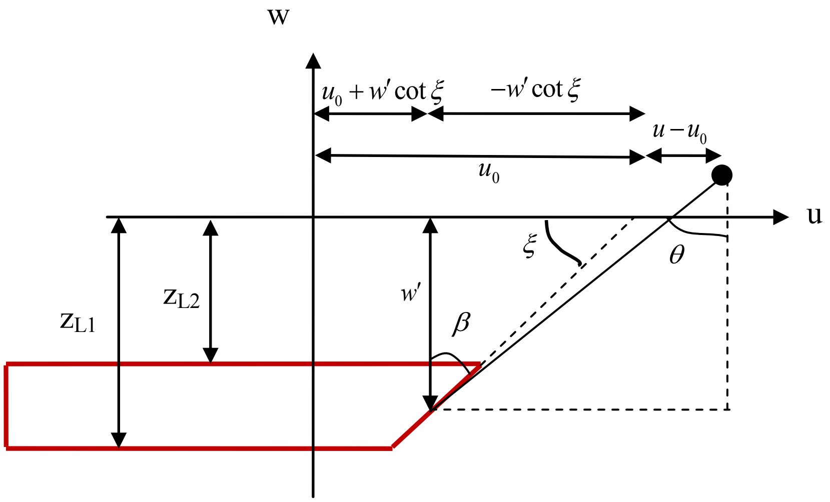


Figure A.2: Definition of the left horizontal slab

In this case, the integration limit for u' starts at $-\infty$ and ends at $u_0 + w' \cot \xi$; and the limits of the integration, $w' = z_{L1}, w' = z_{L2}$ are the lower and upper left planes of the slab, respectively. The integration in eq. (A-10) becomes

$$\int_{-\infty}^{u_0+w'\cot\xi} \frac{1}{r^2}du' = \frac{2}{\sqrt{4(u^2+(w-w')^2)-4u^2}}\tan^{-1}\frac{2u'-2u}{\sqrt{4(u^2+(w-w')^2)-4x^2}}\Bigg|_{-\infty}^{u_0+w'\cot\xi}$$

$$=\frac{2}{2(w-w')}\tan^{-1}\frac{2u'-2u}{2(w-w')}\Bigg|_{-\infty}^{u_0+w'\cot\xi} = \frac{1}{(w-w')}\tan^{-1}\frac{u_0+w'\cot\xi-u}{(w-w')}-\left(-\frac{1}{(w-w')}\frac{\pi}{2}\right)$$

Therefore,

$$\begin{aligned}\frac{\partial V}{\partial w} &= -2G\rho\int_{z_{L1}}^{z_{L2}}\left(\frac{\pi}{2}-\tan^{-1}\frac{u-u_0-w'\cot\xi}{w-w'}\right)dw' \\ &= -2G\rho\left(\frac{\pi}{2}(z_{L2}-z_{L1})-\int_{z_{L1}}^{z_{L2}}\tan^{-1}\frac{u-u_0-w'\cot\xi}{w-w'}dw'\right)\end{aligned} \tag{A-35}$$

Similarly, the integral term in eq. (A-35) can be rewritten as from eq. (A-21)

$$\begin{aligned}\int_{z_{L1}}^{z_{L2}}\tan^{-1}\frac{u-u_0-w'\cot\xi}{w-w'}dw' &= \int_{z_{L1}}^{z_{L2}}\theta dw' = \int_{\psi_1}^{\psi_2}(\psi+\beta)\frac{((u-u_0)\cos\beta-w\sin\beta)\cos\beta}{\sin^2\psi}d\psi \\ &= ((u-u_0)\cos\beta-w\sin\beta)\cos\beta\int_{\psi_1}^{\psi_2}\frac{(\psi+\beta)}{\sin^2\psi}d\psi \\ &= -((u-u_0)\cos\beta-w\sin\beta)\cos\beta\left((\psi_2+\beta)\cot\psi_2+(\psi_1+\beta)\cot\psi_1+\ln\left(\frac{\sin\psi_2}{\sin\psi 1}\right)\right)\end{aligned}$$

Inserting the above integral into the eq. (A-35) leads to

$$\frac{\partial V}{\partial w} = -2G\rho\left(\frac{\pi}{2}(z_{L2}-z_{L1})+((u-u_0)\cos\beta-w\sin\beta)\cos\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln(\frac{\sin\psi_2}{\sin\psi_1})\right)\right) \tag{A-36}$$

Similarly if the derivative of eq. (A-27) is computed with respect to *w*, then the vertical gravitational gradients are obtained as

$$\Gamma_{33}^{(L)} = -2G\rho\left(\begin{array}{l}-\sin\beta\cos\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln(\dfrac{\sin\psi_2}{\sin\psi_1})\right)+ \\ ((u-u_0)\cos\beta-w\sin\beta)\cos\beta\cdot\left(-\theta_2\csc^2\psi_{1,2}\dfrac{\partial\theta_2}{\partial w}+\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial w}\right)\end{array}\right) \tag{A-37}$$

where the terms $\frac{\partial\theta_2}{\partial w}$, $\frac{\partial\theta_1}{\partial w}$ are computed as

$$\frac{\partial\theta_{1,2}}{\partial w} = -\frac{(u-u_0-z_{L1,2}\cot\xi)}{(w-z_{L1,2})^2+(u-u_0-z_{L1,2}\cot\xi)^2} = -\frac{\sin\theta\cos\theta}{w-z_{L1,2}} \ , \tag{A-38}$$

and $\Gamma_{13}^{(L)}$ is the derivative of eq. (A-36) with respect to *u*:

$$\Gamma_{13}^{(L)} = -2G\rho\left(\begin{array}{l} \cos^2\beta\left(\theta_2\cot\psi_2-\theta_1\cot\psi_1-\ln(\dfrac{\sin\psi_2}{\sin\psi_1})\right)+ \\ ((u-u_0)\cos\beta-w\sin\beta)\cos\beta\cdot\left(-\theta_2\csc^2\psi_{1,2}\dfrac{\partial\theta_2}{\partial w}+\theta_1\csc^2\psi_1\dfrac{\partial\theta_1}{\partial w}\right)\end{array}\right) \tag{A-39}$$

where the terms $\frac{\partial\theta_2}{\partial u}$, $\frac{\partial\theta_1}{\partial u}$ are given by

$$\frac{\partial\theta_{1,2}}{\partial u} = \frac{1}{(w-z_{L1,2})}\frac{(u-u_0-z_{L1,2}\cot\xi)}{(w-z_{L1,2})^2+(u-u_0-z_{L1,2}\cot\xi)^2} = \frac{\cos^2\theta}{(w-z_{L1,2})} \tag{A-40}$$

Similarly, from the gradient tensor matrix, the following relationship holds,

$$\Gamma_{11}^{(L)} = -\Gamma_{33}^{(L)} . \tag{A-41}$$