ARMY RESEARCH LABORATORY

Gravitational and Magnetic Anomaly Inversion Using a Tree-Based Geometry Representation

by Raymond A. Wildman and George A. Gazonas

ARL-RP-248 **June 2009**

A reprint from *Geophysics*, Vol. 74, No. 3, pp. 123–135, May–June 2009.

Approved for public release; distribution is unlimited.

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory
Aberdeen Proving Ground, MD 21005-5069

ARL-RP-248 **June 2009**

Gravitational and Magnetic Anomaly Inversion Using a Tree-Based Geometry Representation

Raymond A. Wildman and George A. Gazonas
Weapons and Materials Research Directorate, ARL

A reprint from *Geophysics*, Vol. 74, No. 3, pp. 123–135, May–June 2009.

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
June 2009	Reprint	May 2009–June 2009

4. TITLE AND SUBTITLE
Gravitational and Magnetic Anomaly Inversion Using a Tree-Based Geometry Representation

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Raymond A. Wildman and George A. Gazonas

5d. PROJECT NUMBER
62105AH84

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
ATTN: AMSRD-ARL-WM-MD
Aberdeen Proving Ground, MD 21005-5069

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-RP-248

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
A reprint from *Geophysics*, Vol. 74, No. 3, pp. 123–135, May–June 2009.

14. ABSTRACT
Gravitational and magnetic anomaly inversion of homogeneous 2D and 3D structures is treated using a geometric parameterization that can represent multiple, arbitrary polygons or polyhedra and a local-optimization scheme based on a hill-climbing method. This geometry representation uses a tree data structure, which defines a set of Boolean operations performed on convex polygons. A variable-length list of points, whose convex hull defines a convex polygon operand, resides in each leaf node of the tree. The overall optimization algorithm proceeds in two steps. Step one optimizes geometric transformations performed on different convex polygons. This step provides approximate size and location data. The second step optimizes the points located on all convex hulls simultaneously, giving a more accurate representation of the geometry. Though not an inherent restriction, only the geometry is optimized, not including material values such as density or susceptibility. Results based on synthetic and measured data show that the method accurately reconstructs various structures from gravity and magnetic anomaly data. In addition to purely homogeneous structures, a parabolic density distribution is inverted for 2D inversion.

15. SUBJECT TERMS
BFGS, optimization, nonlinear inverse methods, Boolean, convex polygons

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			George A. Gazonas
Unclassified	Unclassified	Unclassified	UU	24	19b. TELEPHONE NUMBER (*Include area code*) 410-306-0863

GEOPHYSICS, VOL. 74, NO. 3 (MAY-JUNE 2009); P. I23–I35, 19 FIGS.
10.1190/1.3110042

Gravitational and magnetic anomaly inversion using a tree-based geometry representation

Raymond A. Wildman[1] and George A. Gazonas[1]

ABSTRACT

Gravitational and magnetic anomaly inversion of homogeneous 2D and 3D structures is treated using a geometric parameterization that can represent multiple, arbitrary polygons or polyhedra and a local-optimization scheme based on a hill-climbing method. This geometry representation uses a tree data structure, which defines a set of Boolean operations performed on convex polygons. A variable-length list of points, whose convex hull defines a convex polygon operand, resides in each leaf node of the tree. The overall optimization algorithm proceeds in two steps. Step one optimizes geometric transformations performed on different convex polygons. This step provides approximate size and location data. The second step optimizes the points located on all convex hulls simultaneously, giving a more accurate representation of the geometry. Though not an inherent restriction, only the geometry is optimized, not including material values such as density or susceptibility. Results based on synthetic and measured data show that the method accurately reconstructs various structures from gravity and magnetic anomaly data. In addition to purely homogeneous structures, a parabolic density distribution is inverted for 2D inversion.

INTRODUCTION

Gravitational and magnetic anomaly inversion has applications in many fields, including geophysical prospecting and archeology (Oldenburg, 1974). Fortunately, exact forward models for computing gravitational and magnetic fields caused by either infinite polygonal cylinders (2D) or arbitrary polyhedra (3D) exist so that computational modeling and inversion of such structures is very efficient (Hubbert 1948; Won and Bevis, 1987). Several methods for modeling and inversion have been explored, such as those based on a database of known masses (Bullard and Cooper, 1948), linear splines (Murthy and Rao 1993), neural networks (Osman et al., 2006, Osman et al., 2007) continuous curves (Abdelrahman and Essa, 2005; Essa, 2007), linearization of the nonlinear integral equation (Gao et al., 2007), singular value decomposition (Lines and Treitel, 1984), Fourier transform (Mareschal, 1985), simulated annealing (Mundim et al., 1998), 2D binary grid methods (Krahenbuhl and Li, 2006), and 3D prism methods (Rao et al., 1999).

The method presented here focuses on the data structure used to represent the geometry of the inverted structure. First introduced in Wildman and Weile (2007), the data structure is a binary tree that defines a set of Boolean operations performed on convex polygons. Each convex polygon is defined as the convex hull of a list of points. This representation can then use line segments to approximate any geometry or topology. The method can represent exactly any polygon and also approximate curved structures using an arbitrary number of line segments. Also, multiple shapes are easily represented using the tree structure; because the size of the tree is unrestricted, any number of shapes can be imaged.

This method has a number of advantages. First, contrary to spline-based methods, the number of points used in any single convex hull is not restricted. In spline approaches, using too few points can lead to an inaccurate geometry. If too many points are chosen, the optimization can be inefficient. Second, the number of separate shapes is unrestricted. Again, many spline-based methods require a priori knowledge of the number of separate geometries, and the possible intersection between shapes can be difficult to handle. Third, the size of the search space scales up only as the geometry becomes more complicated. In contrast, pixel-based methods rely on a discretization of the region that can be too coarse or too fine. Fine grids are capable of a higher-resolution image; however, the search space is much larger at the outset. Fourth, Fourier-based methods cannot reproduce nonsmooth structures. The use of line segments allows for accurate representation of corners, and because their length can be arbitrarily small, curves are also well approximated. Finally, the method easily extends to three dimensions. Every geometric concept used has an obvious 3D analog. (See 3D examples in the results section; however, the algorithm is not described in detail because it is

Manuscript received by the Editor 6 March 2008; revised manuscript received 12 November 2008; published online 23 April 2009.
[1]U. S. Army Research Laboratory, Weapons & Materials Research Directorate, Aberdeen Proving Ground, Aberdeen, Maryland, U.S.A. E-mail: raymond.a.wildman@arl.army.mil; gazonas@arl.army.mil.
© 2009 Society of Exploration Geophysicists. All rights reserved.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

essentially identical to the 2D algorithm.) In total, our method makes no assumptions regarding the number or complexity of the objects to be imaged.

Previously, a genetic algorithm (GA) was used with the above geometry representation for the electromagnetic inversion of conducting cylinders (Wildman and Weile, 2007). Although the GA approach showed the viability of the method, it required a significant number of function evaluations; subsequently, a greedy search method and a combined GA/local search method were implemented to improve performance (Wildman and Weile, 2008). The method discussed here forgoes any stochastic optimization and adapts a Newton-like local search method, giving a simplified and more efficient approach. Ultimately, the method uses local search combined with geometric operations discussed in previous work.

The local search is used to optimize two quantities: affine transformations that are applied separately to each convex polygon in a tree and the individual points on the convex hulls of each polygon. Each optimization scheme has advantages. The affine transformation is lower dimensional, leading to more efficient convergence of the local search. Approximate size and location of geometry can be determined using this optimization. Optimizing the individual points is more efficient for recovering detailed shape information. The overall optimization algorithm proceeds in two stages, only using higher dimensional searches after other options have been exhausted. The result is an efficient search algorithm that typically returns results with low relative errors in the gravitational or magnetic field (on the order of 10^{-3} to 10^{-4} as measured in the ℓ_2-norm for simulated targets) using on the order of, depending on the complexity of the target, 10^3 to 10^4 function evaluations.

In the next section, the geometry optimization method is detailed. The geometric data structure is reviewed, and a local optimization-based method for inversion is discussed. The numerical results section explores the use of gravitational and magnetic data as well as measured data from the literature. The final section discusses conclusions and future work.

GEOMETRY OPTIMIZATION

This section details the geometry encoding and optimization scheme. In the first subsection, the key data structure is reviewed and discussed. The second subsection details the optimization algorithm which is a combination of a Newton-like search method and several geometric operations. The objective function, which relates the error in a potential solution to some synthetic or measured data, is presented in the third subsection. Next, the gradient-based search method is reviewed and discussed in context with the given objective function and geometry representation. Finally, the algorithm is reviewed in total through an example.

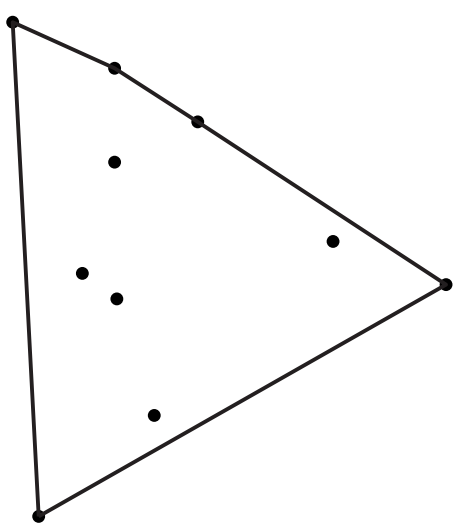

Figure 1. Convex hull of a set of points.

Geometry representation

The basic principle behind this geometry parameterization is the use of convex polygons to build more complex geometries and topologies. Though unnecessary, convex shapes are ideal because they can be represented easily as the convex hull of a list of points. The convex hull of a set of points is defined as the smallest convex set that contains all points or, equivalently, the intersection of all half-planes that contain the points (de Berg et al., 2000). Figure 1 gives an example of the convex hull (solid line) of a set of points (shown as black dots).

Any list of three or more noncollinear points generates a valid convex hull, making self-intersections impossible. A mathematical definition of the convex hull (CH) of a set of N points, $\times = \{\mathbf{p}^{(1)}, \ldots, \mathbf{p}^{(N)}\}$, is given by

$$\mathrm{CH}(\times) = \left\{ \sum_{i=1}^{N} \alpha_i \mathbf{p}^{(i)} \middle| \alpha_i \geq 0\,,\ \forall\, i\,,\ \sum_{i=1}^{N} \alpha_i = 1 \right\}. \quad (1)$$

Though this strict definition defines the set of (infinitely many) points contained in a convex hull, typical algorithms return the subset of points that are located on the vertices of a convex hull (as shown in Figure 1 and denoted as matrix $\mathbf{P}$ throughout) rather than the set of coefficients α_i.

In this work, line segments are used to connect points, giving a linear, polygon-based approximation of target geometries. This is not a limitation of the method because splines or Bézier curves could also be used (Farin and Hansford 2000; Mortenson, 1999). Additionally, the number of points on a convex hull is not restricted in any way, so arbitrarily small line segments can be used to approximate curved structures.

More complex structures, such as multiple and concave shapes, can be generated by combining convex shapes. One way of combining convex polygons is to generate expressions involving Boolean set operations such as union, subtraction, and intersection applied to convex polygon operands. Moreover, mathematical expressions can be represented as binary trees, which can be manipulated with an optimization scheme. Here, the internal nodes of the tree represent Boolean operations and the leaf nodes represent convex polygons.

One difference between this work and Wildman and Weile (2007) is that only union operations are used. Figure 2 gives an example of this scheme. Consider a Boolean expression (shown in Figure 2a as a tree) of the form

$$\mathrm{CH}(\times) \cup \mathrm{CH}(\triangle) \cup \mathrm{CH}(\diamond), \quad (2)$$

where $\times$, $\triangle$, and $\diamond$ represent three separate sets of points (shown in Figure 2b, with dashed lines representing the convex hulls) and $\cup$ represents union.

The first step in generating the geometry defined by expression 2 is to compute the union of the convex polygons represented by $\mathrm{CH}(\times)$ and $\mathrm{CH}(\triangle)$, as shown in Figure 2c. In this case, the polygons

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

are separate, so computing the union is trivial. Next, the union of the result of the previous step (shown in Figure 2d as the shapes with ○s on the vertices) and the convex polygon represented by CH(◇) is computed. In Figure 2d, the result of the union operation is shown as the solid line, and the operands are shown as dotted lines. Figure 2e illustrates the result of expression 2.

Evaluating geometries such as expression 2 requires several computational geometry algorithms. First, there are a number of convex hull algorithms; should structures with a large number of points be necessary, $O(n \log(n))$ (in the total number of points n) algorithms exist. Computing Boolean operations requires an algorithm known as map overlay (de Berg et al., 2000). This algorithm computes a partitioning of the plane based on the input polygons. Intersection points and separate regions are identified so that Boolean operations can be performed. This algorithm also can be computed in $O(n \log(n) + I \log(n))$ time, where the computation time depends on the number of intersections I (de Berg et al., 2000). In this implementation, the Computational Geometry Algorithms Library was used for all 3D geometry operations.

Ultimately, this scheme applies an optimization process to the data structure defined above. The data structure can be summarized as a set of lists or arrays containing 2D (or 3D) points stored in the leaf nodes of a tree. The internal nodes of the tree define some Boolean combination of the convex polygons that each point list represents. An optimization scheme applied to this data structure must then operate on individual points and the actual structure of the tree. A numerical optimization scheme is applied to the convex shapes (directly optimizing the coordinates and geometric transformations), and decisions in the shape of the tree are made based on the performance of several runs of the optimizer.

Optimization algorithm

Given the possible complexity of the geometry representation, an optimization scheme must be designed to construct a target geometry efficiently. To that end, the local optimization method used here gently increases the dimension of the search space by optimizing different geometric transformations and carefully controlling the size of the tree. The method proceeds in two stages: a split stage determines the approximate size, location, and geometric complexity of the target and an optimize stage gives a more accurate image. Each iteration of the split stage divides the leaf nodes of the tree into the union of two separate convex polygons along one or more angles. The best choice is saved and optimized. Once the change in the objective function is sufficiently small, the optimize stage applies higher-dimension searches to the geometry to model the target more accurately.

The optimization algorithm used here is the Broyden-Fletcher-Goldfarb-Shanno (BFGS) local search method (Dennis and Schnabel, 1996). BFGS is a hill-climbing method similar to Newton's method. It differs from Newton's method in that it uses a specific rank-two update to the Hessian matrix of approximated second derivatives. Because the search space is multidimensional, line search is used at a fixed number of search directions to find successive minimized vectors. Throughout this paper, the term *iteration* refers to a single loop through a stage of the global scheme, not separate search directions within BFGS.

The dimension of the space searched by BFGS can be controlled by optimizing various affine transformations, including scaling and translation. An affine transformation, essentially a linear transformation and a translation, can be expressed as

$$\mathbf{x}^* = \mathbf{A}\mathbf{x} + \mathbf{b}, \tag{3}$$

where $\mathbf{A}$ is a square matrix representing the linear transformation and $\mathbf{b}$ is a column vector representing the translation. When applied to a set of 2D points, expression 3 can be rewritten as

$$\mathbf{P}^* = \mathbf{A}(\mathbf{P} - \mathbf{C}) + \mathbf{B} + \mathbf{C},$$

$$\mathbf{B} = \mathbf{b}[11 \ldots 1]_{1\times n},$$

$$\mathbf{C} = \mathbf{c}[11 \ldots 1]_{1\times n}, \tag{4}$$

where $\mathbf{A}$ is a 2×2 matrix representing a linear transformation, $\mathbf{b}$ is a 2D column vector representing a translation vector, $\mathbf{c}$ is a 2D column vector representing the center of mass of the set of points, and $\mathbf{P}$ is a $2 \times n$ matrix representing the components of the n points. (For three dimensions, the dimensions of all matrices and vectors are increased to 3×3 and 3×1, respectively.)

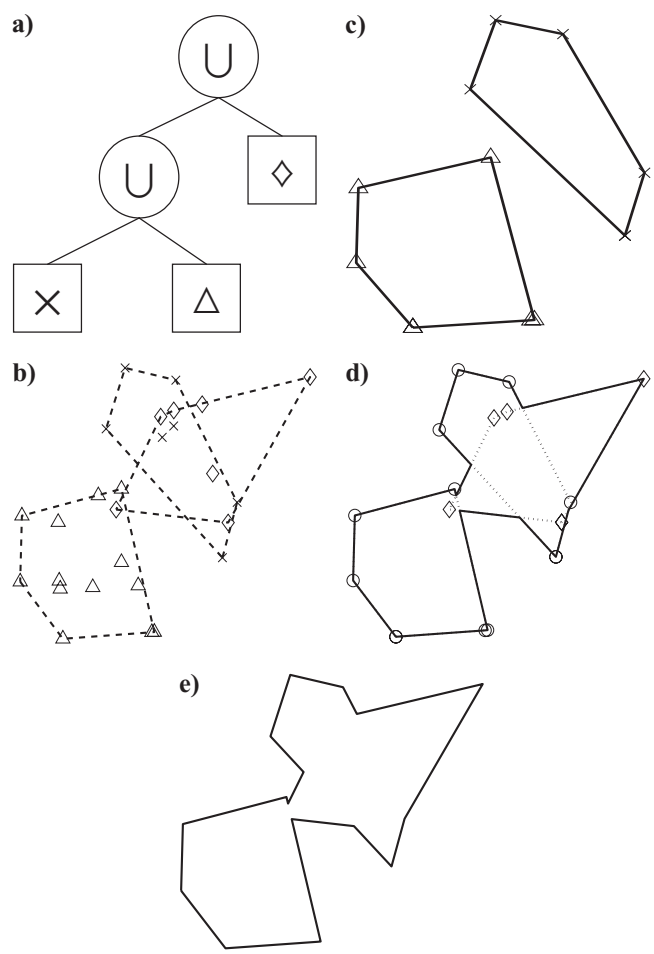

Figure 2. Decoding process. (a) Tree structure. (b) Point lists with their convex hulls. (c) Union of two separate convex hulls. (d) Union with remaining shape. (e) Final result.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

Geometric transforms can be applied (and optimized) by selecting specific parts of an affine transformation given by

$$\mathbf{A}(s_{xx}, s_{xz}, s_{zx}, s_{zz}) = \begin{bmatrix} s_{xx} & s_{xz} \\ s_{zx} & s_{zz} \end{bmatrix}, \quad \mathbf{b}(t_x, t_z) = \begin{bmatrix} t_x \\ t_z \end{bmatrix}. \tag{5}$$

Scaling in two dimensions involves optimizing the quantities s_{xx} and s_{zz}, with

$$\mathbf{A}(s_{xx}, s_{zz}) = \begin{bmatrix} s_{xx} & 0 \\ 0 & s_{zz} \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{6}$$

Similarly, translation in two dimensions involves optimizing the elements of $\mathbf{b}$, t_x, and t_z,

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{b}(t_x, t_z) = \begin{bmatrix} t_x \\ t_z \end{bmatrix}. \tag{7}$$

Combining scaling and translation gives a four-parameter optimization problem:

$$\mathbf{A}(s_{xx}, s_{zz}) = \begin{bmatrix} s_{xx} & 0 \\ 0 & s_{zz} \end{bmatrix}, \quad \mathbf{b}(t_x, t_z) = \begin{bmatrix} t_x \\ t_z \end{bmatrix}. \tag{8}$$

Figure 3 shows an example of an affine transformation applied to a rectangle, characterized by

$$\mathbf{A} = \begin{bmatrix} 1.4 & -0.1 \\ 0.1 & 1.2 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 5 \\ 1.844 \end{bmatrix}, \tag{9}$$

where $\mathbf{A}$ is unitless and $\mathbf{b}$ is measured in kilometers.

In addition, a separate transformation is applied to each point list in a tree. Given k leaf nodes in a tree, optimizing a scaling or translation transformation requires $2k$ parameters, each combined scaling and translation requires $4k$ parameters, and each affine transformation requires $6k$ parameters. The initial guess for each leaf node is chosen as the corresponding identity transformation with some Gaussian random variable (with a small variance) added. (In three dimensions, the corresponding optimizations require $3k$, $6k$, and $12k$ parameters, respectively.)

Individual points are also optimized. Given a set of N points, BFGS must optimize a set of $2N$ parameters (or $3N$ in three dimensions) comprised of the components of each point. The set of optimization parameters is given by

$$\mathbf{P} = \begin{bmatrix} p_x^{(1)} & \cdots & p_x^{(N)} \\ p_z^{(1)} & \cdots & p_z^{(N)} \end{bmatrix}. \tag{10}$$

Again, the initial guess used with BFGS is the original set of points with some small random variable added.

Figure 4 shows a flow chart of the overall optimization scheme. The method is broken into three stages: initialize, split, and optimize. Each stage consists of a loop over a set of optimizations, each with a certain termination criterion. Termination criteria can be a fixed number of iterations, achievement of a fixed absolute error, or stagnation measured as an insufficient change in the error.

Initialize

The initial geometry is always a centered rectangle with x and z dimensions equal to half the specified bounds of the region. This step is denoted as the Rectangle block in Figure 4. Next, the initial geometry is optimized using the scaling and translation transforms given in equations 6 and 7. The two optimizations are repeated a set number of times; typically two or three iterations are sufficient. Using only scaling and translation optimizations gives approximate size and location information.

Split

The split stage determines the geometric and topological complexity of the structure. The splitting operation splits a chosen leaf node along a given line into the union of two separate nodes while adding the intersection points of the line and the hull to the newly created leaf nodes. Including intersection points ensures that the resulting geometry is identical to the original. Figure 5 shows an example of the splitting process: Figure 5a shows the initial and resulting tree structure, and Figure 5b and c shows the corresponding geometries. (The centerline in Figure 5c is for illustrative purposes; the added intersection points ensure that the resulting geometry is identical to the initial.)

The "split all" block attempts to determine the best choice of split between all current convex shapes. Each convex shape is first split along 0° and subsequently 90° and is optimized using the combined translation/scaling given in equation 8. (In three dimensions, the objects are split along the three planes that compose the coordinate axes.) After all combinations of splitting are completed, the best result is saved. Next, optimization of a full affine transformation is applied to the newly split structure; however, the result of this optimization is kept only if one of the termination criteria of the split stage

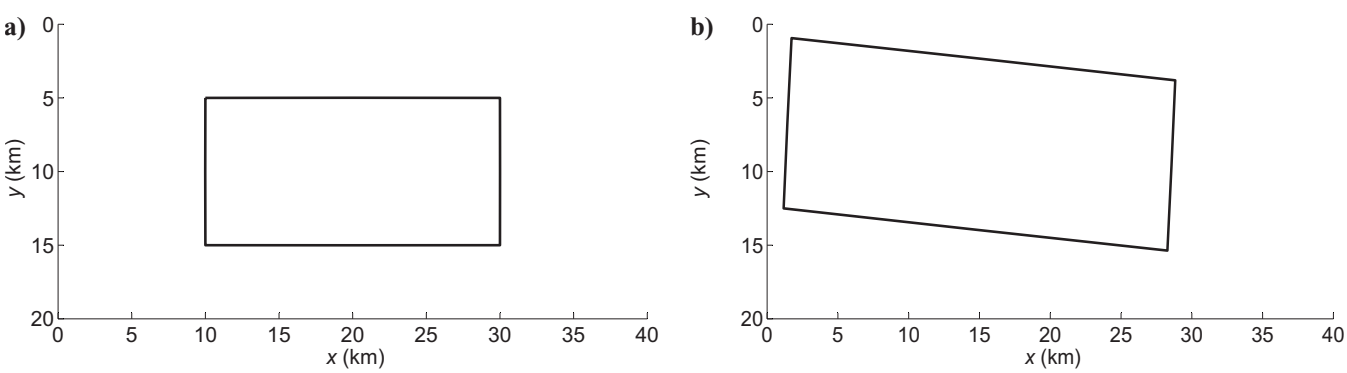

Figure 3. Affine transformation applied to a rectangle. (a) Initial shape. (b) Affine transformation applied.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

is achieved. The termination criteria used here are the reduction below an error of 10^{-2} or the difference between the initial error and error after splitting below 10^{-2}. The former criterion indicates that the structure is likely accurate enough and further subdivision will not aid the final step because the latter indicates that the splitting process has stagnated.

Optimize

The final stage simply optimizes the divided structure alternating between affine transformations (equation 5) and points (equation 10). In addition, new points may be added in between the two optimization runs. In this implementation, a point is added in turn to the midpoint of each line segment (or polygon centroid in three dimensions) and slightly translated outward along the normal direction of the line segment. The objective function of this structure is evaluated, and the new structure with the best objective function value is saved. Finally, the termination criteria are similar to the split stage, though both the error and difference in the error must be below 10^{-3}. In addition, this step can be terminated after four or five iterations.

Objective function

An objective function must be defined to measure the performance of the algorithm relative to some input data. The objective function is then the relative error of a potential solution compared with simulated or actual data produced by a target geometry. Gravitational and magnetic anomalies can be computed efficiently for polygonal geometries such as those produced by the proposed geometry representation in Figure 2 (Won and Bevis, 1987).

For homogeneous polygonal structures (represented as K line segments), the vertical component of the gravity anomaly measured at a station $\mathbf{o}_i$ can be expressed as

$$g_i = 2G\rho \sum_{k=1}^{K} Z_k(\mathbf{o}_i), \tag{11}$$

where G is the gravitational constant, ρ is the density contrast of the structure, and Z_k is a line integral over the kth segment of the structure. Analytical expressions for the line integrals Z_k are available, and one efficient representation is given by Won and Bevis (1987):

$$Z_k(\mathbf{o}_i) = A\left[(\theta_k - \theta_{k+1}) + B\ln\left(\frac{r_{k+1}}{r_k}\right)\right], \tag{12}$$

where

$$A = \frac{(p_x^{(k+1)} - p_x^{(k)})(p_x^{(k)}p_z^{(k+1)} - p_x^{(k+1)}p_z^{(k)})}{(p_x^{(k+1)} - p_x^{(k)})^2 + (p_z^{(k+1)} - p_z^{(k)})^2}, \tag{13}$$

$$B = \frac{p_z^{(k+1)} - p_z^{(k)}}{p_x^{(k+1)} - p_x^{(k)}}, \tag{14}$$

$$r_k^2 = (p_x^{(k)})^2 + (p_z^{(k)})^2, \tag{15}$$

$$\theta_k = \tan^{-1}\left(\frac{p_z^{(k)}}{p_x^{(k)}}\right). \tag{16}$$

The terms $p_x^{(k)}$ and $p_z^{(k)}$ represent the coordinates of the kth vertex of a structure (equation 10, or, e.g., the result of equation 2) relative to the station location $\mathbf{o}_i$, and index arithmetic is performed cyclically. Several special cases exist (e.g., $r_k = 0$), details of which can be found in Won and Bevis (1987). Also, magnetic anomalies $\mathbf{m}_i$ are computed in a similar fashion. Finally, gravity anomalies of polygons with par-

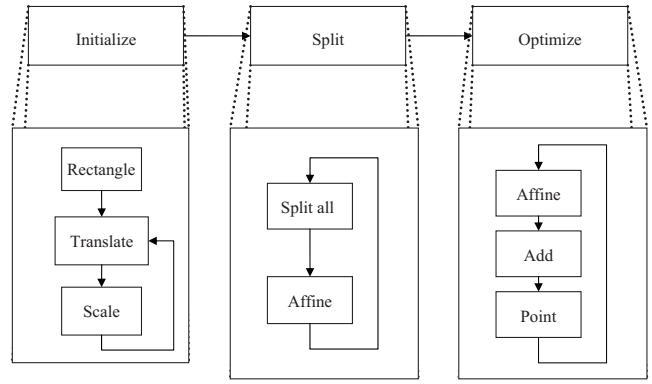

Figure 4. Optimization algorithm flowchart.

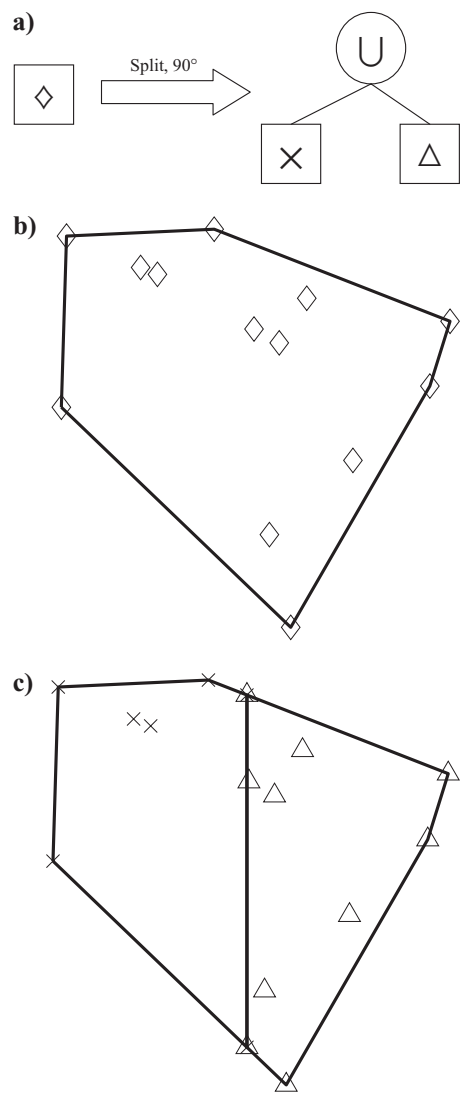

Figure 5. The splitting process. (a) Tree structure. (b) Initial point list with convex hull. (c) Union of two convex hulls at 90°.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

abolic density distributions and 3D polyhedra can be computed as given in Rao et al. (1994) and Singh and Guptasarma (2001).

For 2D geometries, a set of N_s measurements is taken at linearly spaced stations along the $z = 0$, $0 \le x \le y_{\max}$ line segment. (For 3D cases, a uniform grid of observation stations is used to measure the target data.) Although the altitudes of the stations are assumed to be constant, local topography can be modeled by varying the station height z. The objective function is defined as the ℓ_2 errors relative to some measured or synthetic data for gravitational data

$$f_g = \sqrt{\frac{\sum_{i=1}^{N_s} |g_i - g_i^{\text{meas}}|^2}{\sum_{i=1}^{N_s} |g_i^{\text{meas}}|^2}} \tag{17}$$

and magnetic data

$$f_m = \sqrt{\frac{\sum_{i=1}^{N_s} |m_i - m_i^{\text{meas}}|^2}{\sum_{i=1}^{N_s} |m_i^{\text{meas}}|^2}}, \tag{18}$$

where g_i and g_i^{meas} represent the gravitational anomaly at the ith station of the search agent (as computed by equation 11) and target geometry, respectively, and m_i and m_i^{meas} represent the total magnetic anomaly at the ith station. Because the magnetic field $\mathbf{m}$ is a vector, some combination of the magnetic field components m_x, m_y, and m_z must be used to obtain a scalar objective function. The total magnetic field is used and is given by (for 2D cases)

$$m = \sin(\phi^{\text{inc}}) m_z + \sin(\phi^{\text{str}}) \cos(\phi^{\text{inc}}) m_x, \tag{19}$$

where ϕ^{inc} is the inclination angle of the ambient magnetic field measured in degrees below horizontal and ϕ^{str} is the strike angle of the structure relative to magnetic north measured in degrees. In three dimensions, the total magnetic field is

$$m = \sqrt{m_x^2 + m_y^2 + m_z^2}. \tag{20}$$

Finally, noise can be added to synthetic data to simulate a physical system. Here, noise is defined as an additive Gaussian random variable with zero mean and a standard deviation given by some defined signal to noise ratio (S/N), measured in decibels (dB). The measured data is then given by

$$g_i^{\text{meas}} = g_i^{\text{sim}} + \phi(\mu, \sigma)\text{rms}(g^{\text{sim}}),$$

$$\mu = 0, \quad \sigma = 10^{-\text{S/N}/20}, \tag{21}$$

where g_i^{sim} is the simulated gravitational field at the ith station, $\phi(\mu, \sigma)$ is a Gaussian random variable with mean μ and variance σ, and rms$(\cdot)$ indicates the root-mean-square average of the simulated data. A similar expression is defined for the magnetic data m_i^{meas}.

BFGS applied to geometry optimization

BFGS forms the basis of the proposed inversion scheme. It is used to optimize different sets of geometric transformations and also point locations directly; however, the basic search method is the same regardless of the chosen optimization parameters. Again, most steps shown in Figure 4 involve applying BFGS to the current geometry.

To reiterate, BFGS is a Newton-like local search method that uses an approximation of the Hessian matrix of second-order partial derivatives to avoid costly (and possibly inaccurate) second-derivative computations (Dennis and Schnabel, 1996). The method iterates through a number of search directions that are determined by solving

$$\mathbf{H}_k \mathbf{s}_k = -\nabla f(\mathbf{x}_k) \tag{22}$$

for the kth search direction $\mathbf{s}_k$, where $\mathbf{H}_k$ is the approximated Hessian matrix, $\mathbf{x}_k$ is a vector of the independent variables used to compute either f_g or f_m, and the gradient ∇f is computed via finite difference.

The contents of the vector of independent variables depend on which type of optimization is being used because, as discussed above (equations 5–10), scaling, translation, affine transformations, or point locations can be optimized. An affine transformation applied to a single leaf node, for example, is represented by

$$\mathbf{x} = [s_{xx}, s_{xz}, s_{zx}, s_{zz}, t_x, t_z]^{\text{T}}. \tag{23}$$

In total, computing $f(\mathbf{x})$ involves applying the given transformation to the current tree-based geometry (equations 5–8), evaluating the geometry (e.g., as in expression 2), computing the gravitational or magnetic anomaly (equations 12–16), and computing the error relative to the target as given in equations 17 and 18.

With a new search direction in hand, the next point, $\mathbf{x}_{k+1}$, can be found using a line search. The line search proceeds by finding an α_k that minimizes the linear model

$$f^*(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{s}_k), \tag{24}$$

giving a new point

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k. \tag{25}$$

Given a new vector, the Hessian matrix must be updated to find the next search direction. First, the gradient of the function at the new point is computed and the difference of the gradients at the new and previous points is defined as

$$\mathbf{g}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k). \tag{26}$$

The Hessian matrix is then updated using a rank-two approximation, given by

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{\mathbf{g}_k \mathbf{g}_k^{\text{T}}}{\mathbf{g}_k^{\text{T}} \mathbf{s}_k} - \frac{\mathbf{H}_k \mathbf{s}_k (\mathbf{H}_k \mathbf{s}_k)^{\text{T}}}{\mathbf{s}_k^{\text{T}} \mathbf{H}_k \mathbf{s}_k}, \tag{27}$$

which is rank two because it is the combination of two rank-one matrices (i.e., an outer product of two vectors), each with a different basis vector. Finally, the inverse of the Hessian matrix is required to solve equation 22 and can be expressed as

$$\mathbf{H}_{k+1}^{-1} = \mathbf{H}_k^{-1} + \mathbf{g}_k \mathbf{g}_k^{\text{T}} \frac{\alpha_k \mathbf{s}_k^{\text{T}} \mathbf{g}_k + \mathbf{g}_k^{\text{T}} \mathbf{H}_k^{-1} \mathbf{g}_k}{(\mathbf{s}_k^{\text{T}} \mathbf{g}_k)^2} - \frac{\mathbf{H}_k^{-1} \mathbf{g}_k \mathbf{s}_k^{\text{T}} + \mathbf{s}_k \mathbf{g}_k^{\text{T}} \mathbf{H}_k^{-1}}{\mathbf{s}_k^{\text{T}} \mathbf{g}_k}. \tag{28}$$

For the initial approximate Hessian matrix $\mathbf{H}_0$, a diagonal matrix is formed using values corresponding to the inverse squared of the expected magnitude of the variables to be optimized. Expected magnitudes for $\mathbf{A}$ are chosen as one, and magnitudes for $\mathbf{b}$ can be chosen as some fraction of the total region size, given by $x_{\max}$ and $z_{\max}$. For an affine transformation, the initial Hessian matrix is

$$\mathbf{H}_0 = \text{diag}[1,1,1,1(cx_{\max})^{-2}[\cdot](cz_{\max})^{-2}], \tag{29}$$

where diag$[\cdot]$ indicates a diagonal matrix with the argument placed along the diagonal, and c is a constant typically chosen as 0.25. The initial guess $\mathbf{x}_0$ is chosen as the identity operation for the chosen geometric transformation or, for direct point optimization, the set of points of the current geometry. In addition, a small random value chosen from a zero mean Gaussian distribution is added to the initial guess. The variance of the distributions is determined by the typical magnitudes discussed above.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

For an affine transformation, matrix $\mathbf{A}$ is chosen as the identity matrix and the elements of vector $\mathbf{b}$ are set to zero, giving

$$\mathbf{x}_0 = [\phi(1\,,10^{-3})\ \phi(0\,,10^{-3})\ \phi(0\,,10^{-3})\ \phi(1\,,10^{-3}) \times\phi(0\,,cx_{\max}10^{-3})\ \phi(0\,,cz_{\max}10^{-3})]^T \tag{30}$$

where $\phi(\mu\,,\sigma)$ is a Gaussian random variable with mean μ and variance σ.

Finally, BFGS can be run for a given number of search directions or until the objective function value or the magnitude of the gradient falls below some tolerance. Here, a set number of search directions is chosen with a low number (15) for scaling and translation and a higher number (50) for affine transformations and point optimization.

Algorithm example

Figure 6 shows the progress of the algorithm through an example. (Details of the parameters are given in the Results section.) The method begins with a single rectangle (Figure 6a). The set of points that make up the rectangle is labeled $\times_1 = \{\mathbf{p}^{(1)}\,,\ .\ .\ .\ ,\mathbf{p}^{(8)}\}$. Although the initial guess is convex, the result (Figure 6a) is given by

$$\mathbf{P}^* = \mathrm{CH}(\times_1). \tag{31}$$

The matrix of point coordinates $\mathbf{P}^*$ is used to compute the gravitational anomaly (upper portion of Figure 6a) via equation 11, and the objective function value, given by equation 17, is 0.133. Initialization continues by applying a few scale and translate optimizations. A scaling operation (equation 6) is optimized first, given by

$$\mathbf{P}^* = \mathbf{A}(s_{xx}\,,s_{zz})\mathrm{CH}(\times_1). \tag{32}$$

(Here and below, the presentation is simplified by omitting the shift of origin shown in equation 4.)

BFGS is applied to this two-parameter problem, $f_g(s_{xx}\,,s_{zz})$, as discussed in the previous subsection. As shown in equation 22, BFGS requires the gradient of the objective function. This is obtained using a finite-difference approximation; for the example above, it is given by

$$\nabla f_g(s_{xx}\,,s_{zz}) \approx \left[\frac{f_g(s_{xx}+h_1\,,s_{zz}) - f_g(s_{xx}\,,s_{zz})}{h_1}\,,\ \frac{f_g(s_{xx}\,,s_{zz}+h_2) - f_g(s_{xx}\,,s_{zz})}{h_2}\right]. \tag{33}$$

The step size is given by $h_k = \varepsilon_{\mathrm{mach}}^{1/6} x_k^{\mathrm{exp}}$, where $\varepsilon_{\mathrm{mach}}$ is the machine precision and x_k^{exp} is the expected magnitude of the kth argument as discussed in the previous subsection.

Note that each gradient computation requires $K+1$ function evaluations if K is the number of arguments. As K grows, the finite-difference computations can require many function evaluations, though, fortunately, they are all independent and therefore easily parallelizable. The 3D version of the algorithm takes advantage of this as each function evaluation is more costly than in the 2D case.

Next, a translation optimization (equation 7) is performed, given by

$$\mathbf{P}^* = \mathrm{CH}(\times_1) + \mathbf{B}(t_x\,,t_z). \tag{34}$$

Again, this represents a two-parameter optimization problem in t_x and t_z. These optimizations are repeated three times, with the final result shown in Figure 6b. The objective function value at this step is 0.062.

Finally, the result is optimized using a full affine transformation (equation 5) to check for convergence. This operation is given by

$$\mathbf{P}^* = \mathbf{A}(s_{xx}\,,s_{xz}\,,s_{zx}\,,s_{zz})\mathrm{CH}(\times_1) + \mathbf{B}(t_x\,,t_z), \tag{35}$$

and the result is shown in Figure 6c. The objective function value is 0.04, which does not meet the termination criteria.

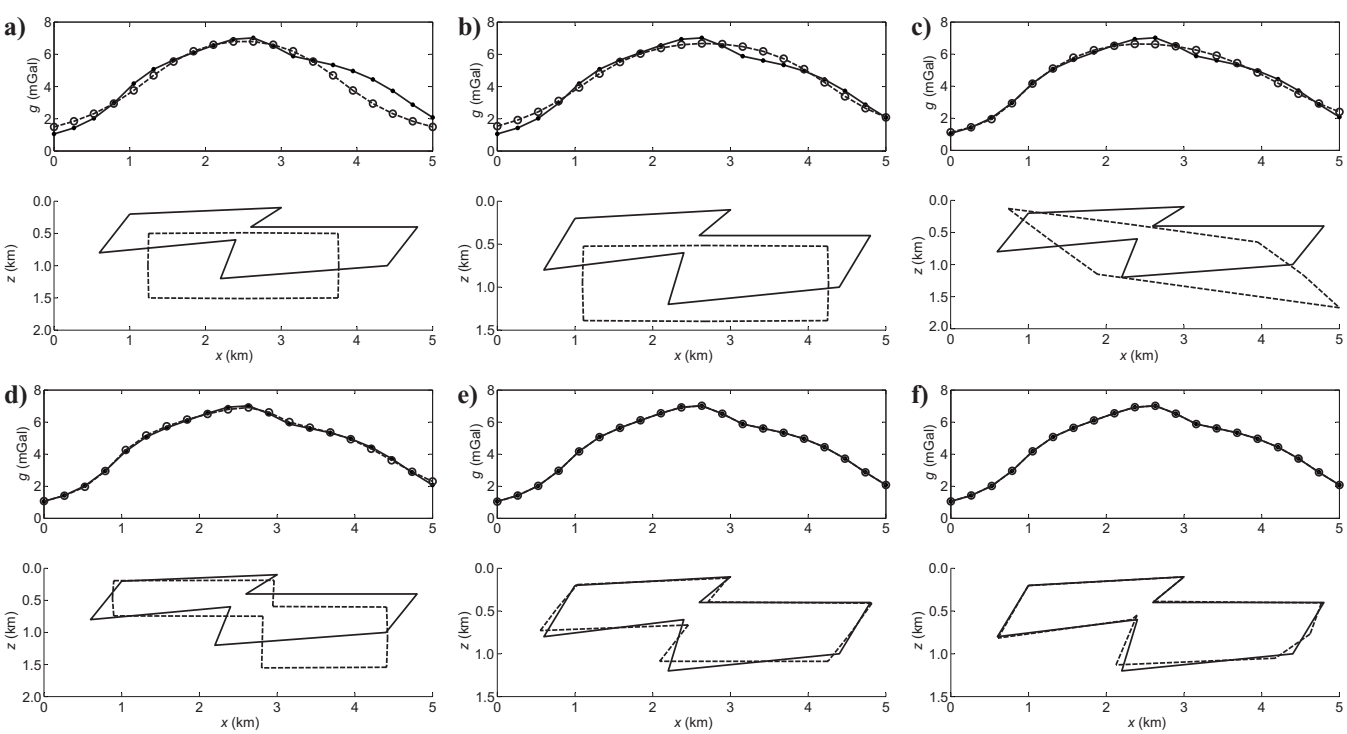

Figure 6. An example of the inversion process. (a) Rectangle. (b) Translation and scaling optimizations. (c) Affine optimization. (d) Splitting. (e) Affine optimization. (f) Point optimization and final result.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

After the initialization stage, the algorithm attempts to add more convex shapes as necessary. The result of the previous stage is split into two point sets, $\times_1$ and $\times_2$, and optimized twice, first after splitting at 0° and again at 90°. Each separate split is optimized using a partial affine operation (equation 8), now given by

$$\mathbf{P}^* = [\mathbf{A}_1(s_{xx}^{(1)}, s_{zz}^{(1)})\mathrm{CH}(\times_1) + \mathbf{B}_1(t_x^{(1)}, t_z^{(1)})] \cup [\mathbf{A}_2(s_{xx}^{(2)}, s_{zz}^{(2)})\mathrm{CH}(\times_2) + \mathbf{B}_2(t_x^{(2)}, t_z^{(2)})], \quad (36)$$

which is an eight-parameter optimization problem. In this example, the better choice was 90° (Figure 6d), the result of which had an objective function value of 0.01. Again, the result is optimized using a full affine transformation to check for convergence, the result of which is shown in Figure 6e. (The optimization problem here is composed of 12 parameters and is similar to that given in equation 36.) This result (Figure 6e) has a sufficiently low error (1.1×10^{-3}) to proceed to the optimize stage.

At this stage, two types of optimizations are run until a termination criterion is achieved. First, another full affine optimization is performed (not shown). Next, new points can be added to either set as described previously. In this example, no additional points were added by the algorithm. Finally, the point sets, $\times_1 = \{\mathbf{p}^{(1)}, \ldots, \mathbf{p}^{(6)}\}$ and $\times_2 = \{\mathbf{q}^{(1)}, \ldots, \mathbf{q}^{(6)}\}$, are optimized directly. The operation used by the optimizer is given by

$$\mathbf{P}^* = \mathrm{CH}[\times_1(\mathbf{p}^{(1)}, \ldots, \mathbf{p}^{(6)})] \cup \mathrm{CH}[\times_2(\mathbf{q}^{(1)}, \ldots, \mathbf{q}^{(6)})], \quad (37)$$

and a total of 24 parameters are required for optimization in this case (six points in both $\times_1$ and $\times_2$). The final result, with an objective function value of 3.8×10^{-5}, is shown in Figure 6f.

RESULTS

Several results, both 2D and 3D and synthetic and measured are presented. The first subsection gives results for synthetic magnetic anomalies and the second comprises synthetic gravitational anomalies. The final subsection gives results for measured data sets of gravitational data. Finally, in each 2D figure, solid lines indicate the target structure and target data (or previously published result) and dashed lines indicate the results of the inversion.

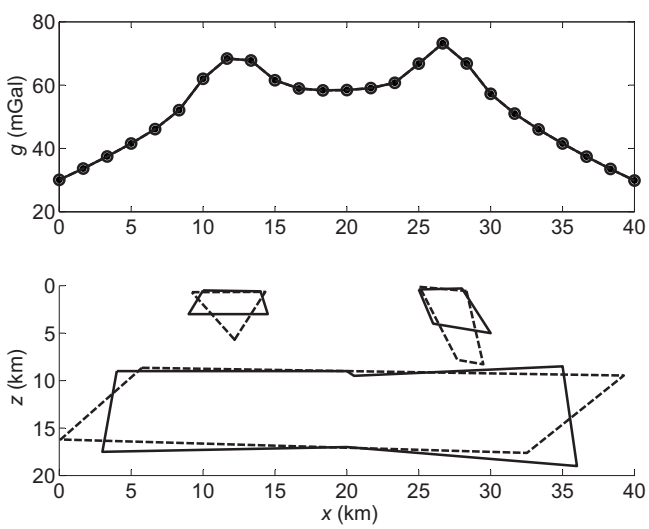

Figure 7. Inversion of a three-mass structure using gravitational data. The target is shown as the solid line and the inversion as the dashed line.

Each example gives the total number of function evaluations, which is a rough measure of the computation time. The 2D method was implemented in MATLAB and the 3D method was implemented in C++. For the MATLAB version, typical run times on a 2-GHz Pentium M laptop are around 7 minutes, though it is important to note that MATLAB is interpreted (not compiled) code and typically is slower than any compiled code. The 3D version was run on a cluster, with the finite-difference computations run in parallel. Run times for this code are on the order of an hour.

Gravitational anomaly

As a synthetic example, a fault was simulated as shown in the example given in the previous section. Twenty stations were spaced equally over 5 km; the density contrast was assumed to be $\Delta\rho = 0.276$ g/cm^3. Figure 6f shows the target as the solid line, the optimized result as the dashed line, and the simulated and inverted gravity anomalies at the top. This result required 5486 function evaluations to achieve an error of 3.8×10^{-5}.

Next, a synthetic structure made up of three masses — one large, deep mass and two shallow, smaller masses as shown in Figure 7 — was used as a target. Again, a set of 25 stations was placed over 40 km and the density contrast was the same as above. Figure 7 shows the results of the inversion, which achieved an error of 1.9×10^{-4} after 11,112 function evaluations.

Additionally, the structure was inverted in the presence of corrupted data. First, a noise level of 50 dB was used, corresponding to a noise level of approximately 0.14 mGal (rms average). The algorithm was run twice, each time with a different set of noisy data. The results are shown in Figure 8. Next, a noise level of 40 dB was used, giving an rms noise level of approximately 0.48 mGal. Figure 9 shows the results for two separate runs.

As a 3D example, a two-body structure was simulated with two randomly generated convex shapes placed at different depths. This example used 30 stations and profiles over a 30×30 km region and a density contrast of $\Delta\rho = 0.5$ g/cm^3. Figure 10 shows the simulated and inverted gravity contours. The result shown in Figure 11 required 12,688 function evaluations to reach a final error of 1.3×10^{-3}. In this figure (and other 3D plots), depth is indicated by the shading, i.e., the shallowest parts of the structure are shown in white and the deepest are shown in black.

Magnetic anomaly

The three-mass structure from the previous subsection was also imaged using magnetic data. The structure covers approximately 40 km and is approximately 20 km deep. A magnetic anomaly was generated assuming a susceptibility of $\chi = 10^{-3}$ (in SI units), a strike angle of 60°, and an ambient magnetic field of 50 μT at an inclination angle of 0°. Data were measured at a set of 25 stations from 0 to 40 km. After 27,122 function evaluations, the result shown in Figure 12 was achieved; it has an error of 2.2×10^{-4}.

Finally, a synthetic 3D example was inverted. The target structure was composed of three randomly generated convex shapes as shown in Figure 14a. The structure had a susceptibility of $\chi = 10^{-3}$ and the ambient field was assumed to have a strength of 50 μT at an inclination angle of 60° and a declination angle of 0°. The final result, which used data from a 30×30 point grid over a 20×20 km area, had an

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

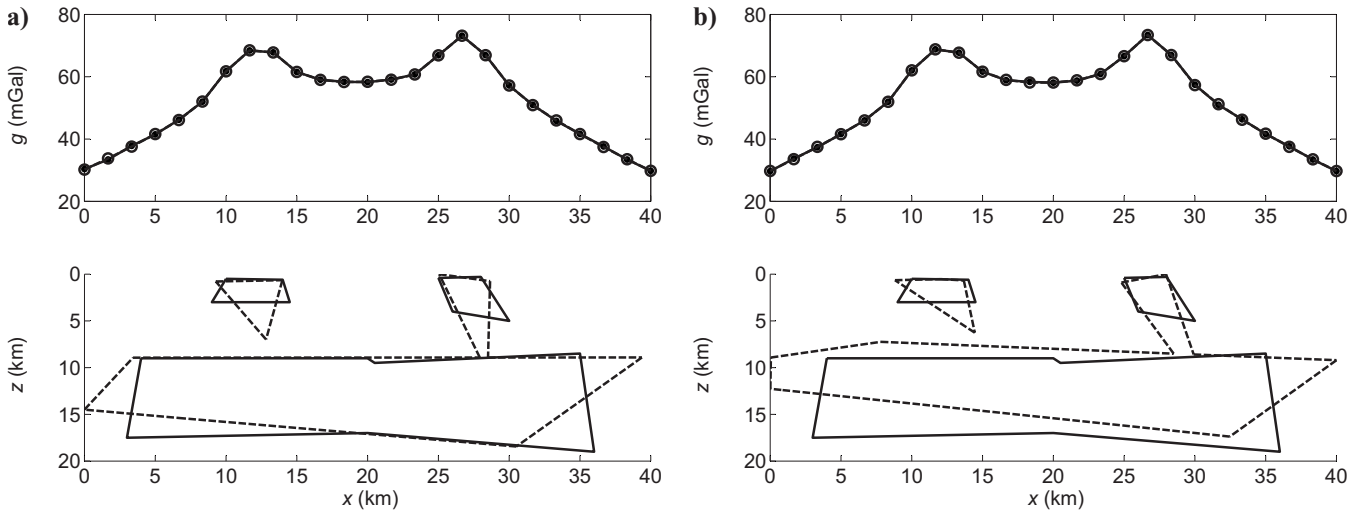

Figure 8. Three-mass structure inverted in the presence of noisy data with an S/N of 50 dB. (a) Run 1. (b) Run 2.

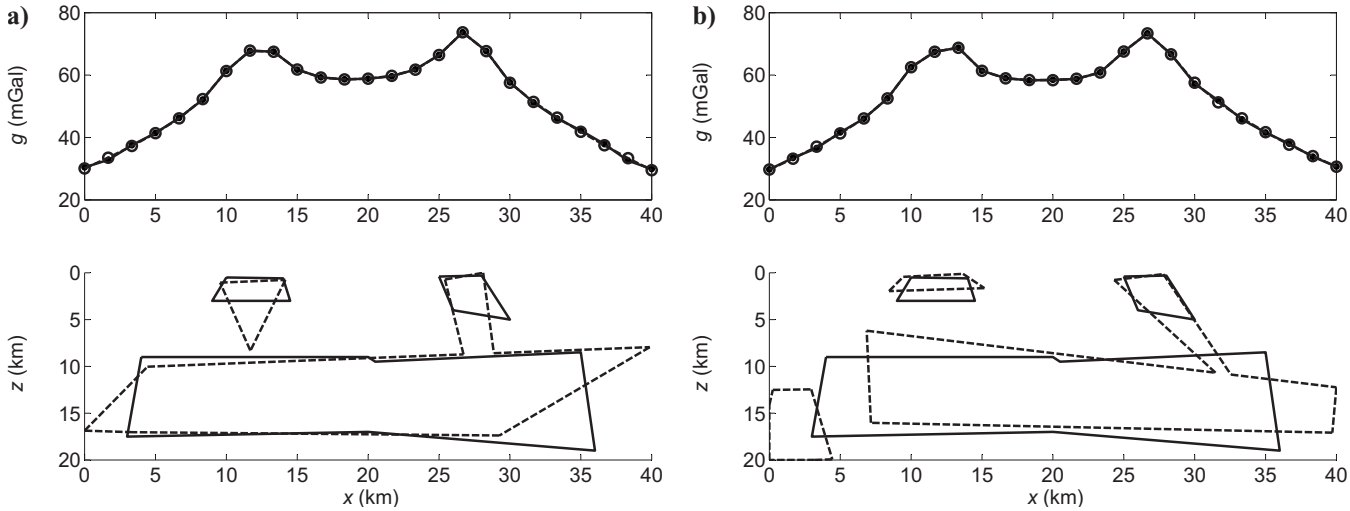

Figure 9. Three-mass structure inverted in the presence of noisy data with an S/N of 40 dB. (a) Run 1. (b) Run 2.

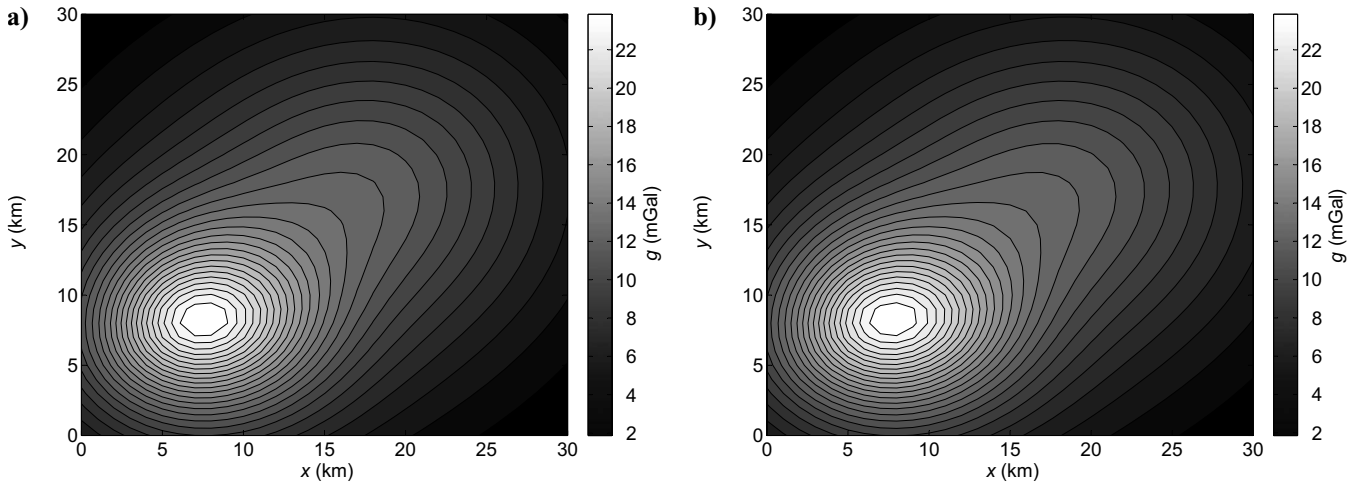

Figure 10. Gravity anomaly contours (in milligals) over a synthetic structure. (a) Target. (b) Inversion.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

error of 1.0×10^{-3} and required 29,901 function evaluations. Results of the inversion are shown in Figures 13 and 14.

Measured data sets

Several measured data sets were also inverted, the first of which is an aeromagnetic anomaly described in Murthy et al. (2001). There, a fault structure was assumed and several parameters, such as intensity of magnetization, depth, and position, were found. Here, an unknown, linearly varying regional magnetic field is assumed and the two coefficients are found during the optimization. The two coefficients of the regional field were added to the vector **x** at each step. The data set comprised 26 stations over flat terrain, between 0 and 50 km. The values found in Murthy et al. (2001) are used: Remnant magnetization was assumed with an intensity of magnetization of 148 nT at an angle of $-97°$, a total field inclination angle of 35°, and a strike angle of 90°.

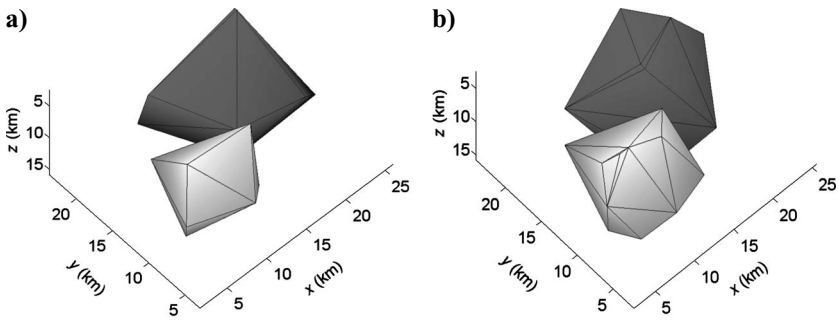

Figure 11. (a) Target and (b) inverted structures.

The final result, shown in Figure 15, required 3403 function evaluations to reach a minimum error of 4.2×10^{-3}. The regional magnetic field had a slope of -1.44 nT/km and an offset of 66 nT. Also, because a fault was being imaged, the result in Figure 15 actually extends to approximately 300 km, though only the relevant portion of the fault is shown. Good agreement is seen with the result in Murthy et al. (2001), especially the top of the fault. Both results are around 7 km.

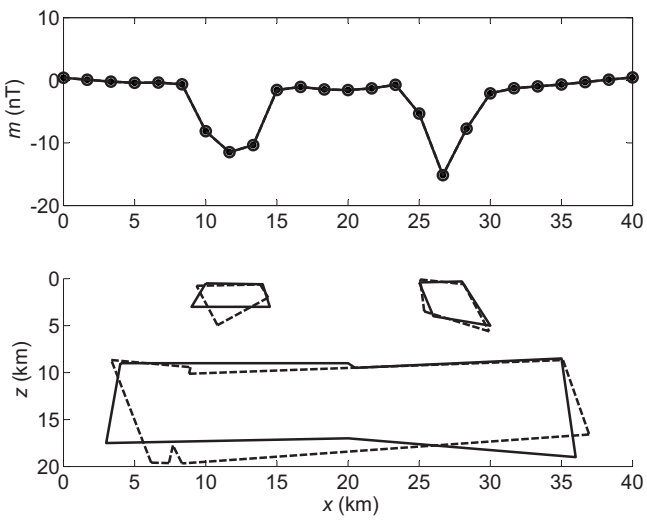

Figure 12. Inversion of a three-mass structure using magnetic data. Target is shown as the solid line and the inversion as the dashed line.

The first gravity anomaly, observed over the Weardale granite body using 23 stations across 55 km of flat terrain and presented in Murthy and Rao (1993), is shown in Figure 16. Again, a linearly varying regional gravitational field was assumed and optimized simultaneously with the geometry at each step. The optimization algorithm required 16,380 function evaluations to achieve a final error of 5.0×10^{-3}. Figure 16 shows the final geometry, with a density contrast, as assumed in Murthy and Rao (1993), of $\Delta\rho = -0.13$ g/cm^3; it is similar to the result in Murthy and Rao (1993). The regional field variation had a slope of 4.72×10^{-3} mgal/km and an offset of -0.133 mgal. As in Murthy and Rao (1993), the top of the structure does not outcrop at the surface.

Next, a basin structure with a parabolic density distribution was inverted. The Bouguer gravity anomaly data was taken from Rao et al. (1994) and had an initial density of $\Delta\rho_0 = -0.5206$ g/cm^3 and decay parameter $\alpha = 0.0576$ km^{-1}. The data set was taken from 13 stations over 48 km of flat terrain. The final result (Figure 17) had an error of 6.0×10^{-4} (5688 function evaluations) and is compared to the result shown in Rao et al. (1994) (solid line). As a basin, the inverted structure should have a flat top across the entire region, as-

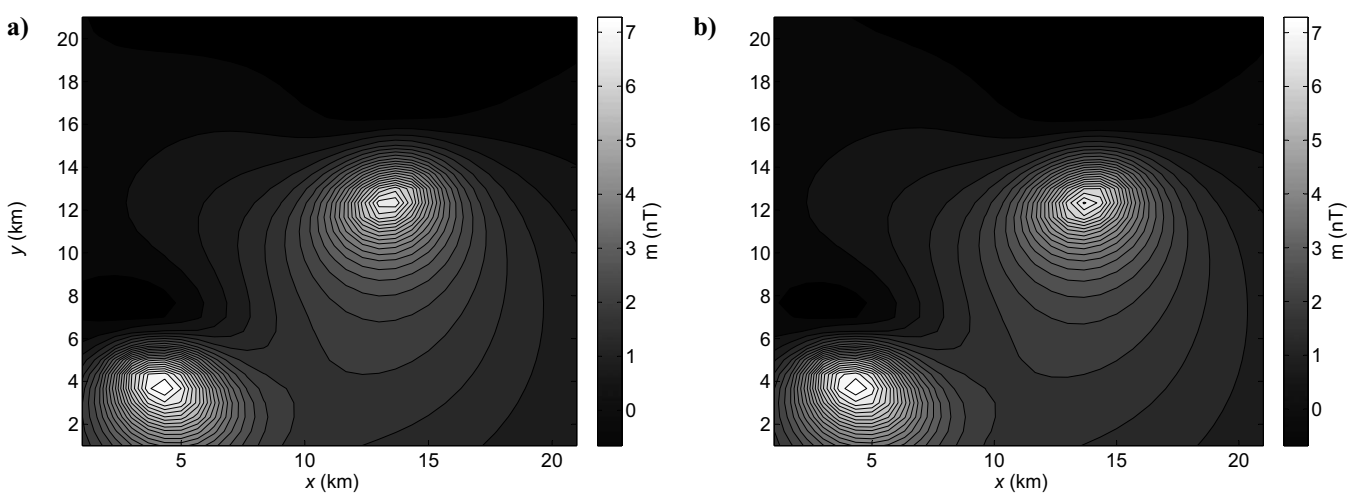

Figure 13. Magnetic anomaly contours (in nanoteslas) over a structure composed of three random bodies. (a) Target. (b) Inversion.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

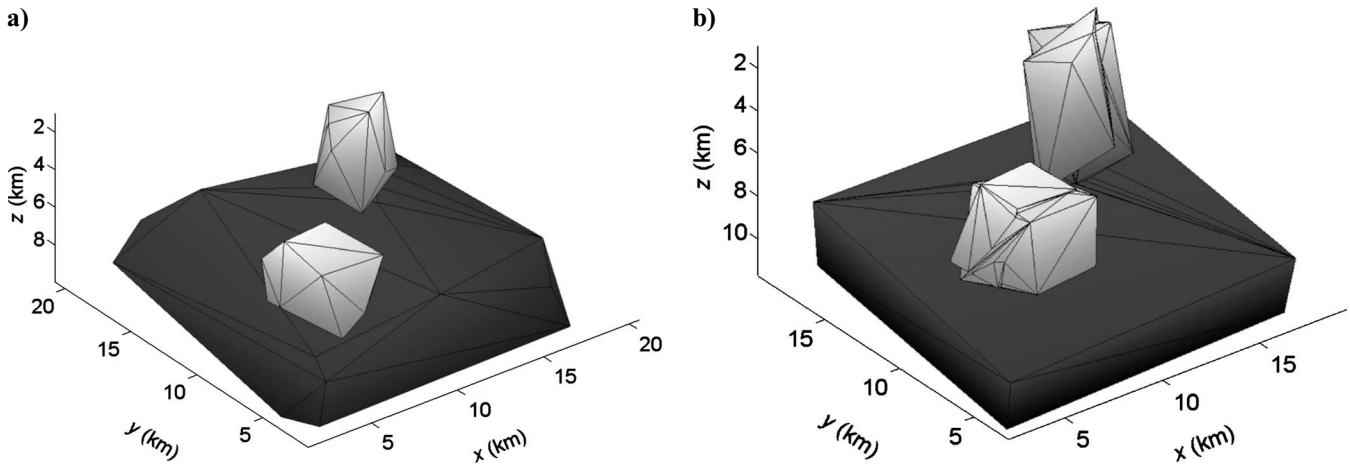

Figure 14. (a) Target and (b) inverted three-body structure.

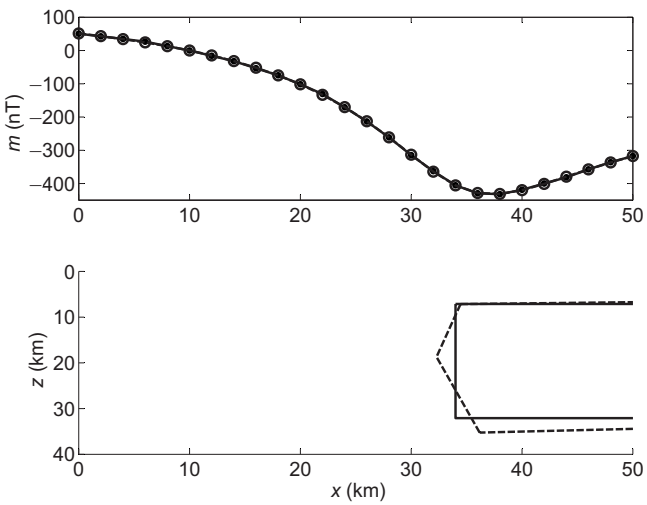

Figure 15. Inversion of aeromagnetic anomaly near Dehri, Bihar, India. Solution from Murthy et al. (2001) is shown as the solid line.

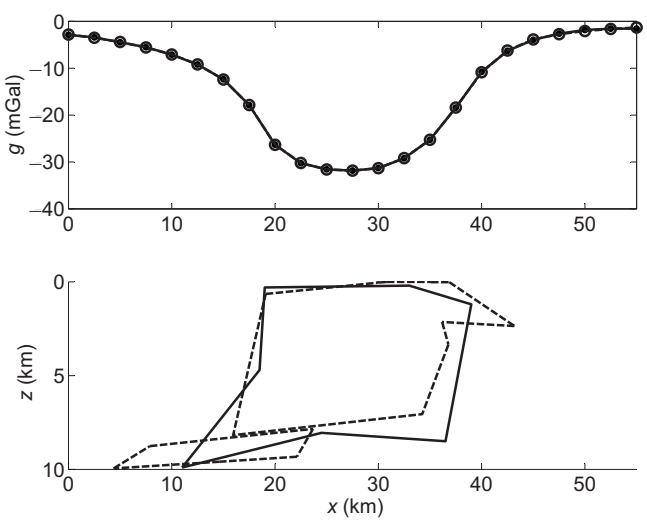

Figure 16. Inversion of Weardale granite body. Solution from Murthy and Rao (1993) is shown as the solid line.

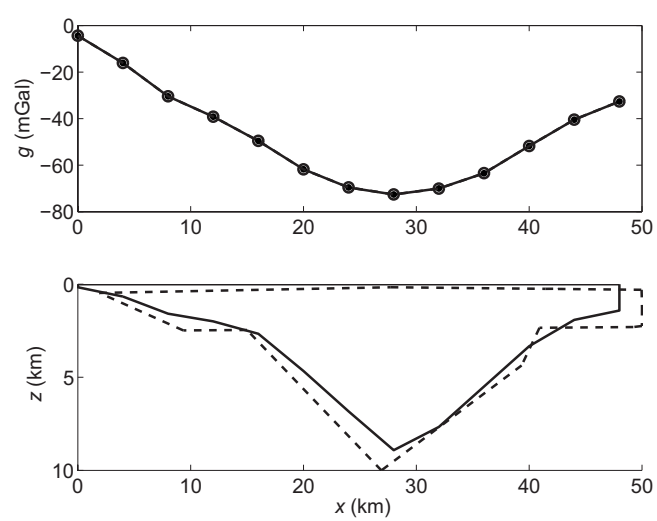

Figure 17. Inversion of Los Angeles basin. Solution from Rao et al. (1994) is shown as the solid line.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

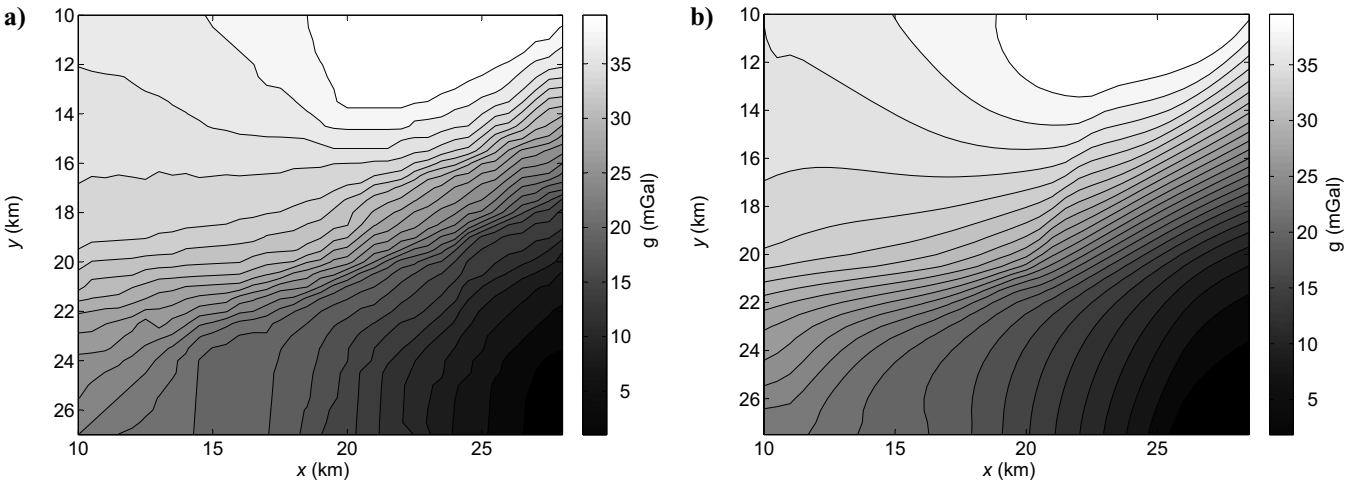

Figure 18. Gravity anomaly contours (in milligals) over the Gelibolu Peninsula. (a) Measured. (b) Inversion.

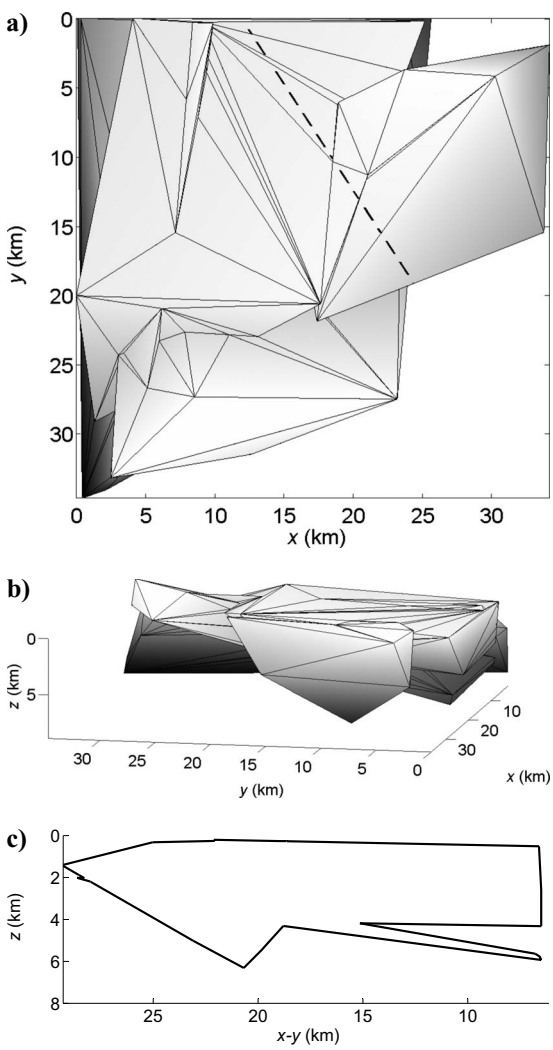

Figure 19. Interpreted structure at the Gelibolu Peninsula. (a) Top view. (b) Perspective view along dashed line. (c) Contour taken in the plane indicated by the dashed line.

sumed in Rao et al. (1994). Here, although no assumptions were made, the top of the structure is mostly flat with some small deviation toward the left side.

As a final example, a measured 3D data set was inverted. The Bouguer anomaly was adapted from Figure 12 in Albora et al. (2007), as shown in Figure 18a. In this inversion, 37 stations were used along 35 profiles over a flat area covering 18.5×17.5 km. The density contrast was assumed to be $\Delta\rho = 0.276$ g/cm^3. The final result, whose anomaly contour plot is shown in Figure 18b, required 42,465 function evaluations and had an error of 1.56×10^{-2}. Figure 19 shows three views of the interpreted structure: top down view (Figure 19a), perspective view perpendicular to the plane indicated by a dashed line in the previous figure (Figure 19b), and a contour taken in the plane indicated by the dashed line (Figure 19c). In Albora et al. (2007), a fault structure was predicted along the dashed line on the right side of Figure 19a. This structure appears in the 3D result shown here, verified in Figure 19c.

CONCLUSIONS

Our method for inverting gravitational and magnetic anomaly data uses a tree-based geometry description that combines arbitrary convex shapes using Boolean operations. This approach is flexible in that no knowledge of the number, approximate location, or complexity of the target geometry is required. A BFGS-based optimization algorithm uses successive runs of the local optimizer applied to geometric transforms and actual points. Separate stages in the algorithm first determine approximate shape and complexity and finally details of the structure. Results showed that different types of geometries, both two and three dimensions, could be reconstructed accurately using both gravitational and magnetic information.

In its current form, the method has some limitations that could be overcome. First, only homogeneous structures were treated here. The method could be extended to inhomogeneous structures by altering the operations of the tree structure. Material values would be added to the leaf nodes in the tree and optimized along with the geometry. Also, as geometries become more complex, the method attempts to search larger spaces. This could be overcome by including only a few of the total number of leaf nodes in each optimization. Another possibility is the use of optimizers other than BFGS to avoid

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

costly finite-difference computations. Stochastic, derivative-free methods could be used or even automated differentiation with adjoint formulations.

ACKNOWLEDGMENTS

This work was supported in part by an appointment to the Postgraduate Research Participation Program at the U. S. Army Research Laboratory (USARL) administered by the Oak Ridge Institute for Science and Education through an interagency agreement between the U. S. Department of Energy and USARL. In addition, the authors would like to thank Juan Garcia-Abdeslem for his time and thorough reviews.

REFERENCES

Abdelrahman, E. M., and K. S. Essa, 2005, Magnetic interpretation using a least-squares, depth-shape curves method: Geophysics, **70**, no. 3, L23–L30.

Albora, A. M., O. N. Ucan, and D. Aydogan, 2007, Modelling potential field sources in the Gelibolu Peninsula (western Turkey) using a Markov random field approach: Pure and Applied Geophysics, **164**, 1057–1080.

Bullard, E. C., and R. I. B. Cooper, 1948, The determination of the masses necessary to produce a given gravitational field: Proceedings of the Royal Society of London Series A, — Mathematical and Physical Sciences, **194**, 332–347.

de Berg, M., M. van Kreveld, M. Overmars, and O. Schwarzkopf, 2000, Computational geometry: Algorithms and applications, 2nd ed.: Springer.

Dennis, J. E., and R. B. Schnabel, 1996, Numerical methods for unconstrained optimization and nonlinear equations: SIAM.

Essa, K. S., 2007, Gravity data interpretation using the *s*-curves method: Journal of Geophysics and Engineering, **4**, 204–213.

Farin, G., and D. Hansford, 2000, The essentials of CAGD: A. K. Peters.

Gao, E.-G., S.-Y. Song, and S.-D. Liu, 2007, Robust iterative inversion of density interface based on gravity anomaly: Journal of the University of Science and Technology of China, 916–921.

Hubbert, M. K., 1948, A line integral method of computing the gravimetric effects of two-dimensional masses: Geophysics, **13**, 215–225.

Krahenbuhl, R. A., and Y. G. Li, 2006, Inversion of gravity data using a binary formulation: Geophysical Journal International, **167**, 543–556.

Lines, L. R., and S. Treitel, 1984, Tutorial — A review of least-squares inversion and its application to geophysical problems: Geophysical Prospecting, **32**, 159–186.

Mareschal, J. C., 1985, Inversion of potential field data in Fourier transform domain: Geophysics, **50**, 685–691.

Mortenson, M. E., 1999, Mathematics for computer graphics applications: An introduction to the mathematics and geometry of CAD/CAM, geometric modeling, scientific visualization, and other CG: Industrial Press.

Mundim, K. C., T. J. Lemaire, and A. Bassrei, 1998, Optimization of non-linear gravity models through generalized simulated annealing: Physica A, **252**, 405–416.

Murthy, I. V. R., and P. R. Rao, 1993, Inversion of gravity and magnetic anomalies of two-dimensional polygonal cross sections: Computers & Geosciences, **19**, 1213–1228.

Murthy, I. V. R., K. V. Swamy, and S. J. Rao, 2001, Automatic inversion of magnetic anomalies of faults: Computers & Geosciences, **27**, 315–325.

Oldenburg, D. W., 1974, Inversion and interpretation of gravity anomalies: Geophysics, **39**, 526–536.

Osman, O., A. M. Albora, and O. N. Ucan, 2006, A new approach for residual gravity anomaly profile interpretations: Forced neural network (FNN): Annals of Geophysics, **49**, 1201–1208.

———, 2007, Forward modeling with forced neural networks for gravity anomaly profile: Mathematical Geology, **39**, 593–605.

Rao, C. V., V. Chakravarthi, and M. L. Raju, 1994, Forward modeling: Gravity anomalies of two-dimensional bodies of arbitrary shape with hyperbolic and parabolic density functions: Computers and Geosciences, **20**, 873–880.

Rao, P. R., K. V. Swamy, and I. V. R. Murthy, 1999, Inversion of gravity anomalies of three-dimensional density interfaces: Computers & Geosciences, **25**, 887–896.

Singh, B., and D. Guptasarma, 2001, New method for fast computation of gravity and magnetic anomalies from arbitrary polyhedra: Geophysics, **66**, 521–526.

Wildman, R. A., and D. S. Weile, 2007, Geometry reconstruction of conducting cylinders using genetic programming: IEEE Transactions on Antennas and Propagation, **55**, 629–636.

———, 2008, Greedy search and a hybrid local optimization/genetic algorithm for tree-based inverse scattering: Microwave and Optical Technology Letters, **50**, 822–825.

Won, I. J., and M. Bevis, 1987, Computing the gravitational and magnetic anomalies due to a polygon: Algorithms and Fortran subroutines: Geophysics, **52**, 232–238.

Downloaded 30 Apr 2009 to 128.175.13.10. Redistribution subject to SEG license or copyright; see Terms of Use at http://segdl.org/

NO. OF COPIES	ORGANIZATION
1 (PDF only)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA 8725 JOHN J KINGMAN RD STE 0944 FORT BELVOIR VA 22060-6218
1	DIRECTOR US ARMY RESEARCH LAB IMNE ALC HRR 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB AMSRD ARL CI OK TL 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIRECTOR US ARMY RESEARCH LAB AMSRD ARL CI OK PE 2800 POWDER MILL RD ADELPHI MD 20783-1197
	ABERDEEN PROVING GROUND
1	DIR USARL AMSRD ARL CI OK TP (BLDG 4600

NO. OF COPIES	ORGANIZATION
1 (CD only)	DPTY ASSIST SCT FOR R&T SARD TT ASA (ACT) J PARMENTOLA THE PENTAGON RM 3E479 WASHINGTON DC 20310-1714
1	PRIN DPTY FOR TCHNLGY HQ US ARMY MATCOM AMCDCGR R PRICE 9301 CHAPEK RD FT BELVOIR VA 22060-5527
3	AIR FORCE ARMAMENT LAB AFATL DLJW W COOK D BELK J FOSTER EGLIN AFB FL 32542
2	DARPA L CHRISTODOULOU W COBLENZ 3701 N FAIRFAX DR ARLINGTON VA 22203-1714
1	DIRECTOR US ARMY ARDEC AMSRD AAR AEE W E BAKER BLDG 3022 PICATINNY ARSENAL NJ 07806-5000
2	US ARMY TARDEC AMSTRA TR R MS 263 K BISHNOI D TEMPLETON MS 263 WARREN MI 48397-5000
1	COMMANDER US ARMY RSRCH OFC AMSRD ARL RO EN B LAMATTINA PO BOX 12211 RESEARCH TRIANGLE PARK NC 27709-2211
1	WORCESTER POLYTECHNIC INST MATHEMATICAL SCI K LURIE WORCESTER MA 01609

NO. OF COPIES	ORGANIZATION
1	COMMANDER US ARMY RSRCH OFC AMSRD ARL RO MM J LAVERY PO BOX 12211 RESEARCH TRIANGLE PARK NC 27709-2211
1	COMMANDER US ARMY RSRCH OFC AMSRD ARL RO EM D STEPP PO BOX 12211 RESEARCH TRIANGLE PARK NC 27709-2211
5	NAVAL RESEARCH LAB E R FRANCHI CODE 7100 M H ORR CODE 7120 J A BUCARO CODE 7130 J S PERKINS CODE 7140 S A CHIN BING CODE 7180 4555 OVERLOOK AVE SW WASHINGTON DC 20375
1	DTRA M GILTRUD 8725 JOHN J KINGMAN RD FORT BELVOIR VA 22060
1	ERDC US ARMY CORPS OF ENGINEERS USACEGSL P PAPADOS 7701 TELEGRAPH RD ALEXANDRIA VA 22315
1	CLEMSON UNIV DEPT MECH ENGINEERS M GRUJICIC 241 ENGRG INNOVATION BLDG CLEMSON SC 29634-0921
1	UNIV OF CALIFORNIA CTR OF EXCELLENCE FOR ADV MATLS S NEMAT NASSER SAN DIEGO CA 92093-0416
1	VIRGINIA POLYTECHNIC INST COLLEGE OF ENGRG R BATRA BLACKSBURG VA 24061-0219

NO. OF COPIES	ORGANIZATION
5	DIRECTOR LANL P MAUDLIN R GRAY W R THISSELL A ZUREK F ADDESSIO PO BOX 1663 LOS ALAMOS NM 87545
7	DIRECTOR SANDIA NATL LABS J BISHOP MS 0346 E S HERTEL JR MS 0382 W REINHART MS 1181 T VOGLER MS 1181 L CHHABILDAS MS 1811 M FURNISH MS 1168 M KIPP MS 0378 PO BOX 5800 ALBUQUERQUE NM 87185-0307
1	DIRECTOR LLNL M J MURPHY PO BOX 808 LIVERMORE CA 94550
3	CALTECH M ORTIZ MS 105 50 G RAVICHANDRAN T J AHRENS MS 252 21 1201 E CALIFORNIA BLVD PASADENA CA 91125
5	SOUTHWEST RSRCH INST C ANDERSON K DANNEMANN T HOLMQUIST G JOHNSON J WALKER PO DRAWER 28510 SAN ANTONIO TX 78284
1	TEXAS A&M UNIV DEPT OF MATHEMATICS J WALTON COLLEGE STATION TX 77843
1	UNIV OF DELAWARE DEPT ELECTRICAL & CMPTR ENGRG D WEILE NEWARK DE 19716

NO. OF COPIES	ORGANIZATION
2	SRI INTERNATIONAL D CURRAN D SHOCKEY 333 RAVENSWOOD AVE MENLO PARK CA 94025
8	UNIV OF NEBRASKA DEPT OF ENGRG MECH D ALLEN F BOBARU Y DZENIS G GOGOS M NEGAHBAN R FENG J TURNER Z ZHANG LINCOLN NE 68588
1	JOHNS HOPKINS UNIV DEPT OF MECH ENGRG K T RAMESH LATROBE 122 BALTIMORE MD 21218
4	UNIV OF UTAH DEPT OF MATH A CHERKAEV E CHERKAEV E S FOLIAS R BRANNON SALT LAKE CITY UT 84112
1	PENN STATE UNIV DEPT OF ENGRG SCI & MECH F COSTANZO UNIVERSITY PARK PA 168023
2	UNIV OF DELAWARE DEPT OF MECH ENGRG T BUCHANAN T W CHOU 126 SPENCER LAB NEWARK DE 19716
2	INST OF ADVANCED TECH UNIV OF TX AUSTIN S BLESS H FAIR 3925 W BRAKER LN STE 400 AUSTIN TX 78759-5316

NO. OF COPIES	ORGANIZATION
2	UNIV OF DELAWARE CTR FOR COMPST MATRLS J GILLESPIE M SANTARE 126 SPENCER LAB NEWARK DE 19716
1	COMPUTATIONAL MECH CONSULTANTS J A ZUKAS PO BOX 11314 BALTIMORE MD 21239-0314
1	LOUISIANA STATE UNIV R LIPTON 304 LOCKETT HALL BATON ROUGE LA 70803-4918
1	APPLIED RSCH ASSOCIATES D E GRADY 4300 SAN MATEO BLVD NE STE A220 ALBUQUERQUE NM 87110
1	INTERNATIONAL RSRCH ASSOC INC D L ORPHAL 4450 BLACK AVE PLEASANTON CA 94566
3	ORNL ENVIRONMENTAL SCI DIV W DOLL T GAMEY L BEARD PO BOX 2008 OAK RIDGE TN 37831
1	UNIV OF ILLINOIS DEPT OF MECHL SCI & ENGRG A F VAKAKIS 1206 W GREEN ST MC 244 URBANA CHAMPAIGN IL 61801
1	UNIV OF ILLINOIS ARSPC ENGRG J LAMBROS 104 S WRIGHT ST MC 236 URBANA CHAMPAIGN IL 61801

NO. OF COPIES	ORGANIZATION
2	WASHINGTON ST UNIV INST OF SHOCK PHYSICS Y M GUPTA J ASAY PULLMAN WA 99164-2814
1	NORTHWESTERN UNIV DEPT OF CIVIL & ENVIRON ENGRG Z BAZANT 2145 SHERIDAN RD A135 EVANSTON IL 60208-3109
1	UNIV OF DAYTON RSRCH INST N S BRAR 300 COLLEGE PARK MS SPC 1911 DAYTON OH 45469
2	TEXAS A&M UNIV DEPT OF GEOPHYSICS MS 3115 F CHESTER T GANGI COLLEGE STATION TX 778431
1	UNIV OF SAN DIEGO DEPT OF MATH & CMPTR SCI A VELO 5998 ALCALA PARK SAN DIEGO CA 92110
1	NATIONAL INST OF STANDARDS & TECHLGY BLDG & FIRE RSRCH LAB J MAIN 100 BUREAU DR MS 8611 GAITHERSBURG MD 20899-8611
1	MIT DEPT ARNTCS ASTRNTCS R RADOVITZKY 77 MASSACHUSETTS AVE CAMBRIDGE MA 02139
1	MIT DEPT MATLS SCI ENGRG E THOMAS 77 MASSACHUSETTS AVE CAMBRIDGE MA 02139

NO. OF COPIES	ORGANIZATION
2	MATERIALS SCI CORP A CAIAZZO R LAVERTY 181 GIBRALTAR RD HORSHAM PA 19044
1	DIR USARL AMSRD ARL D V WEISS 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIR USARL AMSRD ARL SE J PELLEGRINO 2800 POWDER MILL RD ADELPHI MD 20783-1197
1	DIR USARL AMSRD ARL SE SP A EDELSTEIN 2800 POWDER MILL RD ADELPHI MD 20783-1197

ABERDEEN PROVING GROUND

NO. OF COPIES	ORGANIZATION
82	DIR USARL AMSRD ARL WM B FORCH S KARNA J MCCAULEY P PLOSTINS J SMITH T WRIGHT AMSRD ARL WM B J NEWILL M ZOLTOSKI AMSRD ARL WM BA D LYON AMSRD ARL WM BC P WEINACHT AMSRD ARL WM BD P CONROY R PESCE RODRIGUEZ B RICE AMSRD ARL WM BF W OBERLE AMSRD ARL WM M R DOWDING S MCKNIGHT AMSRD ARL WM MA J ANDZELM R JENSEN A RAWLETT M VANLANDINGHAM E WETZEL AMSRD ARL WM MB M BERMAN T BOGETTI M CHOWDHURY W DE ROSSET W DRYSDALE A FRYDMAN D HOPKINS L KECSKES T H LI S MATHAUDHU M MINNICINO B POWERS J TZENG AMSRD ARL WM MC R BOSSOLI S CORNELISON M MAHER W SPURGEON AMSRD ARL WM MD J ADAMS B CHEESEMAN E CHIN K CHO B DOOLEY C FOUNTZOULAS G GAZONAS J LASALVIA P PATEL C RANDOW J SANDS B SCOTT R WILDMAN C F YEN AMSRD ARL WM SG T ROSENBERGER AMSRD ARL WM T P BAKER AMSRD ARL WM TA S SCHOENFELD M BURKINS AMSRD ARL WM TB N ELDREDGE J STARKENBERG

NO. OF COPIES	ORGANIZATION
	AMSRD ARL WM TC
	T BJERKE
	T FARRAND
	K KIMSEY
	M FERMEN COKER
	D SCHEFFLER
	S SCHRAML
	S SEGLETES
	AMSRD ARL WM TD
	S BILYK
	D CASEM
	J CLAYTON
	D DANDEKAR
	N GNIAZDOWSKI
	M GREENFIELD
	R KRAFT
	B LOVE
	M RAFTENBERG
	E RAPACKI
	M SCHEIDLER
	T WEERASOORIYA
	AMSRD ARL WM TE
	J POWELL
	B RINGERS
	G THOMSON
	AMSRD ARL VT UV
	S WILKERSON
	AMSRD ARL VT RP
	J BORNSTEIN

INTENTIONALLY LEFT BLANK.