GEOPHYSICS, VOL. 67, NO. 5 (SEPTEMBER-OCTOBER 2002); P. 1532–1541, 5 FIGS.
10.1190/1.1512749

3-D magnetic inversion with data compression and image focusing

Oleg Portniaguine* and Michael S. Zhdanov*

ABSTRACT

We develop a method of 3-D magnetic anomaly inversion based on traditional Tikhonov regularization theory. We use a minimum support stabilizing functional to generate a sharp, focused inverse image. An iterative inversion process is constructed in the space of weighted model parameters that accelerates the convergence and robustness of the method. The weighting functions are selected based on sensitivity analysis. To speed up the computations and to decrease the size of memory required, we use a compression technique based on cubic interpolation.

Our method is designed for inversion of total magnetic anomalies, assuming the anomalous field is caused by induced magnetization only. The method is applied to synthetic data for typical models of magnetic anomalies and is tested on real airborne data provided by Exxon-Mobil Upstream Research Company.

INTRODUCTION

Interpretation of 3-D magnetic data over inhomogeneous geological structures is a challenging problem in exploration geophysics. Despite significant progress made over the last decade, inversion of magnetic survey data still has many practical difficulties. The major difficulty is related to theoretical nonuniqueness of the magnetic inverse problem. It is well known that there exist magnetic mass distributions generating zero external fields. These nonradiating masses cause equivalence in inverse problem solution, which can be overcome only by introducing a priori information about the geological structures. Several methods have been developed for dealing with the nonuniqueness problem. Most of these methods are based on the parametric inversion, where the geometric parameters of the model are fixed and the parameters inverted for are the magnetic susceptibilities on the grid within the given geometry (e.g., Bhattacharyya, 1980; Rao and Babu, 1991).

Another approach to the solution of this problem was taken by Li and Oldenburg (1996). They applied the powerful tool of a general inversion method to solve the underdetermined problem, with the number of cells significantly larger than the amount of data available. Li and Oldenburg used a priori information to select the desired geological model from a class of possible solutions. This goal was reached by constructing a model objective function with appropriate weighting functions. The parameters of the weighting functions were selected empirically, based on numerical modeling and qualitative analysis of typical magnetic anomalies. Note that the objective function introduced in Li and Oldenburg (1996) has the flexibility to construct many different models that generate practically the same data.

We develop an inversion method based on traditional Tikhonov regularization theory. The objective function (the Tikhonov parametric functional) consists of two terms: a misfit functional and a stabilizing functional. The misfit functional is responsible for fitting the observed data with synthetic data predicted for the given model. The stabilizing functional incorporates information about the basic properties of the type of models used in the inversion. We suggest using the minimum support stabilizing functional, similar to the one introduced by Last and Kubic (1983), for compact 2-D inversion of gravity data. This functional helps generate a sharp, focused inverse image similar to the 3-D gravity inversion considered in Portniaguine and Zhdanov (1999a). The main difference between our approach and the one discussed by Last and Kubic (1983) is in constructing an iterative inversion process in the space of the weighted model parameters. The weighting functions are selected based on sensitivity analysis. They provide equal sensitivity of the observed data to the cells located at different depths and at different horizontal positions. Thus, our weighting functions automatically introduce appropriate corrections for the vertical and horizontal distribution of the anomalous susceptibility. This is one of the main differences between our approach and the one developed by Li and Oldenburg (1996).

Another difficulty in magnetic inverse problems is related to the enormous areal coverage of modern magnetic surveys,

Manuscript received by the Editor May 2, 2000; revised manuscript received January 4, 2002.
*University of Utah, Department of Geology and Geophysics, Salt Lake City, Utah 84112. E-mail: oleg@cs.utah.edu; mzhdanov@mines.utah.edu.

especially in airborne magnetic exploration. Processing a large amount of data collected in an airborne survey requires access to a huge data file stored on a hard drive, which slows the inversion process. To speed the computations and to decrease the amount of memory required, we use the compression technique we outlined earlier in Portniaguine and Zhdanov (1999b) and Portniaguine (1999). We now consider a method with a higher compression factor, based on using cubic polynomials in the compression algorithm.

Our inversion method is designed to invert any component of the anomalous magnetic field, including the total magnetic anomaly, under the assumption that the anomalous field is caused by induced magnetization only.

The code is applied to synthetic data for typical models of magnetic anomalies. It is also tested on real airborne magnetic data, provided by ExxonMobil Upstream Research Company.

FORWARD MODELING OF MAGNETIC ANOMALIES

We divide the lower half-space into small rectangular cells, each filled by magnetic masses with intensity of magnetization $\mathbf{I}(\mathbf{r})$, which is given as a product of the magnetic susceptibility $\chi(\mathbf{r})$, the strength of the inducing geomagnetic field $\mathbf{H}^0$, and its direction, given by a vector $\mathbf{l}$ of unit length.

We denote the coordinates of the cell center as $\mathbf{r}_k = (x_k, y_k, z_k)$, where $k = 1, \ldots, N_m$, and the cell sides as dx, dy, dz. Also, we have a discrete number of observation points $\mathbf{r}'_n = (x'_n, y'_n, 0)$, where $n = 1, \ldots, N_d$. The field at point n from a small cell k with unit susceptibility (the magnetic field kernel f_{nk}) is equal to

$$f_{nk} = H^0\left[\frac{3(\mathbf{l}\cdot\mathbf{r})^2}{\|\mathbf{r}\|^2} - 1\right]\frac{dx\,dy\,dz}{\|\mathbf{r}\|^3}, \tag{1}$$

where $\mathbf{r} = \mathbf{r}'_n - \mathbf{r}_k$ is the vector between the observation point and the cell center. The magnetized small cubic cell is approximated by a dipole located at its center.

The discrete forward modeling operator for total field magnetic anomalies produced by the arbitrary distribution of susceptibility can be expressed in matrix notation as

$$\mathbf{d} = \hat{\mathbf{F}}\mathbf{m}. \tag{2}$$

Here, $\mathbf{m}$ is a vector of model parameters (each component of that vector is the magnetic susceptibility χ of the corresponding cell) of length N_m, $\mathbf{d}$ is a vector of the observed data of length N_d, and $\hat{\mathbf{F}}$ is a rectangular matrix of size $N_d \times N_m$, formed by the corresponding magnetic field kernels [equation (1)].

COMPRESSION IN SOLVING INVERSE PROBLEMS

Expression (2) becomes a matrix equation if the data $\mathbf{d}$ are given and $\mathbf{m}$ is unknown. The matrix $\hat{\mathbf{F}}$ of equation (2) is a full matrix. In the 3-D case, the size of $\hat{\mathbf{F}}$ is large. To store it efficiently, we represent it as a product of sparse matrices. This also speeds the algorithm as a result of the use of sparse arithmetic.

In the magnetic inverse problem, the data dimension N_d is commonly smaller than the model dimension N_m. This suggests applying compression to the model side of $\hat{\mathbf{F}}$. That produces incomplete factorization of $\hat{\mathbf{F}}$:

$$\hat{\mathbf{F}} \approx \hat{\mathbf{F}}_{mc}\hat{\mathbf{W}}^T_{mr}, \tag{3}$$

where T denotes matrix transposition, $\hat{\mathbf{F}}_{mc}$ is a compressed matrix of the forward operator,

$$\hat{\mathbf{F}}_{mc} = threshold\left(\hat{\mathbf{F}}\hat{\mathbf{W}}^T_{mc}, \epsilon\right), \tag{4}$$

and $\hat{\mathbf{W}}_{mc}$ and $\hat{\mathbf{W}}_{mr}$ are the model compression and restoration matrices, respectively, with dimensions of $N_m \times N_m$. Parameter ϵ is a threshold level (in percent) that determines the accuracy of restoration. In actual applications, we set ϵ equal to the noise level in the data.

Substituting equation (3) into equation (2), we obtain

$$\hat{\mathbf{F}}_{mc}\hat{\mathbf{W}}^T_{mr}\mathbf{m} = \mathbf{d}. \tag{5}$$

Formula (5) provides the compressed form of the inverse problem equation.

The greater the amount of information under compression, the higher the compression factor, which is determined as a ratio of the total number of elements of the matrix to the number of nonzero elements. Model side compression not only allows the use of a fine model grid in the lateral direction (without running out of memory to store a huge full matrix), but it also makes it possible to use regular small cells at every depth in the model. This significantly simplifies optimal mesh generation and also streamlines handling and representing the results. The basic principles of the compression technique are outlined in Appendices A and B.

REGULARIZED SOLUTION OF THE MAGNETIC INVERSE PROBLEM IN THE COMPRESSED FORM

In this section we apply the conjugate gradient method for solving a 3-D magnetic inverse problem. We first describe the conventional conjugate gradient method. Remarkably, this method is very versatile. Applied to an overdetermined linear problem, the conjugate gradient method produces the least-squares solution. Applied to an underdetermined linear problem, the method converges to the minimum norm solution. We also demonstrate that the linear problem with Tikhonov regularization can be reformulated easily as a conjugate gradient for the overdetermined problem. In this approach, the regularization parameter must be chosen iteratively. Finally, we consider the basic principles of focusing inversion and introduce a reweighted optimization algorithm for a stable focusing solution of the magnetic inverse problem.

Conjugate gradient method for linear inverse problem solution

The solution of compressed inverse problem (5) is found iteratively according to the following formulas (Fletcher, 1981):

Compressed version		Uncompressed version	
$\mathbf{l}_i = \hat{\mathbf{W}}_{mr}\left(\hat{\mathbf{F}}^T_{mc}\mathbf{r}_{i-1}\right)$	(a)	$\mathbf{l}_i = \hat{\mathbf{F}}^T\mathbf{r}_{i-1}$	(h)
$s_i = \mathbf{l}_i^T\mathbf{l}_i$	(b)		
$\mathbf{h}_i = \mathbf{l}_i + \mathbf{h}_{i-1}\frac{s_i}{s_i - 1}$	(c)		
$\mathbf{f}_i = \hat{\mathbf{F}}_{mc}\left(\hat{\mathbf{W}}^T_{mr}\mathbf{h}_i\right)$	(d)	$\mathbf{f}_i = \hat{\mathbf{F}}\mathbf{h}_i$	(i)
$k_i = \frac{\mathbf{f}_i^T\mathbf{r}_i}{\mathbf{f}_i^T\mathbf{f}_i}$	(e)		
$\mathbf{m}_i = \mathbf{m}_{i-1} - k_i\mathbf{h}_i$	(f)		
$\mathbf{r}_i = \mathbf{r}_{i-1} - k_i\mathbf{f}_i$	(g)		

, (6)

where i is the iteration number, $\mathbf{r}$ is the residual vector, $\mathbf{l}$ is the gradient vector, s is its length, $\mathbf{h}$ is the conjugate direction vector in the space of models, $\mathbf{f}$ is its projection in the space of data, and k is the step length, a scalar. The starting values (for $i=0$) are

$$\begin{cases} \mathbf{m}_0 = 0 & (a) \\ \mathbf{r}_0 = \hat{\mathbf{F}}\mathbf{m}_0 - \mathbf{d} = -\mathbf{d} & (b)\,. \\ s_0 = 1 & (c) \end{cases} \tag{7}$$

Note that in equation (6) the matrix-to-vector multiplications in items (a) and (d) take the most computer time. An uncompressed version of the algorithm is produced by substituting items (h) and (i) for (a) and (d), respectively. Two sparse multiplications in (a) and (d) are much faster than one multiplication by a full matrix in (i) and (h). That is why the compression method speeds up the algorithm.

If the number of parameters in vector $\mathbf{m}$, which we denote as N_m, is not equal to the number of data points in vector $\mathbf{d}$ (denoted as N_d), then $\hat{\mathbf{F}}$ is rectangular. Interestingly, the conjugate gradient method can be applied even in this case.

For an underdetermined problem (where $N_m > N_d$), the conjugate gradient iterations (6) converge to the minimum norm solution $\mathbf{m}_{min}$:

$$\mathbf{m}_{min} = \hat{\mathbf{F}}^T(\hat{\mathbf{F}}\hat{\mathbf{F}}^T)^{-1}\mathbf{d}. \tag{8}$$

Expression (8) is also known as the Riesz representation formula (Parker, 1994).

Regularized conjugate gradient method

The original magnetic inverse problem and its reformulation in the compressed form [equation (5)] are ill posed because of the nonuniqueness and instability of the solution. The conventional way of solving ill-posed inverse problems, according to the regularization theory (Tikhonov and Arsenin, 1977; Zhdanov, 2002), is based on the minimization of the Tikhonov parametric functional, $P^{\alpha}(\mathbf{m})$:

$$P^{\alpha}(\mathbf{m}) = \|\hat{\mathbf{F}}\mathbf{m} - \mathbf{d}\|^2 + \alpha\|\mathbf{m}\|^2, \tag{9}$$

where $\|\hat{\mathbf{F}}\mathbf{m} - \mathbf{d}\|^2$ is a misfit functional between theoretical values $\hat{\mathbf{F}}\mathbf{m}$ and the observed data $\mathbf{d}$, $\|\mathbf{m}\|^2$ is a minimum norm stabilizing functional, and α is a regularization parameter.

The problem of parametric functional minimization,

$$P^{\alpha}(\mathbf{m}) = \min, \tag{10}$$

can be reformulated to apply formula (6). Consider the linear inverse problem:

$$\begin{bmatrix} \hat{\mathbf{F}} \\ \sqrt{\alpha}\hat{\mathbf{I}} \end{bmatrix} \mathbf{m} = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix}, \tag{11}$$

where $\hat{\mathbf{I}}$ is the unit matrix. Two matrices in square brackets denote a single matrix created by appending the two:

$$\hat{\mathbf{A}}_1 = \begin{bmatrix} \hat{\mathbf{F}} \\ \sqrt{\alpha}\hat{\mathbf{I}} \end{bmatrix}, \quad \mathbf{d}_1 = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix}. \tag{12}$$

For example, vector $\mathbf{d}_1$ is created from vector $\mathbf{d}$ by appending a zero vector $\mathbf{0}$ to its tail. Matrix $\hat{\mathbf{A}}_1$ is created by appending a diagonal matrix (with $\sqrt{\alpha}$ on the main diagonal) to matrix $\hat{\mathbf{F}}$.

Equation (11) is the result of adding extra equations to the original equation (2). The number of existing equations in the original formula is N_d. The number of additional equations is equal to the number of free parameters N_m, so the system of linear equations (11) always contains more equations ($N_m + N_d$) than unknowns (N_m), i.e., it is overdetermined. For an overdetermined system, the conjugate gradient method converges to the least-squares solution. This is equivalent to the minimization of the parametric functional expressed in combined matrix notations:

$$\|\hat{\mathbf{A}}_1\mathbf{m} - \mathbf{d}_1\|^2 = \min\,. \tag{13}$$

Reformulating equation (9) as equation (13) and applying formula (6) to the minimization of formula (13), we arrive at the conventional regularized conjugate gradient method (Zhdanov, 2002).

To select an optimal regularization parameter α, we use the Tikhonov method. First, α is set to balance the contribution of a misfit and a stabilizer after the first iteration of a conjugate gradient method:

$$\mathbf{m}_1 = k_1\hat{\mathbf{F}}^T\mathbf{d}, \quad \mathbf{r}_1 = -\mathbf{d} - k_1\hat{\mathbf{F}}\mathbf{m}_1, \quad \alpha_1 = \frac{\|\mathbf{r}_1\|}{\|\mathbf{m}_1\|}. \tag{14}$$

The subsequent iterative values are determined by decreasing α to one-half of its previous value (Tikhonov and Arsenin, 1977):

$$\alpha_{i+1} = \frac{\alpha_i}{2}.$$

The process stops when the value of the misfit functional decreases below the noise level in the data ϕ:

$$\|\mathbf{r}_i\| < \phi.$$

Method of reweighted optimization

In our previous paper (Portniaguine and Zhdanov, 1999a) we introduced a minimum support stabilizing functional $s_{MS}(\mathbf{m})$ to generate a sharp, focused inverse gravity problem solution, similar to the one developed by Last and Kubik (1983):

$$s_{MS}(\mathbf{m}) = \sum_{k=1}^{N_m} \frac{m_k^2}{m_k^2 + \beta^2}, \tag{15}$$

where $\beta > 0$ is a small positive number.

Substituting the minimum norm stabilizing functional in formula (9) by formula (15), we obtain

$$P^{\alpha}(\mathbf{m}) = \|\hat{\mathbf{F}}\mathbf{m} - \mathbf{d}\|^2 + \alpha \sum_{k=1}^{N_m} \frac{m_k^2}{m_k^2 + \beta^2} = \min, \tag{16}$$

where β is a small number needed to avoid the singularity when $m_k = 0$. Thus, the focusing inversion is reduced to the solution of the minimization problem (16). The problem is solved using reweighted optimization (O'Leary, 1990).

To account for the different sensitivities of the data to the model parameters, we have to use an additional weighting matrix $\hat{\mathbf{W}}_m$ for the model parameters. Mehanee et al. (1998) and

Portniaguine and Zhdanov (1999a) have shown that the matrix $\hat{\mathbf{W}}_m$ with this property can be determined as the square root of the integrated sensitivity matrix:

$$\hat{\mathbf{W}}_m = \sqrt{\hat{\mathbf{S}}}, \tag{17}$$

where $\hat{\mathbf{S}}$ is a diagonal matrix formed by the integrated sensitivities of $\mathbf{d}$ to the parameter m_k, determined as the ratio

$$S_k = \frac{\|\delta \mathbf{d}\|}{\delta m_k} = \sqrt{\sum_i (F_{ik})^2}. \tag{18}$$

In formula (18), F_{ik} are the elements of the forward modeling matrix $\hat{\mathbf{F}}$. We denote the diagonal elements of the matrix $\hat{\mathbf{W}}_m$ by $\{w_1, w_2, \ldots, w_k, \ldots, w_{N_m}\}$.

Let us consider the minimization problem with the minimum support stabilizer, weighted with sensitivity weights w_k:

$$P^\alpha(\mathbf{m}) = \|\mathbf{Fm} - \mathbf{d}\|^2 + \alpha \sum_{k=1}^{Nm} \frac{w_k^2 m_k^2}{m_k^2 + \beta^2} = min. \tag{19}$$

We introduce an iterative weighting matrix as follows:

$$\hat{\mathbf{W}}^2(\mathbf{m}) = \mathbf{diag}[\mathbf{m}^2 + \beta^2 \mathbf{I}]\hat{\mathbf{W}}_m^{-2}, \tag{20}$$

where $\mathbf{diag}[\mathbf{m}^2 + \beta^2 \mathbf{I}]$ is a diagonal matrix formed by the elements $m_k^2 + \beta^2$.

Now we can reformulate problem (19) using matrix notation:

$$P^\alpha(\mathbf{m}) = \|\hat{\mathbf{F}}\hat{\mathbf{W}}(\mathbf{m})\hat{\mathbf{W}}^{-1}(\mathbf{m})\mathbf{m} - \mathbf{d}\|^2 + \alpha\|\hat{\mathbf{W}}^{-1}(\mathbf{m})\mathbf{m}\|^2 = min. \tag{21}$$

We transform problem (21) into a space of weighted model parameters $\mathbf{m}_w$ by replacing the variables:

$$\mathbf{m} = \hat{\mathbf{W}}(\mathbf{m})\mathbf{m}_w, \quad \hat{\mathbf{F}}_w = \hat{\mathbf{F}}\hat{\mathbf{W}}(\mathbf{m}). \tag{22}$$

Substituting equation (22) in expression (21), we find

$$P^\alpha(\mathbf{m}_w) = \|\hat{\mathbf{F}}_w\mathbf{m}_w - \mathbf{d}\|^2 + \alpha\|\mathbf{m}_w\|^2 = min. \tag{23}$$

Problem (23) seems to be completely similar to the classical minimum norm optimization problem (9) with only one important difference: the new forward modeling operator, $\hat{\mathbf{F}}_w = \hat{\mathbf{F}}\hat{\mathbf{W}}(\mathbf{m})$, depends on $\mathbf{m}_w$, so it changes in the iteration process.

We can solve problem (23) using the reweighting algorithm, where a minimization problem for $\mathbf{m}_w$ is solved in each step with fixed $\hat{\mathbf{F}}_w$ using the regularized conjugate gradient algorithm, described above. Then, $\mathbf{m}$ and $\hat{\mathbf{F}}_w$ are updated using equation (22) and $\hat{\mathbf{W}}(\mathbf{m})$ is updated using equation (20), where $\mathbf{m}$ is the inversion result in the previous step. This algorithm generates a set of equivalent solutions of the inverse problem which fit the data with the same accuracy. The different models within this set have different degrees of focusing. The model after the first iteration is actually a maximum smoothness solution. The process continues until the required degree of focusing is reached.

To conclude this section, we should note that the reweighted optimization technique has been considered in several earlier publications (Last and Kubic, 1983; Wolke and Schwetlick, 1988; O'Leary, 1990; Farquharson and Oldenburg, 1998). The most significant difficulty in the numerical implementation of this technique is related to selecting the parameter β, because for very small values of β the problem has a singularity where the individual parameters m_i are close to zero. Our approach is different in the way the weighting is introduced in the optimization process. The most significant practical advantage of our approach is that the final set of equations, (22) and (23), involves only $\hat{\mathbf{W}}(\mathbf{m})$ and not the inverse, $\hat{\mathbf{W}}^{-1}(\mathbf{m})$. In this case, according to equation (20), we can assume that $\beta = 0$ without generating any singularity:

$$\hat{\mathbf{W}}^2(m) = \mathbf{diag}[\mathbf{m}^2]\hat{\mathbf{W}}_m^{-2}. \tag{24}$$

This idea is similar to the one considered by Gorodnitsky and Rao (1997). They have also found that the reweighting equation (22) focuses the image.

Also note that our algorithm includes constraints on material properties, implemented via a penalization algorithm (Portniaguine and Zhdanov, 1999a).

Assume that the geological model can be described as a composite of two materials with known physical properties (for example, magnetic susceptibility). One material corresponds to the homogeneous background; the other characterizes the anomalous body. In this situation, the values of the material property in the inversion image can be equal to the background value or to the anomalous value. However, the geometric distribution of these values is unknown. Numerical tests show that focusing tends to produce the smallest possible anomalous domain. At the same time, the material property values m outside of this domain tend to be equal to the background values m_b. We can impose the upper bound for the positive anomalous parameter values m_a and, during the iterative process, cut off all values above this bound. This algorithm can be described as

$$\begin{aligned} m - m_b &= m_a, \quad \text{if} \quad m - m_b > m_a, \\ m - m_b &= 0, \quad \text{if} \quad m - m_b < 0. \end{aligned} \tag{25}$$

Thus, according to formula (25), the material property values m are always distributed within the interval

$$m_b < m < m_b + m_a.$$

A similar rule is applied in the case of negative anomalous parameter values.

In summary, the whole algorithm of 3-D magnetic focusing inversion with compression consists of the following steps:

1) precomputing the compressed matrix F_{mc} using formula (3),
2) calculating the sensitivity weights according to equation (17), and
3) using an iterative focusing inversion, which consist of (a) inversion of data via the conjugate gradient method according to formulas (6), (b) changing weights according to equation (24), and (c) performing penalization of material property distribution, as described above.

MODEL STUDY

We tested our method on typical models of magnetic anomalies. We considered three models similar to those discussed by Li and Oldenburg (1996): (1) a cube with anomalous magnetic susceptibility (Figure 1a), (2) a 3-D magnetic susceptibility model of a dipping slab (Figure 1b), and (3) a 3-D magnetic susceptibility model of a faulted slab (Figure 1c).

For all three models we used a coordinate system where the x-axis is directed toward geographic north, the y-axis points to geographic west, and the z-axis is directed downward. The data at the surface are measured on a 20×20 grid in the x- and y-directions, with sampling intervals of 50 m in both directions.

The model grid used in the inversion consists of cubic cells of $50 \times 50 \times 50$ m^3. In the lateral direction, it covers the area of the data grid and extends down to 500 m in the vertical direction. The number of cells in the model grid is $20 \times 20 \times 10$ (4000 cells).

Li and Oldenburg (1996) have noticed the instability of 3-D magnetic inversion to the uppermost layer of the cells. They proposed to cure that by inverting the data obtained by upward analytical continuation to a height equal to the length of the side of the cubic cell. We followed the same strategy.

The data for models 1, 2, and 3 are displayed in Figures 1d–f, respectively. These pictures represent the total field anomaly at the observation surface. However, for inversion we used the data at a height of 50 m (equal to the length of the cell side). The data were contaminated by Gaussian noise, whose standard deviation was equal to 2% of the data magnitude plus 1 nT. The strength of the inducing field for each model was 50 000 nT. The polarization of the inducing field differed from one model to another.

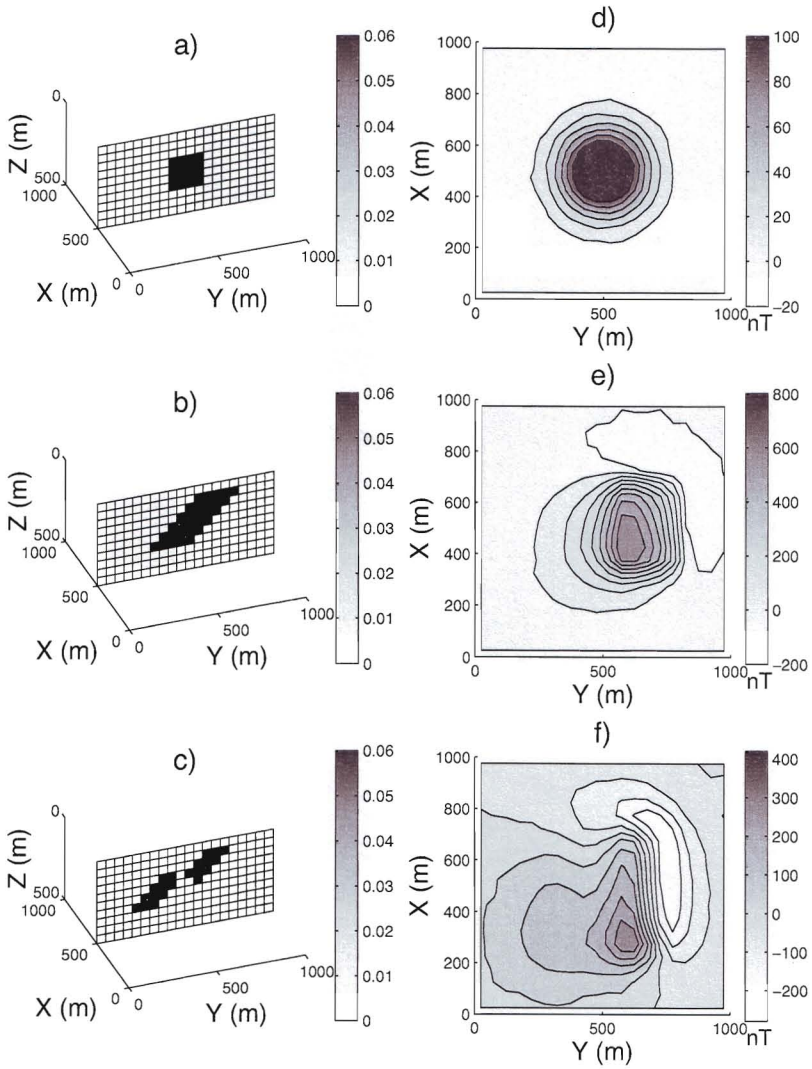

FIG. 1. (a) Model of a cube with anomalous magnetic susceptibility. (b) Model of a dipping slab. The slab strike direction points to the north, continuing from $x_1 = 250$ to $x_2 = 750$ m. (c) Model of a faulted dipping slab. The anomalous susceptibility is uniform within each slab and is equal to 0.06 SI units. (d–f) Data for cube, dipping slab, and faulted dipping slab models, respectively. Gaussian noise with a standard deviation of 2% of data magnitude plus 1 nT was added to the data.

We applied smooth inversion and focused inversion for each model. The sensitivity matrix was stored in compressed form, using the compression algorithm based on the cubic interpolation pyramid. The compression factor for all three models was 22%.

The first model is a cube with a side of 200 m. The top of the cube is buried at a depth of 150 m. Figure 1a shows the slice of the cube through the $x = 500$ m profile. The anomalous susceptibility is uniform within the cube and is equal to 0.06 SI units. The inducing field has a strength of 50 000 nT and vertical polarization (inclination $I = 90°$ and declination $D = 0°$). Figure 1d shows a map of the synthetic observed data for this model. Figure 2a presents the result of the smooth inversion, and Figure 2d demonstrates the result of the focusing inversion. The smooth inversion generates a diffused image of a cube, while the focusing inversion produces a sharp, clear image of the magnetic target. For this model the initial value of regularization parameter α was 0.3, and the final value of α was 0.0094.

The second model is a 3-D magnetic susceptibility model of a dipping slab. Figure 1b shows the slice of the slab through the $x = 500$ m profile. The slab strike direction points to the north, continuing from $x_1 = 250$ to $x_2 = 750$ m. The anomalous susceptibility is uniform within the slab and is equal to 0.06 SI units. The inducing field has the strength of 50 000 nT, $I = 75°$, and $D = 25°$. Figure 1e shows the synthetic observed data for this model. Figures 2b and 2e present the results of the smooth and focusing inversions, respectively. The smooth image provides some information about the location and inclination of the slab,

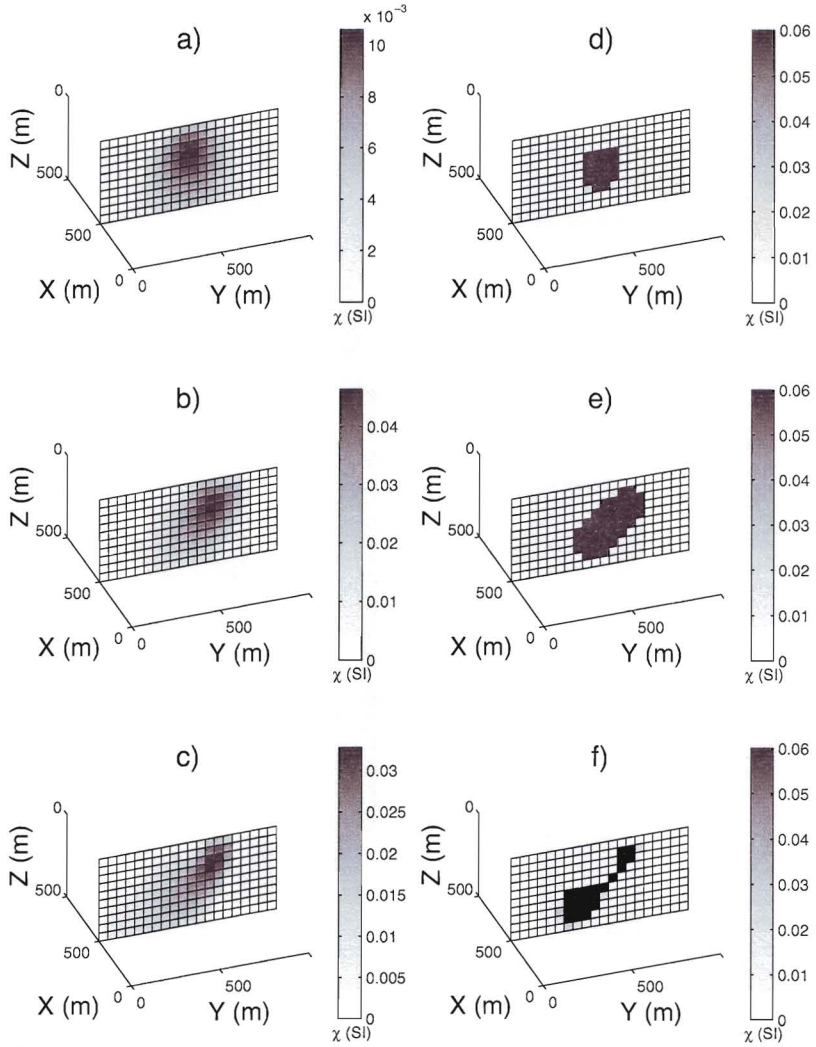

FIG. 2. Results of smooth inversion for (a) cube, (b) dipping slab, and (c) faulted dipping slab. Results of focusing inversion for (d) cube, (e) dipping slab, and (f) faulted dipping slab.

but the image is diffused and unfocused, while the focusing inversion reconstructs very well the original model of the slab.

The third model is a slab with a normal fault. Figure 1c shows the slice of the slab through the $x = 500$ m profile. The fault exists at $y = 500$ m. The inducing field has a strength of 50 000 nT, $I = 45°$, and $D = 45°$. Figure 1f shows the total field data for this model. Figures 2c and 2f present the results of the smooth and focusing inversions, respectively. The fault is vaguely visible in the smooth image, while it is clearly recognized in the sharp image.

The performance of the compression method was tested using model 1, shown in Figure 1a. On a computer with 200 MHz processor speed and 256 Mbytes of memory, we solved five problems with models of different sizes: $N_x, N_y, N_z = 20 \times 20 \times 10, 25 \times 25 \times 12, 30 \times 30 \times 15, 35 \times 35 \times 17$, and $40 \times 40 \times 20$. In each case, data dimensions were changed proportionally to the N_x and N_y model dimensions. Results of testing requirements are shown in Figure 3. Figure 3a shows timing, while Figure 3b shows memory consumption. The Dashed and solid lines show the performance of the uncompressed and compressed versions, respectively. For the uncompressed version, we used matrices with full storage memory organization to preserve efficiency. The size of the problem is referred to the number of points in x-direction, assuming that other dimensions change proportionally for the five considered cases.

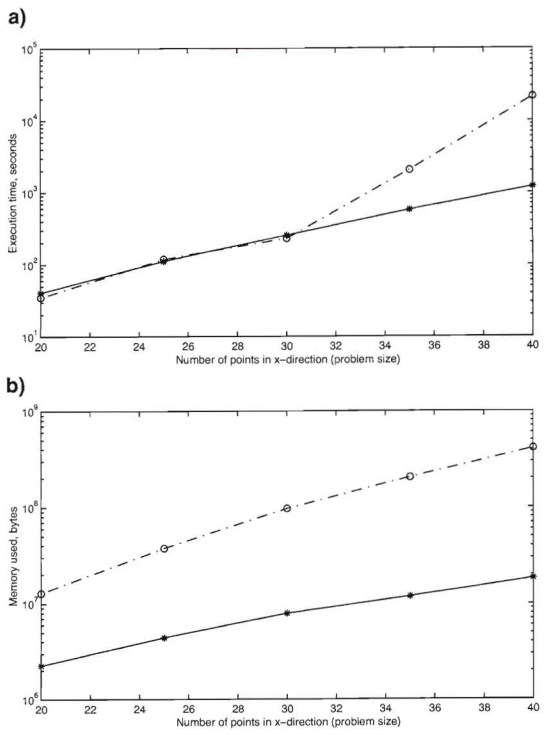

FIG. 3. (a) Increased speed and (b) memory savings because of compression. Compressed version performance is shown by the solid line. Uncompressed version performance is shown by the dashed line. Calculations are performed for test model 1, shown in Figure 1a.

For cases where the dimensions are small, the uncompressed problem has the same speed as the compressed one. That happens because the compressed problem has overhead to fill out the compressed matrix. As the dimension increases, the compressed version performs much better. For the last case, where $N_x = 40$, the uncompressed version does not fit into memory (256 Mbytes); therefore, its execution time increases dramatically.

INVERSION OF REAL DATA

We applied the developed code to interpret airborne magnetic data collected for ExxonMobil Upstream Research Company over an area in northern Canada. Figure 4a presents the map of the observed total magnetic field. The flight line spacing was about 300 m, and the flight elevation was about 100 m. The measurements were taken approximately every 16 m along the lines. In our inversion study, we assumed that the direction of the inducing magnetic field was close to vertical, since the observation area was in northern Canada. The basement (granite) is buried at a depth of about 450 m and is covered by sediments formed by till and sand layers. The goal of the interpretation was to locate the magnetization zones in the upper parts of the section, which manifest themselves as the magnetic anomalies.

In the first stage of interpretation, we divided the observed total magnetic field into regional and residual anomalies. This problem can be solved using polynomial approximation of the regional anomalies. One can use the inversion program to separate the field as well, as described below.

The lower half-space below the observation area was divided into $1 \times 1 \times 1$ km^3 cells to a depth of 20 km. Applying our 3-D inversion code, we obtained the distribution of the magnetic susceptibility within these cells. We determined the regional magnetic anomaly by applying the forward modeling code to the cells located only at depths between 4 and 20 km. The residual field was obtained by subtracting the regional part from the observed data.

In the next stage of interpretation, we divided the residual field into subregional and local anomalies. We introduced a new mesh at depths from 0 to 4 km, formed by cubic cells measuring $400 \times 400 \times 400$ m^3. The distribution of the magnetic susceptibility within this mesh was found by 3-D inversion. The subregional field was computed as the effect of the cells at depths from 1.6–4 km. This field was subtracted from the residual field to calculate the corresponding local anomalies (Figure 4b).

In the last round of the inversion, we applied the 3-D inversion code to the local anomalies only, using a mesh formed by cubic cells measuring $300 \times 300 \times 300$ m^3 located at depths from 0 to 1.5 km. In this stage we used two types of inversion: (1) the conventional maximum smoothness inversion and (2) the focusing inversion.

Figure 4c shows the result of the smooth inversion. It presents a horizontal slice of the anomalous magnetic susceptibility distribution at a depth of 800 m. The result of the focusing inversion is shown in Figure 4d.

We can clearly see the lateral shape and extent of the magnetized rock formations in these figures. However, the smooth solution produces a diffused image of the magnetic targets, while the focused solution provides a much clearer and sharper image.

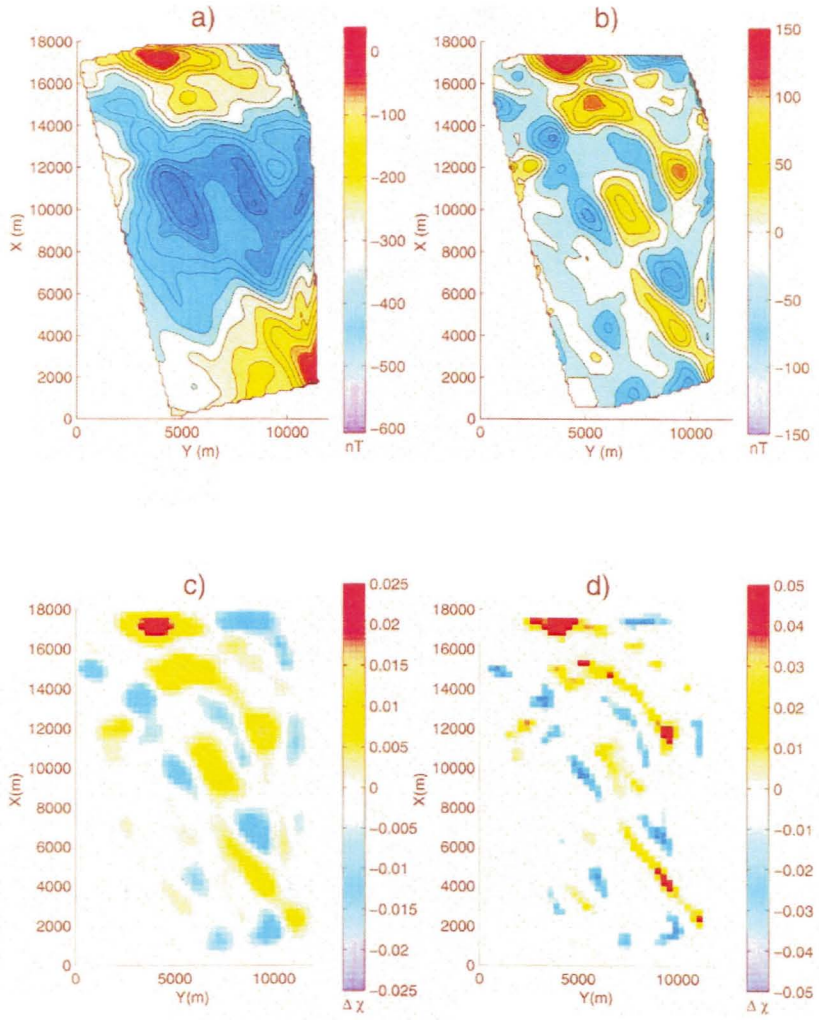

FIG. 4. (a) Airborne magnetic data. (b) Local anomalies. (c) Smooth inversion result, slice at 800 m depth. The color scale shows the anomalous susceptibility in SI units. (d) Focused inversion result, slice at 800 m depth.

ACKNOWLEDGMENTS

The authors acknowledge the support of the University of Utah Consortium for Electromagnetic Modeling and Inversion (CEMI), which includes Advanced Power Technologies Inc., AGIP, Baker Atlas Logging Services, BHP Minerals, ExxonMobil Upstream Research Company, Geological Survey of Japan, INCO Exploration, Japan National Oil Corporation, MINDECO, Naval Research Laboratory, RioTinto-Kennecott, 3JTech Corporation, and Zonge Engineering. We are thankful to ExxonMobil Upstream Research Company for providing the magnetic airborne data.

REFERENCES

Bhattacharyya, B. K., 1980, A generalized multibody model for inversion of magnetic anomalies: Geophysics, **29**, 517–531.

Farquharson, C. G., and Oldenburg, D. W., 1998, Non-linear inversion using general measures of data misfit and model structure: Geophys. J. Internat., **134**, 213–227.

Fletcher, R., 1981, Practical methods of optimization: John Wiley & Sons, Inc.

Gorodnitsky, I. F., and Rao, B. D., 1997, Sparse signal reconstruction from limited data using FOCUSS: A recursive weighted norm minimization algorithm: IEEE Trans. on Signal Proc., **45**, 600–616.

Last, B. J., and Kubik, K., 1983, Compact gravity inversion: Geophysics, **48**, 713–721.

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

Mehanee, S., Golubev, N., and Zhdanov, M. S., 1998, Weighted regularized inversion of the magnetotelluric data: 68th Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts,

O'Leary, D. P., 1990, Robust regression computation using iteratively reweighted least squares: SIAM J. Matrix Anal. Appl., **11**, 466–480.

Parker, R. L., 1994, Geophysical inverse theory: Princeton Univ. Press.

Portniaguine, O., 1999, Image focusing and data compression in the solution of geophysical inverse problems: Ph.D. dissertation, Univ. of Utah.

Portniaguine, O., and Zhdanov, M. S., 1999a, Focusing geophysical inversion images: Geophysics, **64**, 874–887.
——— 1999b, Compression in 3-D EM modeling: 2nd Internat. Symp. on 3-D Electromag., Univ. of Utah, Proceedings, 209–212.
Rao, B. D., and Babu, N. R., 1991, A rapid method for three-dimensional modeling of magnetic anomalies: Geophysics, **56**, 1729–1737.
Tikhonov, A. N., and Arsenin, V. Y., 1977, Solution of ill-posed problems: W. H. Winston & Sons, Inc.
Wolke, R., and Schwetlick, H., 1988, Iteratively reweighted least squares: Algorithms, convergence analysis, and numerical comparisons: SIAM J. Sci. Stat. Comput., **9**, 907–921.
Zhdanov, M. S., 2002, Geophysical inverse theory and regularization problems: Elsevier.

APPENDIX A
COMPRESSION IN ONE DIMENSION

To understand how to represent a full matrix as a product of sparse matrices, let us consider the compression of a full vector. The full matrix can be viewed as a collection of its columns (or rows), which are vectors.

Before we go to the complicated 3-D case, let us consider a simple 1-D vector $\mathbf{d}$. As an illustration, Figure A-1a, shows a smooth function, given as a vector of 17 values.

Let us retain the even values of $\mathbf{d}$ in vector $\mathbf{d}_e$, which has zeroes in place of the odd values. Vector $\mathbf{d}_o$ retains the odd values of $\mathbf{d}$ and has zeroes in place of the even values:

$$\mathbf{d} = \mathbf{d}_e + \mathbf{d}_o. \tag{A-1}$$

Vectors $\mathbf{d}_e$ and $\mathbf{d}_o$ are connected to $\mathbf{d}$ via diagonal matrices $\hat{\mathbf{W}}_e$ and $\hat{\mathbf{W}}_o$:

$$\mathbf{d}_o = \hat{\mathbf{W}}_o\mathbf{d}, \tag{A-2}$$

$$\mathbf{d}_e = \hat{\mathbf{W}}_e\mathbf{d}. \tag{A-3}$$

The main diagonal of $\hat{\mathbf{W}}_e$ has ones for even indices and zeroes for odd indices. The diagonal of $\hat{\mathbf{W}}_o$ has ones for odd indices and zeroes for even indices. Based on that definition, one can easily establish the following properties of $\hat{\mathbf{W}}_e$ and $\hat{\mathbf{W}}_o$:

$$\hat{\mathbf{W}}_o + \hat{\mathbf{W}}_e = \hat{\mathbf{I}}, \quad \hat{\mathbf{W}}_o\hat{\mathbf{W}}_o = \hat{\mathbf{W}}_o, \quad \hat{\mathbf{W}}_e\hat{\mathbf{W}}_e = \hat{\mathbf{W}}_e,$$
$$\hat{\mathbf{W}}_e\hat{\mathbf{W}}_o = \hat{\mathbf{W}}_o\hat{\mathbf{W}}_e = \mathbf{0}, \tag{A-4}$$

where $\hat{\mathbf{I}}$ is the identity matrix.

Consider an interpolation matrix $\hat{\mathbf{W}}_i$ which predicts values at even nodes from values at odd nodes only using cubic polynomials. The matrix $\hat{\mathbf{W}}_i$ contains coefficients of cubic polynomials. Simple calculations show that $\hat{\mathbf{W}}_i$ satisfies the equation

$$\hat{\mathbf{W}}_i = \hat{\mathbf{W}}_e\hat{\mathbf{W}}_i\hat{\mathbf{W}}_o. \tag{A-5}$$

One round of compression transformation consists of (1) predicting even node values, (2) subtracting true even values from those predicted, and (3) retaining odd node values as is. The result of this tranformation is illustrated in Figure A-1b. This transformation can be expressed in matrix notation as

$$\mathbf{d}_{c1} = \hat{\mathbf{W}}_i\mathbf{d}_o - \mathbf{d}_e + \mathbf{d}_o,$$

where $\mathbf{d}_{c1}$ is the transformed data. Taking into account equations (A-2), (A-3), and (A-5), we obtain

$$\mathbf{d}_{c1} = \hat{\mathbf{W}}_e\hat{\mathbf{W}}_i\hat{\mathbf{W}}_o\mathbf{d} - \hat{\mathbf{W}}_e\mathbf{d} + \hat{\mathbf{W}}_o\mathbf{d} = \mathbf{d}_{c1} = \hat{\mathbf{W}}_n\mathbf{d},$$

where

$$\hat{\mathbf{W}}_n = \hat{\mathbf{W}}_e\hat{\mathbf{W}}_i\hat{\mathbf{W}}_o - \hat{\mathbf{W}}_e + \hat{\mathbf{W}}_o. \tag{A-6}$$

We call $\hat{\mathbf{W}}_n$ an elementary compression matrix. Note that $\hat{\mathbf{W}}_n$ is inverse to itself because of equations (A-4) and (A-5):

$$\begin{aligned}\hat{\mathbf{W}}_n\hat{\mathbf{W}}_n &= (\hat{\mathbf{W}}_e\hat{\mathbf{W}}_i\hat{\mathbf{W}}_o - \hat{\mathbf{W}}_e + \hat{\mathbf{W}}_o)(\hat{\mathbf{W}}_e\hat{\mathbf{W}}_i\hat{\mathbf{W}}_o - \hat{\mathbf{W}}_e + \hat{\mathbf{W}}_o)\\ &= \hat{\mathbf{W}}_e\hat{\mathbf{W}}_i\hat{\mathbf{W}}_o - \hat{\mathbf{W}}_e\hat{\mathbf{W}}_i\hat{\mathbf{W}}_o + \hat{\mathbf{W}}_e + \hat{\mathbf{W}}_o\\ &= \hat{\mathbf{W}}_e + \hat{\mathbf{W}}_o = \hat{\mathbf{I}}.\end{aligned} \tag{A-7}$$

In the next round of compression transformation, we use data that is twice as coarse. Such successive transformations are called interpolation pyramids. One compression round is called an elementary compression level. The elementary compression matrices for level n are denoted above as $\hat{\mathbf{W}}_n$. For the first level, for example, it is $\hat{\mathbf{W}}_1$; for the second level it is $\hat{\mathbf{W}}_2$; etc. Figures A-1c and A-1d show the results of compression through the second and third levels.

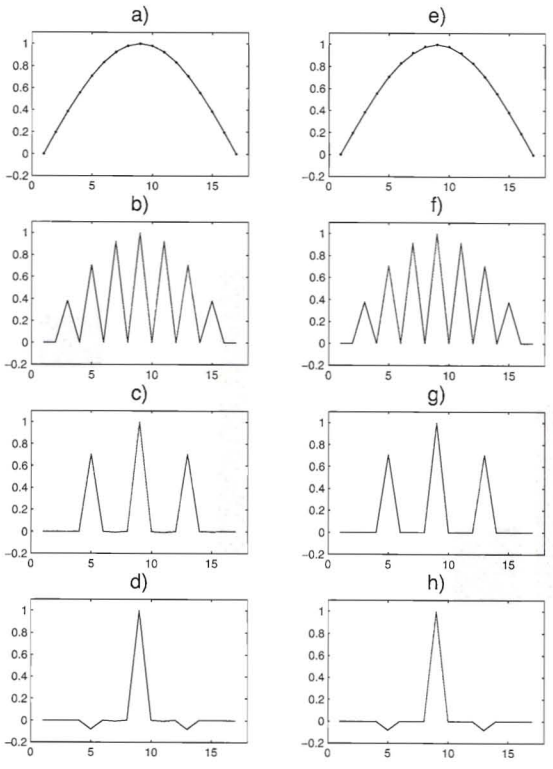

FIG. A-1. Compression with interpolation pyramid for a 1-D vector. (a) Original vector of 17 values. (b, c) Intermediate compression levels. (d) Compressed vector. (e) Restored vector, solid line; original vector, dots. (f, g) Intermediate restoration results. (h) Thresholded and sparsified vector; only three values are retained.

Combining N levels together, we arrive at the full compression transformation:

$$\mathbf{d}_c = \hat{\mathbf{W}}_N, \ldots, \hat{\mathbf{W}}_2\hat{\mathbf{W}}_1\mathbf{d} = \hat{\mathbf{W}}_c\mathbf{d}, \tag{A-8}$$

where $\hat{\mathbf{W}}_c$ is a compression matrix,

$$\hat{\mathbf{W}}_c = \hat{\mathbf{W}}_N, \ldots, \hat{\mathbf{W}}_2\hat{\mathbf{W}}_1. \tag{A-9}$$

Figure A-1a shows the original vector $\mathbf{d}$, a smooth function of 17 values. The result of the compression transformation is shown in Figure A-1d. Note that cubic interpolation predicts the intermediate values of the smooth function very well, and the compressed vector $\mathbf{d}_c$ contains the differences between such predictions and the actual values. Therefore, only a few values in $\mathbf{d}_c$ are significant, and the rest are close to zero. It is therefore possible to store $\mathbf{d}_c$ as sparse, using threshold transformation

$$\mathbf{d}_c = \mathit{threshold}(\hat{\mathbf{W}}_c\mathbf{d}, \epsilon). \tag{A-10}$$

The inverse operation, restoration, is described by the same matrices $\hat{\mathbf{W}}_n$ applied in the reverse [order from property (A-7)]:

$$\mathbf{d} = \hat{\mathbf{W}}_1, \ldots, \hat{\mathbf{W}}_{N-1}\hat{\mathbf{W}}_N\mathbf{d}_c = \hat{\mathbf{W}}_r\mathbf{d}_c, \tag{A-11}$$

where $\hat{\mathbf{W}}_r$ is a restoration matrix:

$$\hat{\mathbf{W}}_r = \hat{\mathbf{W}}_1, \ldots, \hat{\mathbf{W}}_{N-1}\hat{\mathbf{W}}_N. \tag{A-12}$$

Figure A-1h shows vector $\mathbf{d}_c$ thresholded at 1% of its maximum, which contains only three nonzero values and therefore is sparse. Figures A-1f and A-1g illustrate the restoration process. Figure A-1e shows the restored vector as a solid line; the original vector is shown by dots.

APPENDIX B

FACTORIZATION OF MATRICES FOR 3-D COMPRESSION

When solving 3-D magnetic inverse problems, we have to handle model parameters and data in three dimensions. In this section we discuss how the basic principles of 1-D compression can be generalized to the 3-D case.

Consider, for example, a two-level interpolation pyramid applied to a 3-D function depending on three Cartesian coordinates (x, y, z). The compression matrix $\hat{\mathbf{W}}_c$ is the product of six elementary compression matrices:

$$\hat{\mathbf{W}}_c = \hat{\mathbf{W}}_{z2}\hat{\mathbf{W}}_{y2}\hat{\mathbf{W}}_{x2}\hat{\mathbf{W}}_{z1}\hat{\mathbf{W}}_{y1}\hat{\mathbf{W}}_{x1}, \tag{B-1}$$

where the indices x, y, z denote the axis along which a particular matrix is applied and the numerical indices 1, 2 denote the pyramid level.

In the case of 1-D linear compression, we interpolate a function using a two-point scheme. The first-level matrix $\hat{\mathbf{W}}_{x1}$ has two nonzero off-diagonal elements. The matrix $\hat{\mathbf{W}}_c$ turns into a 1-D compression matrix in the x-direction if

$$\hat{\mathbf{W}}_{z1} = \hat{\mathbf{W}}_{y1} = \hat{\mathbf{W}}_{z2} = \hat{\mathbf{W}}_{y2} = \hat{\mathbf{I}}.$$

In 1-D finite-difference cubic interpolation, for example, the scheme is four point and $\hat{\mathbf{W}}_{x1}$ has four off-diagonal elements. This decreases the sparsity of $\hat{\mathbf{W}}_c$.

A 2-D compression matrix over the x- and y-directions is obtained if $\hat{\mathbf{W}}_{z1} = \hat{\mathbf{W}}_{z2} = \hat{\mathbf{I}}$. The compression matrix at the first pyramid level is equal to $\hat{\mathbf{W}}_{y1}\hat{\mathbf{W}}_{x1}$. In 2-D bilinear interpolation, the scheme is four point; in 2-D finite-difference cubic interpolation, the scheme is 16 point.

For 3-D interpolation, the compression matrix at the first pyramid level is a product of all three elementary matrices over the x-, y-, and z-directions:

$$\hat{\mathbf{W}}_c = \hat{\mathbf{W}}_{z1}\hat{\mathbf{W}}_{y1}\hat{\mathbf{W}}_{x1}. \tag{B-2}$$

The interpolation scheme is eight point for trilinear interpolation and 64 point for tricubic interpolation.

The compression matrices tend to be less and less sparse with growth of the dimension and in the complexity of the interpolating function. This effect can be countered by storing $\hat{\mathbf{W}}_c$ as a factorization of elementary compression matrices, as in equation (B-1), without computing their product.

Further, we notice that the structure of the elementary compression matrices is such that at higher pyramid levels only a few points are reduced. The other points are passed without a change, being already reduced on lower levels. For example, a volume of $64 \times 64 \times 64$ points has six pyramid levels, and there are three elementary matrices in the x-, y-, and z-directions at each level correspondingly. Therefore, $\hat{\mathbf{W}}_c$ will be stored as a product of 18 matrices. For the last several levels, these matrices contain few off-diagonal elements (because the last reduction levels are coarse). On the main diagonal, the elements mostly equal 1. We may therefore further reduce the amount of storage by keeping the elementary matrices with the main diagonal subtracted:

$$\begin{aligned}
\hat{\mathbf{W}}_1 &= \hat{\mathbf{W}}_{x1} - \hat{\mathbf{I}},\\
\hat{\mathbf{W}}_2 &= \hat{\mathbf{W}}_{y1} - \hat{\mathbf{I}},\\
\hat{\mathbf{W}}_3 &= \hat{\mathbf{W}}_{z1} - \hat{\mathbf{I}},\\
\hat{\mathbf{W}}_4 &= \hat{\mathbf{W}}_{x2} - \hat{\mathbf{I}},\\
\hat{\mathbf{W}}_5 &= \hat{\mathbf{W}}_{y2} - \hat{\mathbf{I}},\\
\hat{\mathbf{W}}_6 &= \hat{\mathbf{W}}_{z2} - \hat{\mathbf{I}}.
\end{aligned} \tag{B-3}$$

Storing matrices $\hat{\mathbf{W}}_1, \hat{\mathbf{W}}_2$, etc., requires less storage than storing $\hat{\mathbf{W}}_{x1}, \hat{\mathbf{W}}_{y1}$, etc.

Now the compression procedure of a vector $\hat{\mathbf{d}}$ can be described by the recursive formula

$$\hat{\mathbf{d}}_{n+1} = \hat{\mathbf{W}}_n\hat{\mathbf{d}}_n + \hat{\mathbf{d}}_n, \tag{B-4}$$

where n changes from 1 to a number of elementary matrices in the factorization. The restoration is described by formula (B-4) applied in the reverse order:

$$\hat{\mathbf{d}}_n = \hat{\mathbf{W}}_n\hat{\mathbf{d}}_{n+1} + \hat{\mathbf{d}}_{n+1}, \tag{B-5}$$

where n changes from the number of elementary matrices to 1. The use of formulas (B-4) and (B-5) saves space and execution time because the vector under transformation is not multiplied by $\hat{\mathbf{I}}$, which would have been the case if we had used matrices $\hat{\mathbf{W}}_{x1}, \hat{\mathbf{W}}_{y1}$, etc., directly.