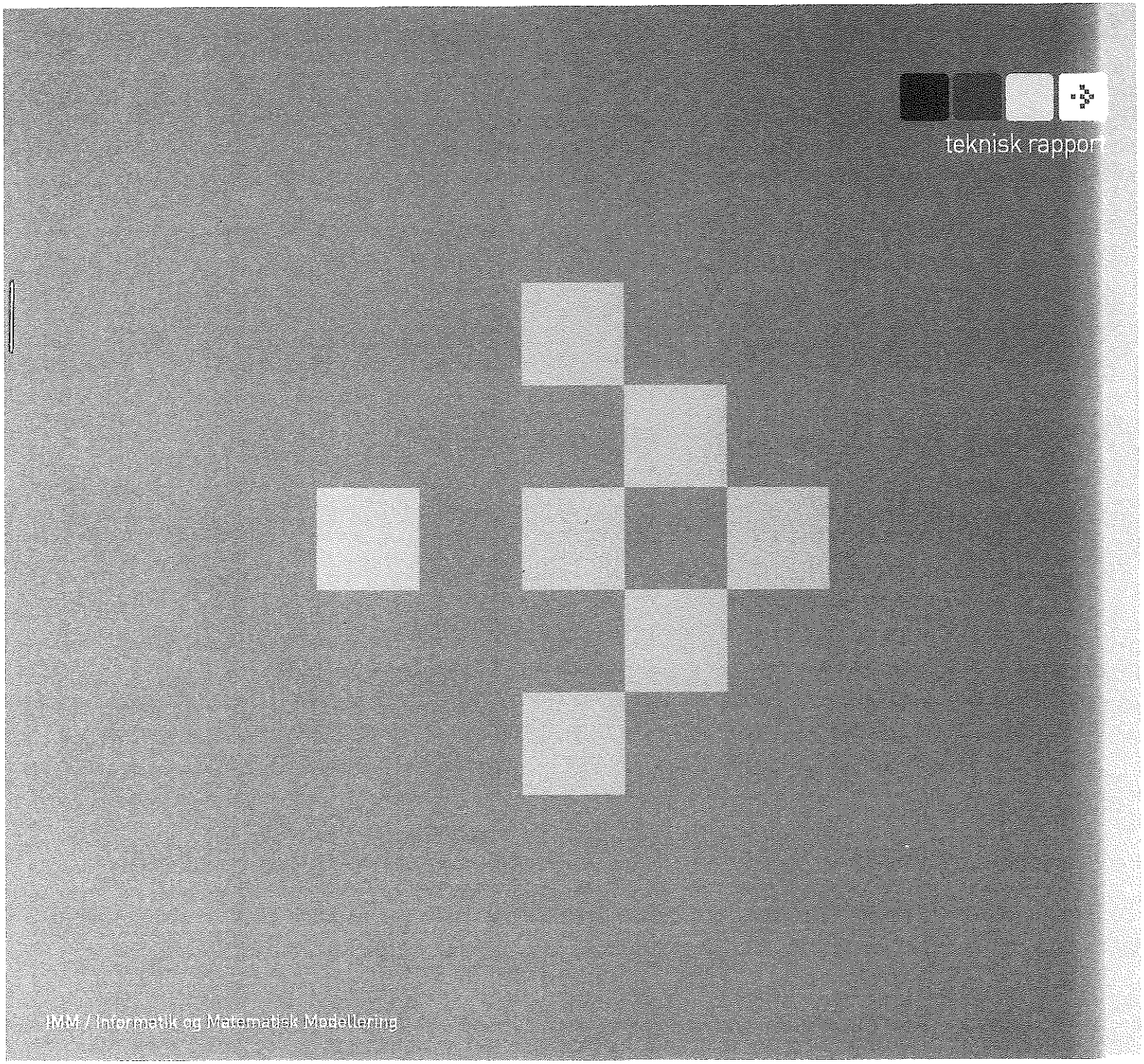

Computational Properties of Algorithms for Multi-Level Inversion of 3-D Potential Fields

M. L. Andersen, M. Fedi, P. C. Hansen, V. Paolitti and A. Rapolla

Kgs. Lyngby 2002
IMM-TECHNICAL REPORT-2002-16

DTU

Computational Properties of Algorithms for Multi-Level Inversion of 3-D Potential Fields

M. L. Andersen[1], M. Fedi[2], P. C. Hansen[1], V. Paoletti[2], A. Rapolla[2]

[1]Informatics and Mathematical Modelling,
Technical University of Denmark,
Building 321, DK-2800 Lyngby, Denmark

[2]Earth Sciences Department, Naples University,
Largo s. Marcellino 10, I-80138 Napoli, Italy

September 29, 2002

Abstract

The goal of this paper is to study various issues in the numerical treatment of inverse potential-field problems in geophysics, focusing on a formulation of the problem that uses multiple levels of data. We use the singular value decomposition to study issues related to the interspacing of the data levels as well as the discretization parameters, and we investigate the use of a new preconditioned iterative solver.

1 Introduction

It is well known that potential field interpretation is characterized by an ambiguity in determination of the source from field data. This non-uniqueness is related to the fact that observations are finite in number and therefore the knowledge of the source function is insufficient. Moreover, approximation errors as well as measurement errors always put a restriction on the accuracy of the computed reconstruction. Still the main difficulty in potential field interpretation is connected to the inherent ambiguity. By Gauss' theorem, any magnetic field measured on the surface can be reproduced by an infinitesimally thin zone of magnetic dipoles immediately beneath the surface. This means a lack of depth resolution in potential-field data inversion.

Over the years the fundamental problem of obtaining depth resolution in both gravitational and magnetic inverse problems has been considered and discussed by many researchers, e.g., Li and Oldenburg [13] and Fedi and Rapolla [5]. One way to achieve depth resolution is to incorporate prior knowledge of the unknown source function. The prior information can be supplied, e.g., by searching for a source whose weighted norm is minimized with respect to some reference model [7], or by solving a constrained linear least-squares problem with upper and lower bounds for the solution and the requirement that the source function increases monotonically

with depth [6]. Other methods seek to avoid dispersion of the source by imposing the condition that the volume of the source body is minimized [12], by constraining the source to have a minimum momentum of inertia with respect to the center of gravity or to an axis of a given dip passing through it [8], or by allowing compactness along several axes given prior information about the axis length [4].

Yet another possibility is to use a Tikhonov formulation, i.e., to minimize a global objective function composed of the model objective function and the data misfit. The prior information is in this case incorporated into the model objective function, e.g., by using one or more appropriate weight functions as done by Li and Oldenburg [13]. In their algorithm the depth resolution is strongly dependent on the chosen weight functions.

More recently, Fedi and Rapolla [5] proposed an alternative Tikhonov formulation that avoids the use of weighting functions. Instead, the authors demonstrated that knowledge of the field in multiple data levels provides the desired depth resolution. When the additional data levels are introduced the discretized system of equations becomes larger, and the good depth resolution is obtained from intrinsic properties of this matrix as discussed by Andersen et al. [1].

The aim of this paper is to study the numerical treatment of the multi-level approach, and to show the importance that the characteristics of the discretization has on the performance of the inversion algorithm. Although the importance of these features is well recognized, we feel that this issue has not been deeply explored yet. In this work we will focus on a discretization based on a division of the source volume into a 3-D grid of rectangular prisms. Then we will use the singular value decomposition to study the influence of both the geometry of the prisms and the interspacing of the data levels on the performance of the reconstruction of the source.

We also study the use of a new preconditioned iterative method for solving the linear system of equations involved in the multi-level approach. The new algorithm incorporates a preconditioner based on a low-dimensional subspace chosen by the user, and we discuss the choice of this subspace for our class of problems. The purpose of the preconditioner is to speed up the convergence of the iterative algorithm, and we demonstrate that we are able to achieve a substantial reduction in computing time compared to existing iterative solvers, without any deterioration of the reconstructed source.

Our paper is organized as follows. In section 2 we introduce the mathematical formulation of the problem and we describe the numerical method which is based on Tikhonov regularization. In section 3 we perform an analysis of the properties of the linear systems of equations, with emphasis on the influence of the thickness of the prisms and the interspacing of the data levels. Finally, in section 4 we study the use of the new iterative algorithm in connection with several choices of the Tikhonov smoothing norm.

2 Formulation and Discretization of the Inverse Geomagnetic Problem

Inverse potential field problems are described by first-kind Fredholm integral equations which, by nature, are ill-posed problems. In this work we consider the inverse

geomagnetic problem of the form

$$\int_{\Omega} K(\mathbf{r}, \mathbf{r}_0)\ M(\mathbf{r}_0)\ d^3\mathbf{r}_0 = \Delta T(\mathbf{r}), \tag{1}$$

where Ω is our rectangular source volume, $\Delta T(\mathbf{r})$ is the measured magnetic field at position $\mathbf{r}$, and the source function $M(\mathbf{r}_0)$ is the unknown distribution of magnetization. The kernel K of the integral equation is given by

$$K(\mathbf{r}, \mathbf{r}_0) = \frac{\partial^2}{\partial\alpha\ \partial\kappa}\ \frac{1}{\|\mathbf{r} - \mathbf{r}_0\|_2}, \tag{2}$$

where the differentiation is with respect to the orientations α and κ of the magnetization of the source and the induced field. We note that in the special case where these directions are both vertical, K takes the special form

$$K(\mathbf{r}, \mathbf{r}_0) = \frac{3(z - z_0)^2}{\|\mathbf{r} - \mathbf{r}_0\|_2^5} - \frac{1}{\|\mathbf{r} - \mathbf{r}_0\|_2^3}, \tag{3}$$

where z and z_0 are the vertical components of $\mathbf{r}$ and $\mathbf{r}_0$, respectively.

The problem (1) is discretized as follows. First we subdivide Ω into prisms arranged in an $N_x \times N_y \times N_z$ grid, and then we use an approximate source function $\widetilde{M}(\mathbf{r}_0)$ that is piecewise constant within each prism. If $\widetilde{M}_j$ denotes the constant value of $\widetilde{M}(\mathbf{r}_0)$ in prism number j, and if P denotes the number of data points in each of the L levels, then we obtain an $LP \times N$ least squares system of linear equations (with $N = N_x\, N_y\, N_z$):

$$\min\ \|\mathbf{K}\,\mathbf{m} - \mathbf{d}\|_2, \tag{4}$$

whose coefficient matrix $\mathbf{K}$ has elements given by

$$K_{ij} = \int_{\Omega_j} K(\mathbf{r}_i, \mathbf{r}_0)\, d^3\mathbf{r}_0, \tag{5}$$

while the data vector $\mathbf{d}$ contains the measurements $d_i = \Delta T(\mathbf{r}_i)$, and the solution vector $\mathbf{m}$ contains the unknown values $\widetilde{M}_j$. The matrix elements K_{ij} are computed by means of Eq. (9.19) in [3].

In order to analyze the discretized problem (4) we use the singular value decomposition (SVD) of the coefficient matrix:

$$\mathbf{K} = \sum_{i=1}^{q} \mathbf{u}_i\, \sigma_i\, \mathbf{v}_i^T, \tag{6}$$

where $q = \min(N, LP)$. Here $\mathbf{u}_i$ and $\mathbf{v}_i$ are the left and right singular vectors, and σ_i are the singular values which are nonnegative and appear in decreasing order.

Throughout the paper we will solely use Tikhonov regularization for the stabilization of the solution, and we recall that in the discretized formulation the Tikhonov solution $\mathbf{m}_\lambda$ solves the regularized problem

$$\min \left\{ \|\mathbf{K}\,\mathbf{m} - \mathbf{d}\|_2^2 + \lambda^2 \|\mathbf{D}\,\mathbf{m}\|_2^2 \right\}, \tag{7}$$

in which λ is a regularization parameter chosen by the user, and $\mathbf{D}$ is a matrix that defines an appropriate smoothing norm suited for the particular problem. This matrix is often either the identity matrix, $\mathbf{D} = \mathbf{I}$, or an approximation to a derivative operator. The case $\mathbf{D} = \mathbf{I}$ represents the smoothing functional $\|M\|_2^2 = \int_\Omega M(\mathbf{r}_0)^2 \, d^3\mathbf{r}_0$, and it is well known that in this case the Tikhonov solution can be written in terms of the SVD as

$$\mathbf{m}_\lambda = \sum_{i=1}^{q} \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2} \frac{\mathbf{u}_i^T \mathbf{d}}{\sigma_i} \mathbf{v}_i \tag{8}$$

where the role of the "filter factors" $\sigma_i^2/(\sigma_i^2 + \lambda^2)$ is to dampen the contributions to $\mathbf{m}_\lambda$ corresponding to the smallest singular values. Note that this damping effectively sets in for singular values σ_i slightly smaller than λ. In this way λ can be used to control the filtering/smoothing of the solution.

The choice of λ depends on the amount of noise in the data $\mathbf{d}$. If the noise in each data element $d_i = \Delta T(\mathbf{r}_i)$ is additive and has standard deviation σ_{noise} then right-hand side coefficients with $|\mathbf{u}_i^T \mathbf{d}| > \sigma_{\text{noise}}$ are dominated by the pure field, while coefficients with $|\mathbf{u}_i^T \mathbf{d}| < \sigma_{\text{noise}}$ are dominated by the noise. Hence λ should be chosen in such a way that the Tikhonov filter factors $\sigma_i^2/(\sigma_i^2 + \lambda^2)$ dampen – or filter out – the coefficients smaller than σ_{noise} while maintaining the large components. The large components to be maintained are, in turn, associated with the largest singular values because, on the average, the coefficients $|\mathbf{u}_i^T \mathbf{d}|$ always decay – cf., e.g., [10].

Consequently, with a small amount of noise many SVD components can be included in the regularized solution and therefore a small λ can be chosen. As the noise level increases, less SVD components can be used and λ must be increased. And since depth resolution improves as the number of included SVD components increases, cf. [1], with a fixed noise level it is desirable to obtain a discretized system in which as many SVD components as possible can be included in the regularized solution $\mathbf{m}_\lambda$. The main goal of this manuscript is to investigate this issue.

The analysis by Andersen et al. [1] shows that the depth resolution can sometimes be improved if the matrix $\mathbf{D}$ is chosen to be different from the identity matrix, such as a matrix that involves spatial derivatives of the source function $M(\mathbf{r}_0)$. In this study, in addition to $\mathbf{D} = \mathbf{I}$, we use the matrix $\mathbf{D}_{xyz}$ that represents the smoothing functional

$$\left\| \frac{\partial^2 M}{\partial x_0^2} \right\|_2^2 + \left\| \frac{\partial^2 M}{\partial y_0^2} \right\|_2^2 + \left\| \frac{\partial^2 M}{\partial z_0^2} \right\|_2^2 =$$

$$\int_\Omega \left(\frac{\partial^2 M(\mathbf{r}_0)}{\partial x_0^2} \right)^2 d^3\mathbf{r}_0 + \int_\Omega \left(\frac{\partial^2 M(\mathbf{r}_0)}{\partial y_0^2} \right)^2 d^3\mathbf{r}_0 + \int_\Omega \left(\frac{\partial^2 M(\mathbf{r}_0)}{\partial z_0^2} \right)^2 d^3\mathbf{r}_0 \tag{9}$$

and the matrix $\mathbf{D}_z$ representing the smoothing functional

$$\|M\|_2^2 + \left\| \frac{\partial^2 M}{\partial z_0^2} \right\|_2^2 = \int_\Omega M(\mathbf{r}_0)^2 d^3\mathbf{r}_0 + \int_\Omega \left(\frac{\partial^2 M(\mathbf{r}_0)}{\partial z_0^2} \right)^2 d^3\mathbf{r}_0. \tag{10}$$

The second main goal of our paper is to investigate how the Tikhonov problem (7) with these different $\mathbf{D}$-matrices can be solved by means of a new iterative method, which is of potential interest for future work on large-scale problems.

3 Influence of Discretization Parameters

The purpose of this section is to study how the reconstruction capabilities of the multi-level approach are influenced by the various discretization parameters. To this end we use a synthetic problem. We keep the physical size of the source volume Ω fixed as a cube of dimensions $6000 \times 6000 \times 3000$ meters, and we use a grid of $N_x \times N_y \times N_z$ prisms with $N_x = N_y = 12$, while N_z varies. Thus the length and width of each prism are both 500 meters, while the prism thickness is given by

$$t = 3000/N_z. \tag{11}$$

Throughout we use $L = 5$ data levels, and the data points in each level lie on a grid consisting of $N_x \times N_y$ points, covering an area of 6000×6000 meters.

The exact source $M(\mathbf{r}_0)$ used in our study consists of a cube of size $1500 \times 1500 \times 600$ meters with constant magnetization, located at depth 600–1200 meters and covering prisms 5–7 in the x and y directions. The magnetization in the remaining prisms is zero.

A word about the number L of data levels is in place here. Previous work [5], [1] has shown that more than one level of data is necessary in order to achieve depth resolution. Recall that the maximum number of SVD components that can be included in the regularized solution is the smallest dimension of the coefficient matrix $\mathbf{K}$, and this number is $q = \min(N, LP) = N_x N_y \min(L, N_z)$. Also recall that with noise in the data, the effective number of SVD components that should be included in the regularized solution is determined by the noise level; the smaller the noise level the more SVD components can be included. Extensive numerical experiments (not included in this paper) have shown that with any realistic noise level in the data, usually we cannot hope to include more than about $5\, N_x N_y$ SVD components in the regularized solution. Hence we have chosen to use $L = 5$ in this study.

3.1 Prism Thickness and Data-Level Interspacing

One of the important parameters to choose in our solution procedure is the distance h_0 from the top of the solution volume Ω to the lowest level of data points. As h_0 increases, the data contained in this level becomes more smooth and thus holds less information about details in the solution $M(\mathbf{r}_0)$. Therefore it is obvious that a small value of h_0 is preferable, as long as several levels of data are used. In the present study we use $h_0 = 10$ meters.

Another important parameter is the interspacing h between the data levels. The physics of the problems dictates that there must be an optimal value of h. When h is very large the highest data levels correspond to a field with very small magnitude, and due to the smoothing process of the integration with the kernel in (1) these fields are all very smooth; for both reasons the information content in the additional levels is small. Thus, there must be an optimal range of values for h for which the collected data contains a maximum amount of information. Figure 1 illustrates this point; when h is either very small or very large then the singular values decay faster than they do for an intermediate value of h. However, a more thorough analysis is needed in order to determine the optimal range of h; we return to this shortly.

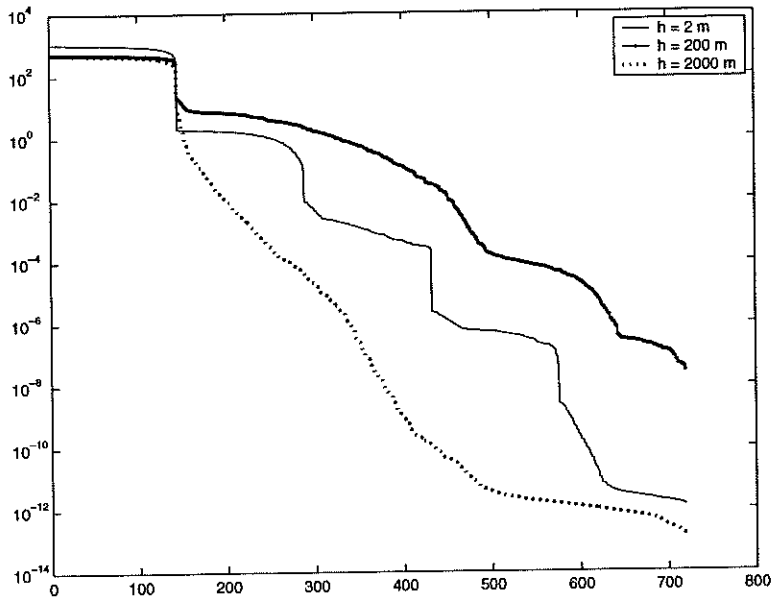

Figure 1: Singular values σ_i for three values of the data-level interspacing: $h = 2$, 200, and 2000 meters.

Yet another important parameter is, of course, the thickness t of the prisms. Again we do not expect to determine a single optimal value, but rather a range of suitable choices.

Our goal is now to determine suitable ranges for h and t, with t given by (11), which is done by numerical experiments. We performed an SVD analysis for all nine combinations of the following parameter values (in meters):

$$h = 600, \quad 200, \quad 66.67; \qquad t = 600, \quad 200, \quad 66.67. \tag{12}$$

The values of N_z that correspond to the above values of t are thus

$$N_z = 5, \quad 15, \quad 45. \tag{13}$$

We remark that $N_z = 5$ leads to a square matrix, while the other two values lead to underdetermined problems.

The results of our SVD analysis are shown in the nine plots in Fig. 2. Each plot shows the first 500 singular values σ_i and the absolute values $|\mathbf{u}_i^T \mathbf{d}|$ of the right-hand side coefficients (recall that the solution coefficients in the Tikhonov solution are $\sigma_i^2/(\sigma_i^2 + \lambda^2)\ \mathbf{u}_i^T \mathbf{d}/\sigma_i$). There is no need to show singular values or coefficients for larger indices, because the right-hand side coefficients will be below any realistic noise level.

In all nine plots we see a distinct plateau in the singular value spectrum in the index range from $i = 1$ to $i = N_x N_y = 144$. The SVD components in this range represent the shallow components of the reconstructed solution, cf. the analysis in [1]. When most of the coefficients $|\mathbf{u}_i^T \mathbf{d}|$ in this range are small, we can conclude that the solution does not have a shallow component; this is actually the case in our study where the source is located at some depth.

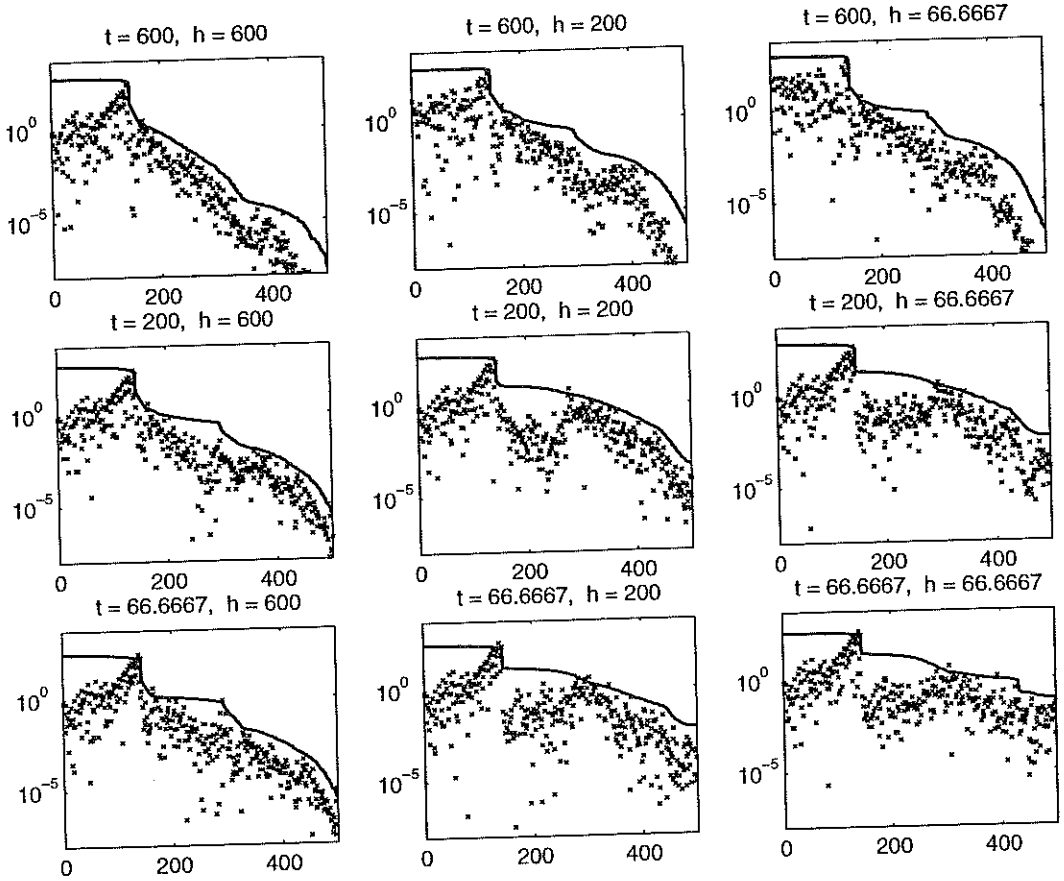

Figure 2: Singular values σ_i (solid line) and right-hand side coefficients $|\mathbf{u}_i^T \mathbf{d}|$ (crosses) for the nine combinations of the data-level interspacing: $h = 600$, 200, and 66.67 meters and the prism thickness $t = 600$, 200 and 66.67 meters.

For indices i beyond 144 the right-hand side coefficients always decay, on the average. This is a consequence of the fact that the discrete Picard condition must always be satisfied for a discretization of an ill-posed problem, cf. [10]. Throughout, when we talk about decays we always consider the average behavior for $i > 144$.

When both h and t are large (top left plot), we see a rather fast decay of both the singular values and the right-hand side coefficients, indicating that the discretized problem is severely ill-posed and only a fairly small number of SVD components can be recovered in the regularized solution when noise is present.

Now we consider the behavior when the interspacing h is fixed and t decreases (and thus N_z increases), i.e., we study the changes in the plots going from top to bottom. Then we see a gradual change toward slower decays of both the singular values and the right-hand side coefficients. The change in decay rate is more pronounced when t changes from 600 to 200, and much less pronounced when t changes from 200 to 66.67. A slower decay is preferable because it allows the recovery of more SVD components, but it should be emphasized that the "cost" is a larger matrix and thus more computations. Hence, in this study we prefer to use $t \approx 200$ which yields a good trade-off between matrix size and depth resolution (as measured by the number of recoverable SVD components). In general, we advocate to use a prism thickness which is about half of (or less than) the prism dimensions along the x and y directions.

Next we consider the behavior when the prism thickness t is fixed and h decreases, i.e., we study the changes in the plots going from left to right. Again we see a change towards slower decays of both singular values and right-hand side coefficients, the change in decay being more pronounced when going from $h = 600$ to 200 than when going from 200 to 66.67. Since there is no change in the matrix size when h varies, there is no clear optimal choice of h to be deduced from these plots with respect to computational work in solving the problem. More experiments are performed below to decide on the optimal range of h.

3.2 Synthetic Noisy Data

The above analysis showed that for the particular problem studied here, a prism thickness of $t = 250$ is a good choice; but no definite conclusion could be made about the optimal choice of h. We know that h must not be too small; the question remains what is "too small," and the purpose of this section is to study this aspect in more detail, again using the synthetic model but now with noise added to the data.

To be specific, the perturbed data is given by

$$\tilde{\mathbf{d}} = \mathbf{d} + \mathbf{e} \tag{14}$$

where the vector $\mathbf{d}$ is the "pure data" from the exact source, and the vector $\mathbf{e}$ is a random vector with elements from a Gaussian distribution with zero mean and standard deviation $\sigma_{\text{noise}} \approx 10^{-3}$. Consequently we expect that the perturbed right-hand side coefficients $|\mathbf{u}_i^T \tilde{\mathbf{d}}|$ will level off at a plateau around σ_{noise}. Only the right-hand side coefficients above this noise level should be included in the regularized solution.

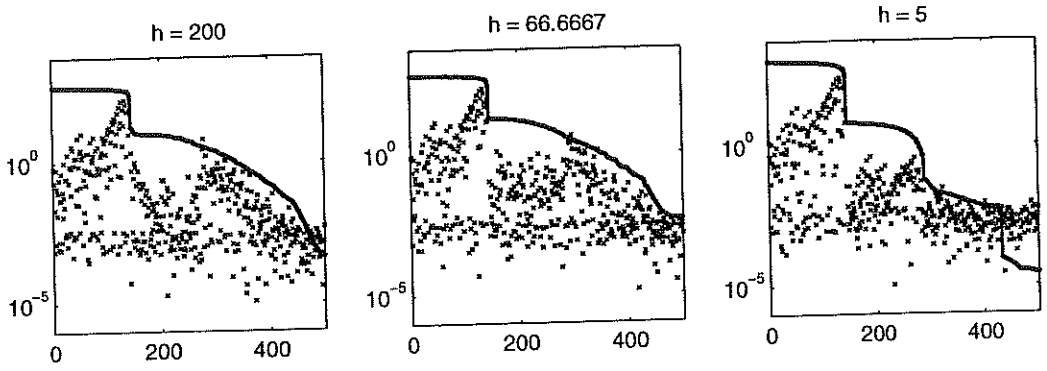

Figure 3: Singular values σ_i (solid line) and right-hand side coefficients $|\mathbf{u}_i^T\mathbf{d}|$ (crosses) for three values of the data-level interspacing: $h = 200$, 66.67 and 5 meters.

Figure 3 shows the singular values σ_i and right-hand side coefficients$|\mathbf{u}_i^T\mathbf{d}|$ for the following three values of the data-level interspacing (in meters):

$$h = 200, \quad 66.67, \quad 5.$$

The first two values are identical to those used in the previous experiments, while the third was chosen to illustrate the behavior of the discretized problem when h is too small. For the smallest value of h we see a faster decay of the singular values, caused by the fact that information carried in the five levels is almost redundant.

In all three plots in Fig. 3 we see that the right-hand side coefficients indeed level off at the noise level $\sigma_{\text{noise}} \approx 10^{-3}$. For $h = 200$ this means that the first 450 SVD components can be included in the regularized solution, and consequently we should choose $\lambda \simeq 2 \cdot 10^{-2}$. Similarly, for $h = 66.67$ and $h = 5$ we can include, respectively, 440 and 300 SVD components in the regularized solution, and therefore we choose $\lambda \simeq 3 \cdot 10^{-2}$ and $\lambda \simeq 3 \cdot 10^{-2}$, respectively. In this way, we have chosen λ in all three cases such that as many SVD components are included, with respect to the given noise level.

The three regularized solutions computed by these values of λ are shown in Fig. 4, together with the exact solution. For clarity, we show a vertical slice through the center of the source volume Ω. For $h = 200$ and $h = 66.67$ we compute almost the same regularized solution, in which the depth localization is reasonably good (with the usual "shadow" beneath the reconstructed source). However, for the small value $h = 5$ the reconstruction is unsatisfactory. The figure clearly illustrates the observation made above that there is no obvious choice of an optimal value of h, as long as it is not too small. When h is too small, the reconstruction deteriorates. From Fig. ?? we also know that h should not be too large, either.

Our conclusion is that with the discretization parameters chosen here, we choose to use h of the order 100, which yields discretized problems that are not too ill-posed and therefore provide the desired depth resolution. In general, as for the prims thickness, we suggest to use a level interspacing h which is about half of (or less than) the prism dimensions along the x and y directions.

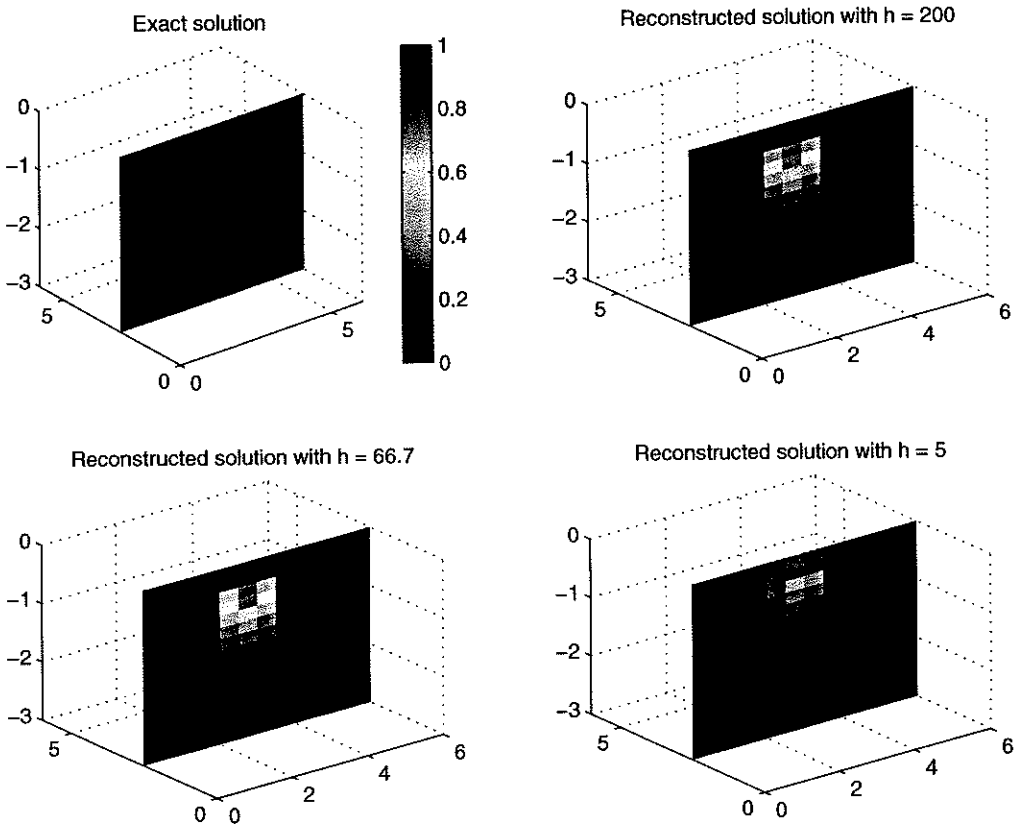

Figure 4: Exact solution (top left) and three regularized solutions for $h = 200$, 66.67 and 5 meters. In each case, the regularization parameter was chosen to include as many SVD components as possible, with respect to the noise level.

4 Solving the Linear System by an Iterative Algorithm

The purpose of this section is to study the use of a state-of-the-art iterative algorithm for solving the general Tikhonov problem (7), in connection with the three D-matrices mentioned in section 2.

4.1 Introduction to SP-LSQR

Large linear systems of equations are often solved by means of iterative methods such as the conjugate gradient (CG) method. The Tikhonov problem with a general smoothing norm is equivalent to a least squares problem of the form

$$\min \left\| \begin{pmatrix} \mathbf{K} \\ \lambda \mathbf{D} \end{pmatrix} \mathbf{m} - \begin{pmatrix} \mathbf{d} \\ \mathbf{0} \end{pmatrix} \right\|_2 \tag{15}$$

and this problem should be solved by specialized versions of CG, such as CGLS or LSQR. In this study, we use LSQR which is known to be more accurate than CGLS for ill-conditioned problems. We refer to [2] for theoretical and algorithmic details; suffice it here to say that LSQR (and CGLS) involve multiplications with the coefficient matrix in (15) and its transpose. Any sparsity or structure of the matrices should be utilized in these multiplications.

The key ingredient in our new preconditioned version of LSQR involves an matrix $\mathbf{W}$ whose k columns should preferably be chosen to approximate the k principal (i.e., the smoothest) right singular values of the coefficient matrix $\mathbf{K}$. The Tikhonov solution is then split into two components, one lying in the column space of $\mathbf{W}$ and one lying in its orthogonal complement. The first component is expected to carry a dominating part of the information about the reconstructed source, and this component is computed by means of a direct equation solver that factorizes a $k \times k$ matrix. The latter component is a correction necessary to obtain the Tikhonov solution, and this component is computed by the LSQR algorithm.

The gain in the use of the subspace splitting comes from the fact that usually we can choose the number of columns k in $\mathbf{W}$ much smaller than the dimensions of $\mathbf{K}$. The direct method is then used to solve a small $k \times k$ system, while LSQR is still used to solve a large system for a correction term. The splitting of the solution into two components thus, effectively, acts as a *preconditioner* for LSQR. For more details about the subspace preconditioned algorithm, hereafter referred to as SP-LSQR, we refer to [11] and the appendix.

The choice of the columns in the matrix $\mathbf{W}$ is important. Ideally we want to use the principal singular vectors, but these vectors are too expensive to compute to be of practical use. One alternative is to use the first k basis vectors of the discrete cosine transform (DCT). These vectors have the property that they are smooth and have an increasing number of sign changes – thus resembling the generic behavior of the singular vectors of $\mathbf{K}$. We shall refer to this matrix as $\mathbf{W}_{\text{DCT}}$, and we remark that all computations involving $\mathbf{W}_{\text{DCT}}$ can be implemented very efficiently by means of the fast Fourier transform (FFT) algorithm.

In connection with the geomagnetic inverse problem considered here, we found in [11] that another good choice of $\mathbf{W}$ consists of the principal right singular vectors of the coefficient matrix corresponding to a *single* level of measurements. This

D / W	LSQR I	LSQR $\mathbf{D}_{xyz}$	LSQR $\mathbf{D}_z$	SP-LSQR — I — $\mathbf{W}_{\text{DCT}}$	SP-LSQR — I — $\mathbf{W}_{\text{SVD}}$	SP-LSQR — $\mathbf{D}_{xyz}$ — $\mathbf{W}_{\text{DCT}}$	SP-LSQR — $\mathbf{D}_{xyz}$ — $\mathbf{W}_{\text{SVD}}$	SP-LSQR — $\mathbf{D}_z$ — $\mathbf{W}_{\text{DCT}}$	SP-LSQR — $\mathbf{D}_z$ — $\mathbf{W}_{\text{SVD}}$
(A) problem with $N_x \times N_y \times N_z = 12 \times 12 \times 15$									
sec.	2987	534	55	1984	903	317	219	53	38
its.	5597	1586	235	2009	1048	452	304	68	43
(B) problem with $N_x \times N_y \times N_z = 20 \times 20 \times 15$									
sec.	5185	13221	11827	1722	979	5212	3253	4228	2497
its.	1656	4663	4026	572	321	1486	903	1338	777

Table 1: Comparison of performance of the ordinary LSQR algorithm with no preconditioning, and the new subspace preconditioned version SP-LSQR. Each cell contains two numbers: the computing time in seconds to reach a relative accuracy of 10^{-3} (top) and the number of iterations (bottom). (A) is a small system with $N_x \times N_y \times N_z = 12 \times 12 \times 15$, while (B) is a larger system with $N_x \times N_y \times N_z = 20 \times 20 \times 15$. In both problems the number of data levels is $L = 5$.

coefficient matrix is much smaller than the matrix $\mathbf{K}$ in a multi-level problem, thus making the computation of the singular vectors feasible. We refer to this SVD-based matrix as $\mathbf{W}_{\text{SVD}}$.

4.2 Comparison of Performance

The following numerical experiments demonstrate the performance gain achieved by using the subspace preconditioned LSQR algorithm, compared to the standard LSQR method. We use two test problems, both with $L = 5$ data levels and a noise level $\sigma_{\text{noise}} \approx 10^{-3}$ in the data. The small test problem uses a discretization with $N_x \times N_y \times N_z = 12 \times 12 \times 15$ cells and 144 columns in both $\mathbf{W}_{\text{DCT}}$ and $\mathbf{W}_{\text{SVD}}$. The large test problem is a $20 \times 20 \times 15$ discretization with 200 columns in $\mathbf{W}_{\text{DCT}}$ and $\mathbf{W}_{\text{SVD}}$. The exact solutions to the two test problems are shown in Fig. 5.

In order to compare the performance of the LSQR and SP-LSQR algorithms, we first compute the Tikhonov solution $\mathbf{m}_\lambda$ explicitly by means of a QR factorization. In all cases the regularization parameter λ was chosen to yield the Tikhonov solution closest to the exact solution $\mathbf{m}$.

Then we compute the Tikhonov solution again using the iterative algorithms, and we stop the iterations when the iterative solution $\widetilde{\mathbf{m}}_\lambda^{(j)}$ after j iterations has achieved a relative accuracy of 10^{-3}, i.e., when

$$\frac{\|\mathbf{m}_\lambda - \widetilde{\mathbf{m}}_\lambda^{(j)}\|_2}{\|\mathbf{m}_\lambda\|_2} \leq 10^{-3}.$$

Table 1 shows the computing times, using a Matlab implementation of the algorithms, while the computed solutions are shown in Figs. 6 and 7.

From the table we see that the computing times (in seconds) for the subspace-preconditioned LSQR methods can be up to three times smaller than those for the standard LSQR method – in spite of the fact that each SP-LSQR iteration requires

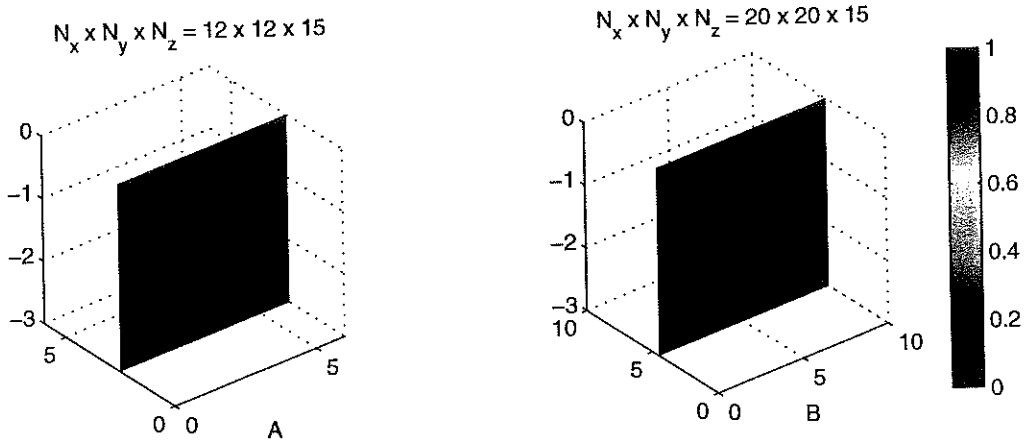

Figure 5: Exact solutions for the small and the larger test problems mentioned in the text.

more computing time than a LSQR iteration. The reason is, of course, that much fewer iterations are needed in SP-LSQR, compared to LSQR – which is an effect of the choice of the subspace matrix $\mathbf{W}$. We also see that $\mathbf{W}_{\text{SVD}}$ leads to somewhat smaller computing times, at the expense of the need to explicitly compute and store the $\mathbf{W}_{\text{SVD}}$ matrix (while the DCT-based approach can be implemented without the matrix $\mathbf{W}_{\text{DCT}}$ because all computations are implemented using the FFT algorithm). The time to compute $\mathbf{W}_{\text{SVD}}$ is 6 seconds and ? seconds, respectively, for the small and the large problem.

The table also shows that the number of iterations for both LSQR and SP-LSQR depends on the choice of the matrix $\mathbf{D}$. This is expected, since the choice of $\mathbf{D}$ influences the properties (and therefore the potential for achieving depth resolution) of the Tikhonov solutions.

Figures 6 and 7 show the computed solutions using QR factorization, LSQR, and SP-LSQR with $\mathbf{W}_{\text{SVD}}$ and $\mathbf{W}_{\text{DCT}}$, and using our three different choices of $\mathbf{D}$. These figures clearly illustrate that we compute the same solutions with the different iterative methods, which is as desired – only the computing time is different. For both test problems we achieve the best reconstruction from noisy data with $\mathbf{D} = \mathbf{D}_z$, while the two other choices of $\mathbf{D}$ lead to less satisfactory reconstructions; especially the reconstruction for $\mathbf{D} = \mathbf{D}_{xyz}$ has a "shadow" and the source appears to be located too deep.

Figure 8 shows convergence histories for the different methods applied to the small test problems, i.e., plots of the relative error $\|\mathbf{m}_\lambda - \widetilde{\mathbf{m}}_\lambda^{(j)}\|_2 / \|\mathbf{m}_\lambda\|_2$ versus the number of iterations j. Clearly the convergence rate improves as the dimension k of the subspace (i.e., the number of columns in the matrix $\mathbf{W}$ increases. For the DCT-based preconditioner, the convergence improvement is quite inexpensive because the all the DCT computations are done with the FFT algorithm, i.e., without the need to store the matrix $\mathbf{W}_{\text{DCT}}$ explicitly. For the SVD based preconditioner, one should take into account the overhead involved in the explicit computation and storage of the matrix $\mathbf{W}_{\text{SVD}}$. The actual choice between the two preconditioners is obviously

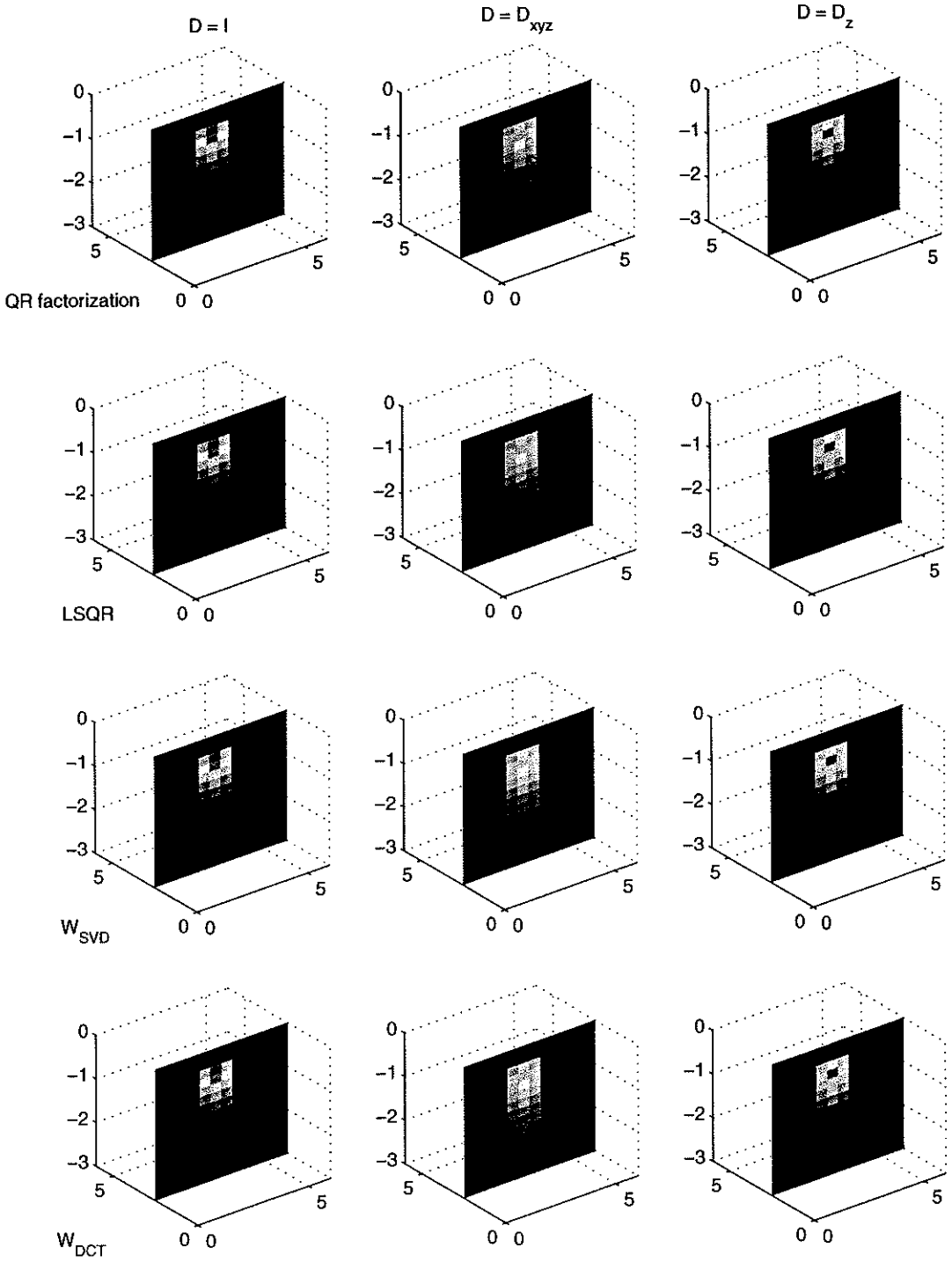

Figure 6: Tikhonov solutions to the small and noisy data problem with $N_x \times N_y \times N_z = 12 \times 12 \times 15$. computed with QR factorization, LSQR, and SP-LSQR with both $\mathbf{W}_{\text{SVD}}$ and $\mathbf{W}_{\text{DCT}}$. The solution depends on the choice of $\mathbf{D}$, but not on the choice of iterative algorithm or the choice of $\mathbf{W}$, as desired.

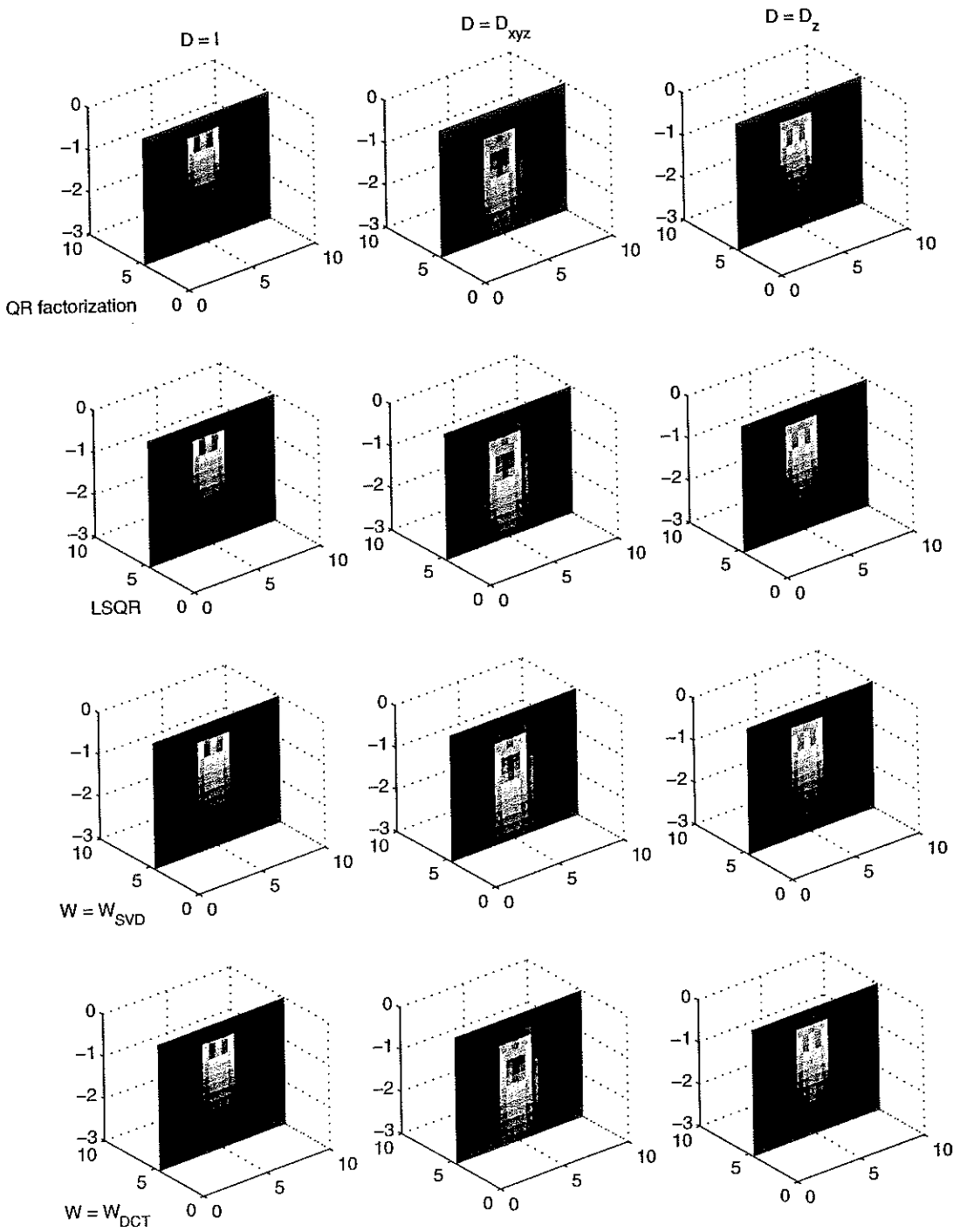

Figure 7: Same as previous figure, except that the problem is now larger, with dimensions $N_x \times N_y \times N_z = 20 \times 20 \times 15$.

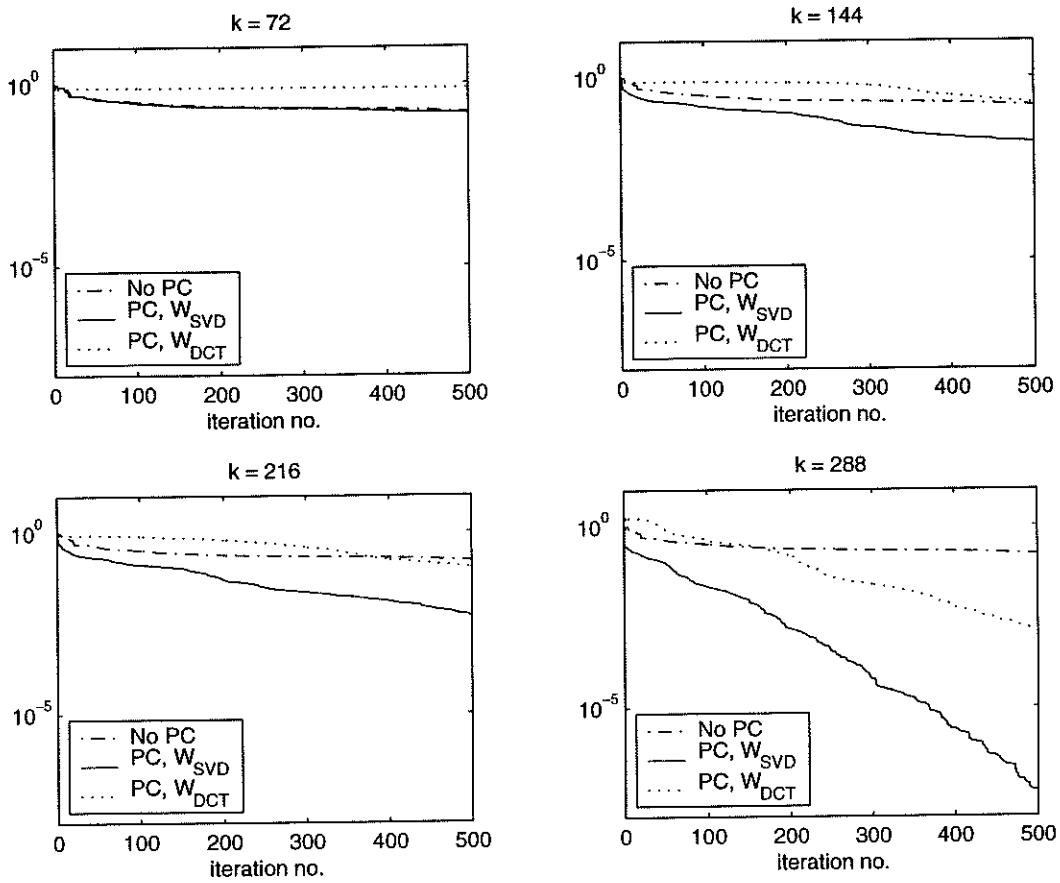

Figure 8: Convergence histories, i.e., plots of the relative error $\|\mathbf{m}_\lambda - \widetilde{\mathbf{m}}_\lambda^{(j)}\|_2 / \|\mathbf{m}_\lambda\|_2$ versus the number of iterations j, for LSQR and SP-LSQR with $\mathbf{W}_{\text{SVD}}$ and $\mathbf{W}_{\text{DCT}}$, and with four different values of the subspace dimension k (equal to the number of columns in the matrix $\mathbf{W}$.

problem dependent.

The main conclusion is that we can compute regularized solutions faster with the new algorithm SP-LSQR than with LSQR, without loosing any accuracy. The new preconditioned method is always faster than the ordinary method, independent of the choice of $\mathbf{D}$.

5 Conclusion

We have demonstrated how to perform an analysis of the discretization parameters involved in the numerical treatment of multi-level geomagnetic inverse problems, in order to achieve the best depth resolution possible within the given noise level in the data. We also demonstrated the use of a new preconditioner for the LSQR iterative least-squares solver, and illustrated the performance gain that can be achieved by the preconditioner.

Appendix: LSQR and SP-LSQR

As mentioned in section 4, the Tikhonov problem is actually a linear least squares problem if the form (15), and this problem should be solved by specialized versions of CG, such as CGLS or LSQR [2]. These methods apply the CG method implicitly to the normal equations associated with (15):

$$\left(\mathbf{K}^T\mathbf{K} + \lambda^2\,\mathbf{D}^T\mathbf{D}\right)\mathbf{m} = \mathbf{K}^T\mathbf{d}. \tag{16}$$

In this study, we use LSQR which is known to be more accurate than CGLS for ill-conditioned problems. The LSQR algorithm produces a sequence of solutions that is guaranteed to converge to the Tikhonov solution $\mathbf{m}_\lambda$, and the number of iterations is proportional to the condition number of the normal equation matrix in (16):

$$\mathbf{A} = \mathbf{K}^T\mathbf{K} + \lambda^2\,\mathbf{D}^T\mathbf{D}.$$

For the case $\mathbf{D} = \mathbf{I}$, this condition number is approximately given by

$$\mathrm{cond}(A) \simeq \|\mathbf{K}\|_2^2 \,/\, \lambda^2,$$

and although this number is smaller than the condition number for the unregularized problem, the convergence may still be quite slow. Hence there is a need to speed up the iterations and we show this can achieved by introducing preconditioning.

Preconditioning for Tikhonov regularization is known to be difficult because the normal equation matrix $\mathbf{A}$ has a large cluster of singular values at λ^2. Hanke and Vogel [9] proposed a subspace preconditioner which takes the characteristics of the Tikhonov problem into account. Their idea is to choose a so-called "smooth subspace" spanned by the k columns of a matrix $\mathbf{W}$, and then formally consider the problem

$$\begin{pmatrix} \mathbf{W}^T\mathbf{A}\,\mathbf{W} & \mathbf{W}^T\mathbf{A}\,\mathbf{Z} \\ \mathbf{Z}^T\mathbf{A}\,\mathbf{W} & \mathbf{Z}^T\mathbf{A}\,\mathbf{Z} \end{pmatrix} \begin{pmatrix} \mathbf{y}_w \\ \mathbf{y}_z \end{pmatrix} = \begin{pmatrix} \mathbf{W}^T\mathbf{d} \\ \mathbf{Z}^T\mathbf{d} \end{pmatrix} \tag{17}$$

in which

$$\begin{pmatrix} \mathbf{y}_w \\ \mathbf{y}_z \end{pmatrix} = (\ \mathbf{W}\ \ \mathbf{Z}\ )^{-1}\,\mathbf{m}. \tag{18}$$

Then the Tikhonov solution is given by

$$\mathbf{m}_\lambda = \mathbf{W}\,\mathbf{y}_w + \mathbf{Z}\,\mathbf{y}_z. \tag{19}$$

The matrix $\mathbf{Z}$ is chosen such that the matrix ($\mathbf{W}$ $\mathbf{Z}$) is square and invertible, and we emphasize that $\mathbf{Z}$ is introduced only for the derivation of the algorithm – it is not needed in the actual implementation.

The k columns of the matrix $\mathbf{W}$ should preferably be chosen to approximate the principal k right singular values of the matrix $\mathbf{K}$, in which case most of the Tikhonov solution is contained in the component $\mathbf{W}\,\mathbf{y}_w$ in [9], while the component $\mathbf{Z}\,\mathbf{y}_z$ contains the small but necessary correction to obtain the desired solution.

In [11] it is proved that the latter component is the solution to a least squares problem obtained from $\mathbf{K}$ and $\mathbf{W}$, and the actual implementation uses the LSQR iterative algorithm to solve this least squares problem for the term $\mathbf{Z}\,\mathbf{y}_z$. Once this term is computed by LSQR, the second term $\mathbf{W}\,\mathbf{y}_w$ is easily computed by solving a small $k \times k$ system for $\mathbf{y}_w$; see [11] for details. The key issue is that we still use LSQR to solve a problem with large dimensions, but since we are using LSQR to compute a correction term to the term $\mathbf{W}\,\mathbf{y}_w$ which, hopefully, contains most of the desired information, we can expect fast convergence of the LSQR method – at the expense of a slight overhead in setting up the modified problem and computing the term $\mathbf{W}\,\mathbf{y}_w$ after completion of the LSQR algorithm. We refer to this new algorithm as the subspace preconditioned LSQR method, SP-LSQR.

References

[1] M. L. Andersen, M. Fedi, P. C. Hansen and V. Paoletti, 2002, SVD and GSVD analysis of depth resolution in 3D multi-level potential field inversion; submitted to Inverse Problems.

[2] Å. Björck, 1996, Numerical Methods for Least Squares Problems, SIAM, Philadelphia, 1996.

[3] R. J. Blakely, 1996, Potential Theory in Gravity and Magnetic Application, Cambridge University Press.

[4] V. C. F. Barbosa and J. B. C. Silva, 1994, Generalized compact gravity inversion: Geophysics, 59, pp. 57–68.

[5] M. Fedi and A. Rapolla, 1999, 3-D inversion of gravity and magnetic data with depth resolution: Geophysics, 64, pp. 452–460.

[6] N. J. Fisher and L. E. Howard, 1980, Gravity interpretation with the aid of quadratic programming, Geophysics, 45, pp. 403–419.

[7] W. R. Green, 1975, Inversion of gravity profiles by use of a Backus-Gilbert approach: Geophysics, 40, pp. 763–772.

[8] A. Guillen and V. Menichetti, 1984, Gravity and magnetic inversion with minimization of a specific functional: Geophysics, 49, pp. 1354–1360.

[9] M. Hanke and C. R. Vogel, 1999, Two-level preconditioners for regularized inverse iroblems I: theory: Numer. Math., 83, pp. 385–402.

[10] P. C. Hansen, 1990, The discrete Picard condition for discrete ill-posed problems: BIT, 30, pp. 658–672.

[11] P. C. Hansen, M. Jacobsen and M. A. Saunders, 2002, Subspace preconditioned LSQR for linear ill-posed problems; submitted to BIT.

[12] B. J. Last and K. Kubik, 1983, Compact gravity inversion: Geophysics, 48, pp. 713–721.

[13] Y. Li and D. W. Oldenburg, 1996, 3-D inversion of magnetic data: Geophysics, 61 (1996), pp. 394–408.

OTHER IMM TECHNICAL REPORTS 2002.

Hansen, J. and J. Clausen:
Crane scheduling for a plate storage.
Nr. 1/2002. 16 pp.

Madsen, K., H.B. Nielsen and J. Søndergaard:
Robust Subroutines for non-linear optimization.
Nr. 2/2002. 54 pp.

Poulsen, N.K., M. Rostgaard and O. Ravn:
Delta-operator technique for pay-load estimation of a 2DOF flexible link robot.
Nr. 3/2002. 41 pp.

Buchholtz, M., H.R. Nielson and F. Nielson:
Experiments with succinct solvers.
Nr. 4/2002. 61 pp.

Juel, H., J. Brimberg and A. Schöbel:
Properties of 3-dimensional line location models.
Nr. 5/2002. 16 pp.

Sørensen, L. and R.V.V. Vidal:
The anatomy of soft approaches.
Nr. 6/2002. 23 pp.

Thomsen, P.G.:
A generalized runge kutta method of order three.
Nr. 7/2002. 22 pp.

Nielson, H.R.:
Computer science and engineering.
Nr. 8/2002. 115 pp.

Larsen, J.:
Speeding up the solution process for the VRPTW.
Nr. 9/2002. 25 pp.

Vidal, R.V.V.:
From Action to Learning. The systematisation of alternative consulting experiences.
Nr. 10/2002. 26 pp.

Vidal, R.V.V.:
The vision conference. Facilitating creative processes.
Nr. 11/2002. 18 pp.

Lophaven,S.N., H.B. Nielsen and J. Søndergaard:
Dace - a matlab kriging toolbox.
Nr. 12/2002. 34 pp.

Lophaven,S.N., H.B. Nielsen and J. Søndergaard:
Aspects of the matlab toolbox dace.
Nr. 13/2002. 44 pp.

Thomadsen, Tommy and J. Clausen:
Hierarchical network design using simulated annealing.
Nr. 14/2002. 29 pp.

Andersen, M.L.:
Subspace preconditioned LSQR in 3-D geomagnetic reconstruction.
Nr. 15/2002. 35 pp.

Andersen, M.L., M. Fedi, P.C. Hansen and A. Rapolla.:
Computational properties of algorithmes for multi-level inversion of 3-D potential fields.
Nr. 16/2002. 19 pp.

Jacobsen, M., P.C. Hansen and M.A. Saunders.:
Subspace preconditioned LSQR for discrete Ill-posed problems.
Nr. 17/2002. 16 pp.

Hansen, P.C. M.E. Kilmer and R.H. Kjeldsen.
Exploiting residual information in the regularization of discrete Ill-posed problems.
Nr. 18/2002. 20 pp.