

# 3-D Electromagnetic Modeling and Inversion on Massively Parallel Computers

Gregory A. Newman and David L. Alumbaugh  
*Sandia National Laboratories*  
*P.O. Box 5800 MS 0750*  
*Albuquerque, NM 87008*

## Abstract

A numerical modeling algorithm has been developed to simulate the electromagnetic response of a three dimensional earth to a dipole source for frequencies ranging from 100Hz to 100MHz. The numerical problem is formulated in terms of a frequency domain - modified vector Helmholtz equation for the scattered electric fields. The resulting differential equation is approximated using a staggered finite difference grid which results in a linear system of equations for which the matrix is sparse and complex symmetric. The system of equations is solved using a preconditioned quasi- minimum-residual method.

Dirichlet boundary conditions are employed at the edges of the mesh by setting the tangential electric fields equal to zero. At frequencies less than 1MHz, normal grid stretching is employed to mitigate unwanted reflections off the grid boundaries. For frequencies greater than this, absorbing boundary conditions must be employed by making the stretching parameters of the modified vector Helmholtz equation complex which introduces loss at the boundaries.

An iterative solution to the non linear 3-D electromagnetic inverse problem is obtained by successive linearized model updates using the method of conjugate gradients. Full wave equation modeling is employed to compute model sensitivities and predicted data in the frequency domain with the 3-D finite difference algorithm.

Necessity dictates that both the forward and inverse solutions be implemented on a massively parallel computing platform for reasonable execution times because realistic reconstructions require the solution of tens of thousands of parameters. In addition large scale 3-D forward modeling is needed for computing upwards to several million electric-field unknowns.

## 1 Introduction

The solution of the three-dimensional (3-D) electromagnetic (EM) inverse problem has been a goal of geophysicists for many years. The search for this solution has been motivated by its potential applications in mapping electrical conductivity, dielectric permittivity and magnetic permeability. Knowledge of these electrical properties are extremely important since they are needed in hydrological modeling, chemical and nuclear waste site evaluations, mineral and oil and gas exploration and more recently reservoir characterization.

Unfortunately, the solution of the 3-D inverse problem is non trivial. One obstacle to constructing a solution to this problem have been the scarcity of efficient forward modeling solutions needed for computing model sensitivities and predicted data at fine parameterization levels. Tens of thousands of cells are needed to allow for smooth reconstructions, which stabilizes the inversion process, but can require the solution of up to several million field unknowns in the forward problem.

Nevertheless, great strides have been made over the last decade in forward modeling using staggered 3-D finite differences. Druskin and Knizherman (1988 and 1994), Smith (1992), Wang and Hohmann (1993), and Newman (1995) all employ some type of staggered finite difference grid (Yee,1966) to solve for the EM fields in both the time and/or frequency domain. Yet even with these computationally efficient solutions, the complexity, and thus the realism of the models that can be simulated on traditional serial computers is limited by memory and flop rate of the processor. Moreover, implementation of a 3-D inversion capability that uses these solutions is still not practical.

However, with the rapid advancements in massively parallel computers the limitations posed by serial computers is disappearing. This is due to the fact that the rate at which the simulations can proceed is dramatically increased because thousands of processors can operate on the problem simultaneously. Because of this computational efficiency it is even possible to propose a realistic attack on the 3-D inverse problem.

Outlined below is our approach to solving the 3-D forward and inverse problems on an MP platform. For the forward problem we will examine the implementation of a frequency-domain-finite difference (FD-FD) scheme based on a staggered grid. Building on this we will implement the corresponding inverse. We will next briefly describe how to implement these schemes on a massively parallel computer. Finally we will demonstrate the forward code's usefulness over a wide frequency range for different types of geophysical scenarios and provide some initial test results for the inversion scheme.

## 2 Theoretical Development of the Forward Problem

### 2.1 Finite Difference Formulation

In order to simulate the EM response of a 3-D earth, we numerically solve the frequency domain version of the vector Helmholtz equation for the scattered electric fields using a finite difference approximation on a staggered grid(Yee ,1966). The FD solution we shall outline has been designed to compute the 3-D EM response for a wide variety of earth properties for frequencies ranging from approximately 100 hz up to 100 Mhz. This scheme is similar to those outlined in Alumbaugh and Newman (1994) and Newman and Alumbaugh (1995), but has been extended as described in Alumbaugh et al.(1996) to include both variable magnetic permeability as well as absorbing boundary conditions (ABCs). The ABCs are required to simulate the response for frequencies greater than 10 Mhz as without them erroneous results are produced. We have chosen to employ the "perfectly matched layer" (PML) absorbing boundary conditions originally developed by Berenger(1993) for 2-D time-domain calculations and later modified for 3-D by Katz et al. (1994) and Chew and Weedon (1994). This method uses a modified form of the Helmholtz equation in which the absorption is incorporated through the use of complex grid stretching.

Because the responses we are interested in simulating, for example airborne EM simulations, often employ a dipole source located far away from zones of anomalous electrical properties, we have chosen to work with the scattered field versions of the governing equations. Often this allows us to employ a coarser discretization about the source location than would be employed with a total field solution and helps to limit storage overhead. In addition, because the scattered field versions of Maxwell's equations numerically decouple in the air at low frequencies (< 10MHz) we have chosen to work with the scattered electric field version of the modified Helmholtz equation instead of the coupled Maxwell system, which has the form

$$\begin{aligned} \nabla_h \times \frac{\mu_p}{\mu} \nabla_e \times \mathbf{E}_s = -i\omega\mu_p(\sigma + i\omega\epsilon)\mathbf{E}_s - i\omega\mu_p[(\sigma - \sigma_p) + i\omega(\epsilon - \epsilon_p)]\mathbf{E}_p \\ - i\omega\mu_p \nabla_h \times [(\frac{\mu - \mu_p}{\mu})\mathbf{H}_p] \end{aligned} \quad (1)$$

where

$$\nabla_e = \mathbf{i} \frac{1}{e_x} \frac{\partial}{\partial x} + \mathbf{j} \frac{1}{e_y} \frac{\partial}{\partial y} + \mathbf{k} \frac{1}{e_z} \frac{\partial}{\partial z} \quad (2)$$

and

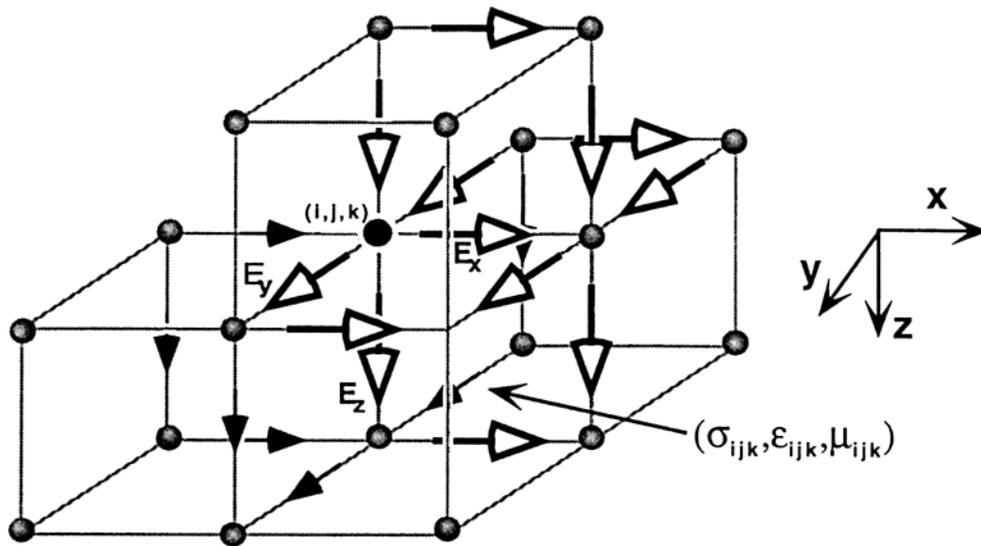


Figure 1: Finite difference stencil for solving the scattered electric field Helmholtz equation. The arrows represent the position of the electric fields, with the open arrows representing those unknowns needed to form the equation for  $E_x$ .

$$\nabla_h = \mathbf{i} \frac{1}{h_x} \frac{\partial}{\partial x} + \mathbf{j} \frac{1}{h_y} \frac{\partial}{\partial y} + \mathbf{k} \frac{1}{h_z} \frac{\partial}{\partial z}. \quad (3)$$

In these equations the electrical conductivity, magnetic permeability and dielectric permittivity are denoted by  $\sigma$ ,  $\mu$ , and  $\epsilon$ , respectively, with the 'p' designating a whole-space background value,  $\mathbf{E}_s$  and  $\mathbf{E}_p$  are the scattered and primary electric field whose sum is equal to the total electric field,  $\mathbf{H}_p$  is the primary magnetic field and  $e_i$  and  $h_i$  for  $i=x,y,z$  are coordinate stretching variables which stretch the  $x,y$ , and  $z$  coordinates. As shown in Chew and Weedon (1994), when  $e_i$  and  $h_i$  are complex then the medium is perfectly absorbing. Note that the terms at the end of equation (1) are "equivalent source" terms which are calculated wherever the properties of the medium are different from that of the assumed background. The boundary conditions employed are Dirichlet conditions, i.e., the tangential component of  $\mathbf{E}_s$  is set to zero on the grid boundary.

The scattered electric fields are assigned to each cell following the staggered grid scheme given in Figure 1. For node  $(i, j, k)$ , the  $x$ ,  $y$  and  $z$  components of electric field are sampled at  $(i + 1/2, j, k)$ ,  $(i, j + 1/2, k)$  and  $(i, j, k + 1/2)$ , respectively. For modeling the coupled Maxwell's equations this corresponds to assigning the electric fields to the edges of the cell and the magnetic field to its faces. In addition this formulation requires that the conductivity and dielectric permittivity be computed halfway along a given cell edge in Figure 1, and the magnetic permeability to be computed in the center of the cell face. This is accomplished through the simple averaging schemes described in Alumbaugh et al. (1995).

After numerically discretizing equation (1) to form the finite difference equations a linear system is assembled,

$$\mathbf{K}\mathbf{E}_s = \mathbf{s} \quad (4)$$

where  $\mathbf{K}$  is the stiffness matrix containing the numerical approximations to the derivatives as well as the electrical properties of the medium,  $\mathbf{E}_s$  is the unknown vector for the scattered electric field and  $\mathbf{s}$  is the equivalent source vector. Alumbaugh et al.(1996) show that  $\mathbf{K}$  is complex symmetric, even when complex grid stretching is employed. The solution vector can be obtained using the quasi-minimum residual (QMR) (Freund ,1992) technique with preconditioning to iteratively determine

the solution within a predetermined error level, which is defined here to be

$$er = \frac{\|\mathbf{K}\mathbf{E}_s - \mathbf{s}\|^2}{\|\mathbf{s}\|^2}. \quad (5)$$

Tests with different types of incomplete decomposition and polynomial preconditioners has found that simple Jacobi scaling provides a simple, time efficient method of preconditioning.

After the scattered fields at the grid points have been determined, the fields at the receivers must be calculated. The electric field is simply calculated using bi-linear interpolation while the magnetic field is calculated by first taking a numerical approximation of Faraday's law for the scattered electric fields on the grid surrounding the receiver,

$$\nabla_e \times \mathbf{E}_s = -i\omega\mu\mathbf{H}_s + (\mu - \mu_p)\mathbf{H}_p \quad (6)$$

and then interpolating the result to the point of interest. In this expression  $\mathbf{H}_s$  is the scattered magnetic field. Note, for each new source position and/or frequency a new system must be solved, although some time savings can be implemented by using the previous solution vector as an initial guess.

## 2.2 Properties of the PML Absorbing Boundary Condition

Although their calculations employ the coupled modified Maxwell's equations in the time domain, Chew and Weedon (1994) develop theory in the frequency domain to demonstrate how lossy, non-reflecting conditions are created along the mesh boundaries. The complex stretching parameters are assigned a value of the form  $1+a - ib$ . On the internal portion of the mesh,  $a = b = 0$  such that the modified Helmholtz equation reduces to the normal form. Near the edges of the mesh  $e_i$  and  $h_i$  are allowed to vary over several cells, but only in the direction that is perpendicular to the boundary. For example along the  $+z$  boundary  $e_x = e_y = h_x = h_y = 1$  and only  $e_z$  and  $h_z$  are allowed values of  $a$  and  $b$  that are not equal to zero.

Because we are solving an implicit rather than explicit system, we have found that in order to incorporate a given amount of loss, or attenuation, across a number of cells serving as the absorbing boundary, it is better to set  $a$  and  $b$  constant rather than gradually increasing their value toward the mesh boundaries as suggested by Berenger (1993); gradually increasing their value results in a greater number of iterations needed to achieve convergence. Simple MATLAB experiments have shown that this is due to the fact that the condition number of  $\mathbf{K}$  increases as the ratio between the complex amplitudes of the largest and smallest cell dimensions in the mesh increases. Thus gradually increasing the stretching parameters outward will produce a cell along the edge of the mesh which is effectively much larger than any of the cells employing constant stretching. Because the smallest cell size is the same in either case, the solution of the model that employs the gradual stretching will take longer to converge.

Currently, we are investigating methods for choosing optimal stretching parameters for a given frequency and background wave number, defined as

$$k_p = \sqrt{-i\omega\mu_p(\sigma_p + i\omega\epsilon_p)} = \alpha - i\beta. \quad (7)$$

where  $\alpha$  and  $\beta$  are both real. This analysis is based on the assumption that the loss that is incorporated through complex grid stretching is caused by 'pseudo' electrical parameters within cells of constant size. Through this assumption we can develop a pseudo-skin depth in each cell which is defined by

$$\delta^{ps} = \frac{1}{(\alpha b + \beta a)} \quad (8)$$

and a pseudo-wavelength defined as

$$\lambda^{ps} = \frac{1}{(\alpha a - \beta b)}. \quad (9)$$

To this point we have found that for frequencies greater than 1MHz, accurate results and quick solution convergence are achieved when  $a$  and  $b$  are chosen such that five pseudo-skin depth's of attenuation are provided for across the stretching region without out significantly changing the pseudo wavelength from that of the natural background wavelength. At frequencies below 100 kHz the analysis seems to become more difficult as the manner in which the grid is stretched can significantly alter the convergence of the system. In general at these frequencies we have obtained good results using only real grid stretching; i.e. setting  $b=0$  and varying only  $a$ .

## 3 Theoretical Development of the Inverse Problem

### 3.1 Regularized Least Squares

The parameterization used in the 3-D inverse solution will be kept sufficiently fine because we are interested in reconstructions that do not under parameterize the earth. This forces the 3-D inverse problem to be underdetermined, which makes it unstable and ill posed. Reliable estimates of the model parameters  $\mathbf{m}$  may be possible if a least squares inversion is stabilized with regularization (Tikhonov and Arsenin, 1977). Regularization removes solutions that are too rough by imposing an additional constraint on the data fit. Reconstructions are required to be smoothed versions of the earth's electrical properties at the expense of an increase in the fitting error.

Linearizing about a given earth model,  $\mathbf{m}^{(i)}$ , the following functional can provide smooth reconstructions if it is minimized with respect to the model parameters,  $\mathbf{m}$ , which can include electrical conductivity, dielectric permittivity and magnetic permeability:

$$S = [(\mathbf{D}(\mathbf{d} - \mathbf{d}^{P(i)} - \mathbf{A}^{P(i)}(\mathbf{m} - \mathbf{m}^{(i)}))\mathbf{H}(\mathbf{D}(\mathbf{d} - \mathbf{d}^{P(i)} - \mathbf{A}^{P(i)}(\mathbf{m} - \mathbf{m}^{(i)})) - \chi^2) + \lambda(\mathbf{W}\mathbf{m})(\mathbf{W}\mathbf{m})^t. \quad (10)$$

Here we are going to minimize model roughness  $(\mathbf{W}\mathbf{m})(\mathbf{W}\mathbf{m})^t$  subject to a specified square error,  $\chi^2$ . The superscripts  $\mathbf{H}$  and  $\mathbf{t}$  denote the Hermitian and transpose operators. The matrix  $\mathbf{W}$  is the roughness matrix consisting of a finite difference approximation to the Laplacian  $\nabla^2$  operator and is sparse. In equation (10) the observed data are represented by the vector  $\mathbf{d}$  and the predicted data arising from the model  $\mathbf{m}^{(i)}$  are denoted by  $\mathbf{d}^{P(i)}$ . The data weighting matrix  $\mathbf{D}$  is diagonal consisting of the reciprocal of the data standard deviations, the reciprocal of the data amplitude or in some instances an identity matrix if data weighting is unwarranted. The Jacobian or model sensitivities matrix,  $\mathbf{A}^{P(i)}$ , is determined from the frequency-domain forward solver algorithm described above. Note we assume the estimated model parameters to be always real. Determining and manipulating the elements of the Jacobian matrix in the most efficient manner is critical for a robust 3-D inverse solution, since calculation and use of these elements can be a bottleneck in the inversion. Derivation and efficient use of  $\mathbf{A}^{P(i)}$  in the inversion is given below.

The parameter  $\lambda$  is the tradeoff parameter between model smoothness and data fit. Its selection requires special care if the inverse solution is to provide acceptable results. There is no universal or unique strategy for selecting  $\lambda$  in equation (10). Selecting tradeoff parameters that are too small can produce models that are physically unreasonable; the models are spatially too rough, but produce superior data fits. Selecting tradeoff parameters that are too large produce highly smoothed models, but these models show poor dependence on the data. We shall defer further discussion of this parameter until we discuss the iterative nature of equation (10).

Minimization of equation (10) yields the model step, where

$$\mathbf{m} = [(\mathbf{D}\mathbf{A}^{P(i)})\mathbf{H}(\mathbf{D}\mathbf{A}^{P(i)}) + \lambda(\mathbf{W})^t(\mathbf{W})]^{-1}(\mathbf{D}\mathbf{A}^{P(i)})^t(\mathbf{D}\delta\mathbf{d}^{(i)}) \quad (11)$$

with

$$\delta \mathbf{d}^{(i)} = (\mathbf{d} - \mathbf{d}^{\mathbf{P}^{(i)}} + \mathbf{A}^{\mathbf{P}^{(i)}} \mathbf{m}^{(i)}). \quad (12)$$

Because negative parameter estimates are an admissible solution arising from equation (11), it is advisable that before minimizing equation (10), it should be reformulated so one can invert for the logarithm of the parameters, instead of the parameters themselves. This causes the imaged properties to be always positive which is a physical requirement. Using the log parameterization it is also possible to incorporate a lower bound positivity constraint in the inverse solution, where

$$\delta(m^{(i)}) = (m^{(i)} - \epsilon) \delta \ln(m^{(i)} - \epsilon), \quad (13)$$

with  $\delta m^{(i)} = (m - m^{(i)})$  along with  $\delta \ln(m^{(i)} - \epsilon) = \ln((m - \epsilon)/(m^{(i)} - \epsilon))$ , and  $m > \epsilon$  and  $\epsilon > 0$ .

### 3.2 Model Update via Conjugate Gradients

Using equation (11) to compute the updated model,  $\mathbf{m}$ , directly is not feasible for the full 3-D problem since direct matrix inversion is prohibitive, even on an MP platform, when the number of unknowns exceed several thousand. Instead we opt for an iterative solution. Since equation (10) satisfies the normal equations, the linear system is symmetric semidefinite even when regularized. Thus the conjugate gradient (CG) method of Hestenes and Stiefel (1952) can be used to get the solution. More importantly, following Mackie and Madden (1993) and Zhang et al. (1995), it is possible to avoid explicitly forming the Jacobian Matrix,  $\mathbf{A}^{\mathbf{P}^{(i)}}$  and its Hermitian form altogether with this approach, thus saving considerable computer storage. In the conjugate gradient methods all one needs is one matrix-vector multiply per relaxation step. Because the matrix in question is given by the product of  $\mathbf{D}\mathbf{A}^{\mathbf{P}^{(i)\mathbf{H}}}$  with  $\mathbf{D}\mathbf{A}^{\mathbf{P}^{(i)}}$  we really require two matrix vector multiplies instead of one; in addition two matrix-vector multiplies arise in the CG routine from the regularization matrix  $\mathbf{W}$  and its transpose. However, these multiplies are easy to compute and need no further elaboration until the MP implementation. Explicitly we have

$$\mathbf{y} = \mathbf{D}\mathbf{A}^{\mathbf{P}^{(i)}} \mathbf{u} \quad (14)$$

and

$$\mathbf{z} = \mathbf{D}\mathbf{A}^{\mathbf{P}^{(i)\mathbf{H}}} \mathbf{y}, \quad (15)$$

where  $\mathbf{u}$  is an arbitrary real vector,  $\mathbf{y}$ , is a complex vector and  $\mathbf{z}$  is a real vector.

For the CG algorithm to be effective it is necessary to compute the matrix-vector products with utmost efficiency, which requires efficient manipulation of the Jacobian matrix elements. To show how this can be accomplished, consider a single data measurement defined for a given transmitter where

$$d_j = d_j^b + \mathbf{g}_j^t \mathbf{E}_s. \quad (16)$$

In this equation  $d_j^b$  is the background field at location  $j$  and is either specified by  $\mathbf{E}_p$  or  $\mathbf{H}_p$  and  $\mathbf{E}_s$  is the scattered electric field vector with dimension of  $NT \times 1$  and is determined from our forward solver at the node points from equation (4),  $NT$  represents the number of field unknowns, and the vector  $\mathbf{g}_j^t$  is an interpolator vector for the  $j$ th measurement point and is of dimension  $1 \times NT$ . This vector will interpolate the field values on the staggered grid to the measurement point of interest and can also numerically approximate the curl of the electric field so that magnetic field measurements are allowed for in equation (16). With this definition an element of the Jacobian matrix is written as

$$\partial d_j / \partial m_k = \mathbf{g}_j^t \partial \mathbf{E}_s / \partial m_k. \quad (17)$$

From the forward problem we know that the scattered electric fields are related to the source vector of a given transmitter,  $\mathbf{s}$ , by the linear system

$$\mathbf{K}\mathbf{E}_s = \mathbf{s}, \quad (18)$$

where  $\mathbf{K}$  is the sparse finite-difference stiffness matrix. Thus,

$$\partial\mathbf{E}_s/\partial m_k = \mathbf{K}^{-1}(\partial\mathbf{s}/\partial m_k - \partial\mathbf{K}/\partial m_k\mathbf{E}_s) \quad (19)$$

and an element of the Jacobian matrix can be written as

$$\partial d_j/\partial m_k = \mathbf{g}_j^t \mathbf{K}^{-1}(\partial\mathbf{s}/\partial m_k - \partial\mathbf{K}/\partial m_k\mathbf{E}_s). \quad (20)$$

We now determine the  $j$ th element of the first matrix vector multiply in equation (14) to be

$$y_j = \text{Cmplx}(\text{Re}(\mathbf{g}_j^t \mathbf{K}^{-1} \sum_{k=1}^M u_k (\partial\mathbf{s}/\partial m_k - \partial\mathbf{K}/\partial m_k \mathbf{E}_s)) \text{Re}(D_{jj}), \quad (21)$$

$$\text{Im}(\mathbf{g}_j^t \mathbf{K}^{-1} \sum_{k=1}^M u_k (\partial\mathbf{s}/\partial m_k - \partial\mathbf{K}/\partial m_k \mathbf{E}_s)) \text{Im}(D_{jj}))$$

where  $M$  is the total number of parameters to be estimated and  $D_{jj}$  is the  $j$ th diagonal entry of the matrix  $\mathbf{D}$ . Using the same approach one can also show that for the second matrix- vector multiply

$$z_k = \text{Re}(\sum_{i=1}^N \text{Cmplx}(\text{Re}(d_i)\text{Re}(y_i), \text{Im}(d_i)\text{Im}(y_i))^* \mathbf{g}_i^t \mathbf{K}^{-1} (\partial\mathbf{s}/\partial m_k - \partial\mathbf{K}/\partial m_k \mathbf{E}_s)), \quad (22)$$

where  $N$  is the amount of data used in the inversion and the symbol '\*' stands for complex conjugation. Note that in equations (21) and (22) the term  $(\partial\mathbf{s}/\partial m_k - \partial\mathbf{K}/\partial m_k \mathbf{E}_s)$  is rapid to compute, where the vector  $\partial\mathbf{s}/\partial m_k$  has 12 non-zero entries and the matrix  $\partial\mathbf{K}/\partial m_k$  can have 12 non-zero entries if  $m_k$  represents the conductivity or permittivity or 77 non-zero entries if  $m_k$  represents the magnetic permeability. It is important to mention here that when magnetic permeability properties are being estimated we restrict the measurement point to be in uniform background medium, such as air. In that way the interpolator vector  $\mathbf{g}_j^t$  will not depend on changes in the magnetic permeability and equation (20) remains valid. Recall this vector operator approximates Faraday's law and because of this requires the magnetic permeability to be determined at the measurement point in order to compute the magnetic field.

In addition to the forward solves needed for the different transmitters, we can carry out the matrix-vector multiplies in equations (21) and (22) efficiently by solving a series of forward problems corresponding to the total number of unique data measurements locations, where

$$\mathbf{v}_j^t = \mathbf{g}_j^t \mathbf{K}^{-1}, \quad (23)$$

or

$$\mathbf{K}\mathbf{v}_j = \mathbf{g}_j, \quad (24)$$

since  $\mathbf{K}^t = \mathbf{K}$  (recall that the matrix  $\mathbf{K}$  is complex symmetric). A unique measurement location comprises a specific field component measurement made at a site using different transmitters positions at a single frequency. Thus to get the total number of forward solves needed for each model update we have  $N_{rx} + N_{tx}$ , where  $N_{tx}$  and  $N_{rx}$  are the total number of transmitters and unique receiver positions used in the inversion; note that use of multiple frequency data will require additional forward solves for the sources and unique receiver positions. Handling the Jacobian matrix

elements in this manner is much more efficient than attempting to compute them directly using equation (20) and then using the results to form the matrix-vector multiplies. For example if we are estimating over 30,000 parameters this would then require 30,000 separate forward solves, which is impractical. On the other hand because the amount of data used in the inversion is limited, we anticipate no more than several hundred to several thousand forward solves per model update with our approach. This approach has also been recommended by McGillivray and Oldenburg (1990) and Oldenburg (1990) because of its efficiency. It has been used by Park (1983), Mackie and Madden (1993) and Zhang et al. (1995) in their constructions of the inverse solution.

### 3.3 An Iterative Solution

Because of the computational cost of using an exact forward solution in the inversion we do not have the luxury of slowly reducing the tradeoff parameter or determining an optimal  $\lambda$  at a given iteration to insure against arbitrarily rough models. However, experience indicates that smooth models can be produced with the strategy we are now going to discuss.

We initiate an inversion assuming an initial background model, where we compute the predicted data for all transmitter locations. At the first iteration we use our scheme to efficiently determine the matrix-vector multiplies in the CG algorithm and determine the model update via equation (11). This model is determined once the tradeoff parameter,  $\lambda$ , is selected. To assure a smoothed solution at the first iteration, we select the tradeoff parameter as

$$\lambda = \text{MaxRowSum}(\mathbf{DA}^{\mathbf{P}^{(i)}})^{\mathbf{H}}(\mathbf{DA}^{\mathbf{P}^{(i)}})/2^{i-1}, \quad (25)$$

where  $i=1$  for the first iteration. We have selected this method of choosing  $\lambda$  because it is an estimate of the largest eigenvalue of the non-regularized least squares system matrix. Thus weighting  $\mathbf{W}^t \mathbf{W}$  by this amount allows only the largest eigenvalues to influence the solution. The maximum row sum is easy to compute and follows from equations (14) and (15) with  $\mathbf{u}$  selected to be the unit vector.

We proceed to the next iteration if the data error (sum of square errors) is above  $\chi^2$ . If this is true, the model is linearized again about the new model  $\mathbf{m}$ , the predicted data and electric fields are computed from the updated background model, and then the new model update determined once the tradeoff parameter is specified with equation (25). In general we have found that for the first few iterations this method of selecting the tradeoff parameter reduces the error by about a factor of 2. The iterative procedure, just outlined, is continued until the data error is below  $\chi^2$ , convergence occurs or a pre specified number of iterations has taken place.

Even with this procedure, it is possible to drive the tradeoff parameter down too quickly with equation (25), especially when one attempts to fit the data to an unrealistic noise level or uses an excessive number of iterations. However, it has been our experience that if the tradeoff parameter is not relaxed sufficiently the inversion can stall out far above the estimated noise level in the data. Our solution to this difficulty is to have a good estimate of the data noise and monitor the tradeoff parameter and squared error in the inversion. If excessive model structure is being incorporated into the image, or if the inversion is over fitting the data, we stop the inversion and relaunch it using an acceptable reconstruction and tradeoff parameter at some previous iteration. After this restart, the tradeoff parameter is kept fixed for the rest of the inversion. In addition, we may change the data weighting scheme if it is decided that bad data are weighted too large or good data too little. While this strategy is somewhat subjective, it has yielded acceptable results.

At each iteration we restrict the number of relaxation steps in the CG routine, since only a modest number of steps are sufficient to produce an accurate model update, especially during the early stages of scheme (Zhang et al., 1995). For the first and second iterations, 20 and 40 relaxation steps are used, respectively. Subsequent iterations use 60 steps.

## 4 Implementation on Massively Parallel Computers

In order to simulate larger, more realistic models and more importantly, image more complex structures than has previously been possible, the original serial versions of the forward and inverse codes have been modified to run on massively parallel MIMD (multiple instruction multiple data) machines. Such machines can have thousands of processors and are employed by assigning a given number of processors in each direction of the forward and inverse modeling domain ( $n_x$  in  $x$ ,  $n_y$  in  $y$  and  $n_z$  in  $z$ ) and then breaking up the model across the processor bank such that each individual processor is in charge of a 3-D subset, with all processors sharing the same data. Because each processor needs only to make the necessary calculations for this subset, and because all of the processors are making their appropriate calculations simultaneously, the solution time is reduced by a factor which is approximately equal to the total number of processors employed ( $n_x*n_y*n_z$ ).

The first step in converting the serial version of the code to a parallel version is to divide the problem up among the processors such that it is optimally load balanced. This preprocessing step is necessary to ensure that large banks of processors are not standing idle for long periods of time while a single or small number of processors complete their calculations. As one would imagine, this type of scenario is an extremely inefficient use of resources. Rather the problem is broken up such that each processor has as close to an equal number of calculations as possible.

The second issue that needs to be addressed is the manner in which the model is input; for the inverse this could constitute a starting model needed to launch the inverse or a restart model in the event that the inversion needs to be restarted midway through the process. To accomplish the input, we have decomposed the input data into two different sets: a global data set and a local data set. Global data are those variables that each processor needs to know such as the source and receiver positions, the frequencies, what type of solver is being employed, the location of the mesh nodes, etc. These form a fairly small data set which can easily be read in by a "lead" processor and then "broadcast" to all other processors. The second type of input is the local data, or local model parameters (conductivity, dielectric permittivity and magnetic permeability) that are assigned to each cell within the model. Because each processor needs only a small subset of this data and contains only a small amount of local memory, the local data is broken up into multiple files, one for each processor, which are then read in individually from a parallel disk system which allows several files to be read in simultaneously.

Even with the increased performance of an MP platform, memory considerations will dictate the largest model that can be simulated, particularly for the inverse. EM inversion in 3-D can easily require the solution of at least several hundred forward solves per iteration. We also anticipate that each solve could constitute over a million field unknowns. Nevertheless, it is still possible on large scale platforms, such as the 1840 node Intel Paragon, to execute all solves without writing to disk. A significant portion of the storage required to preform the inversion is taken up by the electric field solution vectors that are obtained from the forward solver. These vectors are needed to complete matrix-vector multiplies in the CG routine, previously discussed.

### 4.1 Message Passing Required in the Forward Problem

In order to complete the calculations required in the forward problem, information will need to be exchanged between processors. This exchange of information is called message passing. To show how it arises, consider the forward problem after the data have been accessed and each processor has constructed its own portion of the stiffness matrix  $\mathbf{K}$  and the source vector  $\mathbf{s}$ .

Each processor proceeds to solve for its portion of the solution vector in equation (4). However, each iteration within the QMR solver requires one matrix-vector multiply and several vector dot products. These operations pose problems because in order to complete them, information must be exchanged both between all of the processors as well as small subsets of processors. The dot

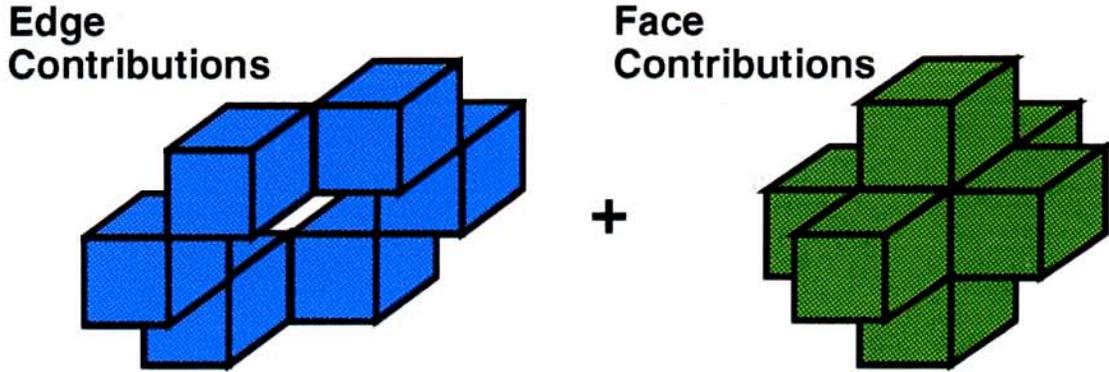


Figure 2: Processor stencil employed for message passing in order to correctly complete the matrix-vector multiply for the forward problem.

products are fairly easy to implement as they involve; 1) a local calculation in which each processor computes the dot-product of its portion of the vector and 2) a global calculation in which all the local calculations are "gathered" by the lead processor, summed, and the result broadcast across the machine.

The vector-matrix multiply is more difficult to implement because it requires that each processor communicates with those "neighboring" processors that are solving for the scattered electric fields in adjacent portions of the model. Determining these neighboring processors and the actual unknowns that need to be communicated is accomplished in the following manner. If we assume that each processor contains only a single node, then we can imagine it as a cubic shape enclosing node  $(i, j, k)$  as well as all other nodes in Figure 1. Careful examination then indicates that there are two types of communication that each processor needs to execute with its appropriate neighbors. The first type of communication will occur across the "faces" of the cube. For node  $(i, j, k)$  this implies communication with those nodes directly connected to it by the gray lines of the finite difference stencil, i.e., nodes  $(i - 1, j, k)$ ,  $(i + 1, j, k)$ ,  $(i, j - 1, k)$ ,  $(i, j + 1, k)$ ,  $(i, j, k - 1)$  and  $(i, j, k + 1)$ . For these communications either two or three unknowns are exchanged per nodal position. The second type of communication occurs across certain "edges" of the cube, and involves those nodes which are not directly connected to  $(i, j, k)$  by the stencil lines, for example node  $(i + 1, j, k - 1)$ . This type of communication requires only one unknown per node being communicated each way. If we now expand the idea such that each processor cube contains a 3-D distribution of nodal points, then we can develop the processor communication stencil shown in Figure 2.

The last point to be addressed is the message passing needed for data output. Because for any given source we only need to know the results at only a limited number of receiver positions, all of which may lie on the same processor, the data output is inherently non-parallel and is accomplished in the following steps. 1) Each processor determines which processor holds the portion of the model that contains the receiver position. 2) This "receiver" processor then determines if it need any values from adjacent processors, completes the necessary point-to-point communication with those processors, and then does the necessary bilinear interpolation. 3) The results are then sent to the lead processor which outputs them to disk.

## 4.2 Message Passing Required in the Inverse Problem

Beyond the message passing needed in the forward problem, there will be additional message passing needed for the inverse. Of primary importance is to limit the amount of inter-processor communication within the CG routine, which incorporates the matrix vector multiplies for both the Jacobian and regularization matrices. To conserve space we only show one processor stencil here which can

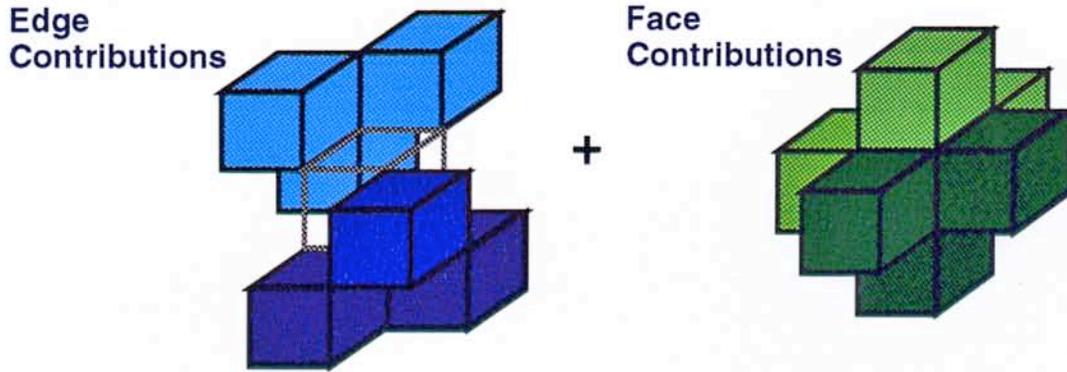


Figure 3: Processor stencil employed for message passing in order to correctly complete the matrix-vector multiplies in the inverse.

be used to explain the three different local communications needed within the inverse. The local communication pattern for a given processor is illustrated in Figure 3.

The communication needed to complete the matrix-vector multiplies involving the Jacobian matrix consists along the faces of processors as well as along edges. Thus before the CG routine is called, electric field values arising from the forward solver are exchanged which provides each processor with the necessary values to complete the calculations in equations (21) and (22). Specifically information is passed from the central processor to those neighbors designated by the lighter colors in Figure 3. Likewise those neighboring processors which are darker pass information to the central processor.

Local communication for multiplies with the regularization matrix and its transpose involve only communication along processor faces in Figure 3. Here all such processors send elements of the CG vectors to the central processor as well as receive from it. Local communication occurs every time the matrix-vector multiply is encountered in the CG routine because we have explicitly formed the regularization matrix and the CG vectors are constantly updated for each relaxation step. In addition there are five global dot products within a generic CG routine and one in equation 21 that incur additional global communication overhead at each iteration.

After exiting the CG routine, additional message passing is needed. This is because electrical properties for cells along processor boundaries need to be communicated with neighboring processors in order to accurately calculate the correct average conductivity, dielectric permittivity and magnetic permeability for subsequent forward modeling in the inversion. Those face and edge processors designated with lighter colors send to the central processor, while those that are darker receive information from it.

### 4.3 Message Passing Software

To provide for the required message passing we have chosen to employ the Message Passing Interface (MPI, Skjellum et al., 1993) rather than using machine specific commands. This provides portability to the code as it will be able to run on any parallel machine and/or distributed network of machines on which this public domain library is available. To this point both forward and inverse codes have been implemented on the 1840 processor Intel Paragon at Sandia National Laboratories.

## 5 Demonstration of the FD Forward Solution

To illustrate the versatility and speed of the numerical solution when implemented on a parallel platform, we have simulated two different models which represent measurement configurations that

might be employed in the field and employ a wide range of frequencies. The first simulation will be involve frequencies in the low GPR range while the second will simulate a portion of helicopter EM survey. In the two cases the Krylov solver was assumed to have converged to an adequate error level when equation (5) was found to be less than or equal to  $10^{-7}$  and  $10^{-8}$ , respectively. These error levels are empirical and are based on extensive comparisons of the solution with other numerical solutions and scale model experiments (eg., Alumbaugh and Newman, 1994).

### 5.1 High Frequency Simulation for the 'VETEM' project

The 'VETEM' (Very Early Time ElectroMagnetic) project is an attempt to build an electromagnetic prospecting system that operates above traditional geophysical induction frequencies (100 kHz) yet below ground penetrating Radar frequencies ( 100 MHz) (Pellerin et al., 1995). To illustrate the ability of the code to simulate the electromagnetic response at these frequencies, variations of the model shown in Figure 4 has been employed. This example was designed to simulate a test site at the Colorado School of Mines where a prototype of the VETEM system known as the High Frequency Sounder (HFS) (Stewart et al., 1994) was first tested. The model is particularly difficult to simulate because of two conflicting conditions that are imposed by the material properties; 1) the wavelength in the block at 28.5 MHz is approximately 1.6m which requires a maximum cell dimension of 0.16m to avoid grid dispersion (Chew, 1990, p 244) and 2) the skin depth in the first layer at that same frequency is 17.8m which requires the boundaries to be placed very far away to avoid reflections off the grid. The small cell size coupled with the large distance to the boundaries produces a very large mesh if no absorbing boundary conditions are employed.

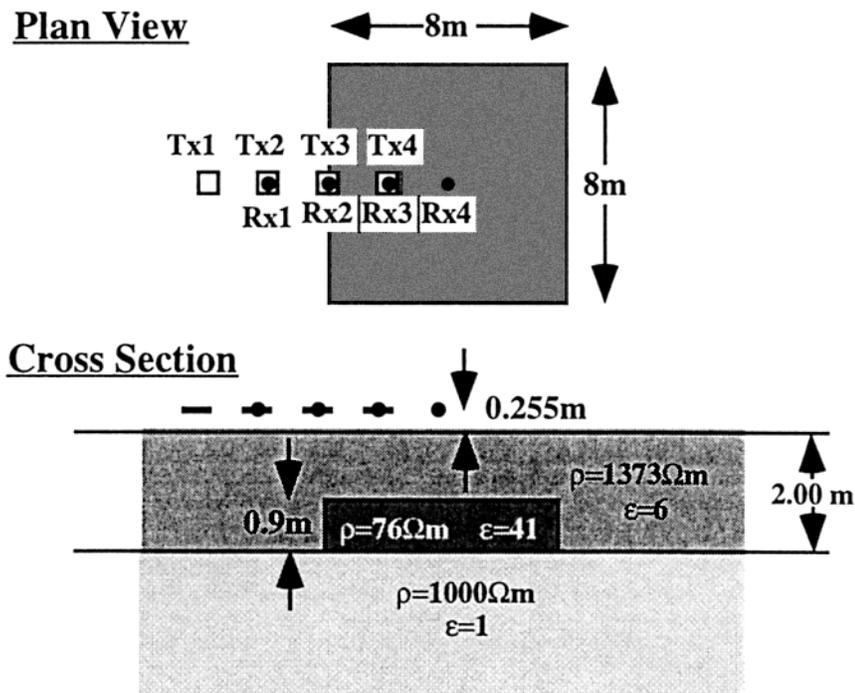


Figure 4: The Colorado School of Mines 3-D (CSM3D) model.

To simulate this example a 120 x 120 x 120 cell mesh was employed with a constant cell size of 0.15m in the x and y directions. This places the total distance across the mesh at 18m. In z, the maximum cell size was also 0.15m, with a minimum cell size of 0.13m to accommodate the layer thicknesses. Note, this mesh produces a total of  $5 \times 10^6$  unknowns for which to solve, which is much too large a problem for all but a supercomputer. The VMD source was placed at the center of the

mesh in  $x$  and  $y$ , i.e., 9m from each boundary, and a background conductivity of  $\sigma=10^{-16}$ S/m was assumed. To incorporate loss and thus avoid reflections off the edges of the mesh,  $b$  was set equal to 0.6 over 20 cells along each edge of the mesh.

In the first case we simulate two 1-D models, a two layer which assumes the block is absent, and a 3 layer model which assumes that the block extends to infinity in the  $x$  and  $y$  directions. This allows us to compare to a 1-D code developed by Ki Ha Lee at Lawrence Berkeley Laboratory. Because HFS directly measures tilt angle and ellipticity of the magnetic field (Smith and Ward, 1974) the results have been plotted in terms of these parameters rather than amplitude and phase of the different components. As can be seen in Figure 4a for the layered models, the 3-D code reproduces the 1-D calculations extremely well.

In Figure 5b the 3-D results for four different source positions are plotted with the results for the two 1-D models. Notice that the 3-D responses never reproduce the 1-D results even when the source-receivers are completely outside or within the block. This indicates that 3-D effects are measurable at greater distances than is immediately evident and that 1-D inversions probably would not accurately reproduce the structure of the subsurface.

To demonstrate some of the questions that must be answered when using the parallel machines, the solution time as well as the flop rate has been plotted against the number of processors employed for the 3 layer model at 10.1 MHz on the Intel Paragon. Figure 6 shows that a large decrease in run time occurs with an increasing number of processors from 200 up to 1000. This corresponds to solving for 24000 to 3000 unknowns per processor and indicates that the processors are spending the majority of their time performing calculations rather than communicating. However, the relatively small decrease in run time with increasing number of processors over 1000 indicates the solution time is beginning to be dominated by message passing if less than 3000 unknowns are being solved for on each processor. Thus we are left with a decision to make. If we wish to use the machine most efficiently, we would employ less than 1000 processors such that the internal computations are dominating the solution time. We could then run several jobs simultaneously such that the efficiency increases proportionally to the number of jobs. On the other hand if we desire as quick a turn around time as possible for a single computation, then we would want to operate near the right end of the curve.

## 5.2 Airborne Simulation

The second example simulates an helicopter EM survey flown to define the location of a buried paleo-channel through which conductive salt water is migrating, and is designed after a survey flown in Australia in the early 1990's (Doug Frazer, personnel communication). Figure 7 shows a plan view of the model at 5m depth below the earth's surface as well as two cross sections through the model. The flight lines are 30m above the earth's surface, are spaced at 200m intervals from top to bottom in Figure 7, and along each line the sampling interval is 100m. This yields a total of 187 source positions. A VMD source is operating at 0.9 KHz, 7.2 kHz and 56 kHz, with the receiver located 8m to the right of the source. The three frequencies coupled with the 187 positions yields a total of 561 forward solves.

To calculate this with the 3-D finite difference code, the earth and air were divided into a 208 x 184 x 49 cell grid which yields a total of  $5.6 \times 10^6$  unknowns for which to solve. To avoid reflections off the mesh boundaries normal grid stretching (i.e.  $b=0$ ) was employed to move them out to 400m from the nearest sampling point. The smallest cell size employed was 5m x 5m by 2.5m and was employed at the air-earth interface underneath each source array. The largest cell size employed was in the corners of the mesh and was 20m x 20m x 20m. A background conductivity of  $s=10^{-16}$ S/m was assumed to simulate the electrical properties of the air.

Notice in Figure 8 that for all three frequencies the channel is clearly defined, although its resistivity is not as accurately defined at higher frequencies. This is due to the increased sensitivity

at higher frequencies to the surface resistive layer. To run this model on 1360 processors of the Intel Paragon took approximately two days. This reasonably quick turn around time for this complicated model illustrates the utility of these machines for solving realistic geologic problems. In addition, although this solution may of taken too much time to be employed for inversion of this large of a model, the near future development of machines with tens-of-thousands of processors should allow for it.

## 6 3-D Data Inversion – Synthetic Example

Figure 9 shows a model used to test the 3-D inverse. The data from this model were generated from the integral equation solution of Newman et al. (1986) and provides a stronger check on the inversion scheme than using data generated by the staggered finite difference code; using data generated with the same forward code as embedded in the inverse will be prone to the same numerical errors and thus will not be a fully independent check. The test model consists of a 0.2 S/m cube, 50 m on a side, residing in a 0.005 S/m background. Eight wells surround the target, each containing 15 vertical magnetic dipole (VMD) transmitters at 10 m intervals straddling the target. The vertical magnetic fields were calculated in all other wells, excluding the transmitter well at 10 m intervals. Because the frequency of excitation used in this test is only 20 kHz, the dielectric properties of the target and host are not important in the simulation and only the conductivity properties need be estimated; the magnetic permeability is assumed constant and set to free space throughout the model.

Gaussian noise equal to two percent of the data amplitude were added to each data point. The data were then weighted by the noise, before inversion. In total, they comprise a 12 600 transmitter-receiver pairs. The inversion domain consist of 29 791 cells, but only 13 824 cells are shown in the interwell region in Figure 10; cells outside this region are used to keep the boundary of the inversion domain at distance so as to not affect the conductivity estimates in the interwell region. The inversion has recovered fairly well the location and geometry of the cube, but a smeared version of its conductivity; the estimates vary from 0.1 to 1 S/m. The conductivity estimates of the background range as low as 0.0014 S/m. Improved estimates on the background can be obtained by tightening the lower bound positivity constraint. In this example, the conductivity estimates were restricted to be greater than 0.001 S/m.

Ten iterations were needed to obtain this reconstruction, where the reduction in relative error against iteration count is illustrated in Figure 11. Assuming Gaussian noise with zero mean the inversion is assumed to have converged when the relative error approaches the value of one. Because the final error level is still above one in Figure 11 this might suggest that more information could be extracted from the data. However, we ascribe the final error level to originate from bias in the data caused by using a different forward modeling algorithm than the one used in the inverse. Finally the processing time needed to produce the image in Figure 10 was approximately 21 hours on the Paragon, with 512 processors utilized.

## 7 Conclusions and Discussion

In this paper we have presented a schemes to model and invert frequency domain electromagnetic response of a 3-D earth over a wide band of frequencies using massively parallel computers. The difficulties associated with providing absorbing boundary conditions for the forward solution and porting the serial versions of both the forward and inverse problems to a parallel machine have been outlined. Two simulations have been included to demonstrate the versatility of the forward code. Further, the inversion code has been implemented such that reconstructions can be produced that

do not underparameterize the earth; these reconstructions involve tens of thousands of cells. Since the 3-D MP inverse also includes rigorous 3-D forward modeling for computing model sensitivities and predicted data, it is our hope that this solution will also serve as an accuracy benchmark on approximate inverse methods now being implemented on workstation platforms. Such solutions are beginning to see widespread use in the EM geophysical community (cf. Torres-Verdin and Habashy, 1995 and 1994; Zhandnov and Fang, 1995).

Although the demonstrations shown here would have been nearly impossible prior to the parallel implementation due to the size of the models and/or the number of frequencies and sources involved, we believe that there is still much research to be done with regards to the implementation of these types of schemes. A notable location for improvement is in the area of preconditioners for the forward problem, which will correspondingly impact the solution time for the inverse problem. Here we are currently considering the use of multigrid preconditioners, and methods to separately treat the real and imaginary components of the matrix system. In addition, a scheme to accelerate the convergence for very low frequency simulations where channeling currents dominate needs to be developed in order to simulate natural field measurements as well as extend the frequency band down below 100 Hz; Smith (1992) has found that a static correction can be incorporated to accommodate this.

Finally, better ways are needed to manage the memory needed to carry out a 3-D inverse on large data sets. Since electric field vectors need to be computed and stored in memory for all the different transmitters and receivers this will eventually limit the model size that can be inverted. To overcome this limitation, we are currently investigating methods that skeletonize the inversion domain, but still retain the fine parameterization level in the forward solves for accuracy. Although we are seeking smooth images, we can argue that a coarser inversion grid is acceptable, provided it is not too coarse and ample smoothing applied in the inversion. With the coarser grid, the electric field vectors needed in the inverse will be interpolated from a fine grid based on the forward modeling to a sparser grid as needed in the inversion, thus reducing the storage overhead of the electric field vectors and increasing dramatically the amount of data that can be inverted. Consider a problem where the inversion grid is eight times coarser than the forward modeling grid. If  $120^3$  nodes are used in the forward calculations, the skeletonize inversion grid which still comprises 216 000 cells, allows the number of transmitter and receivers to increase from 700 to over 3000 on the Intel Paragon.

## 8 Acknowledgements

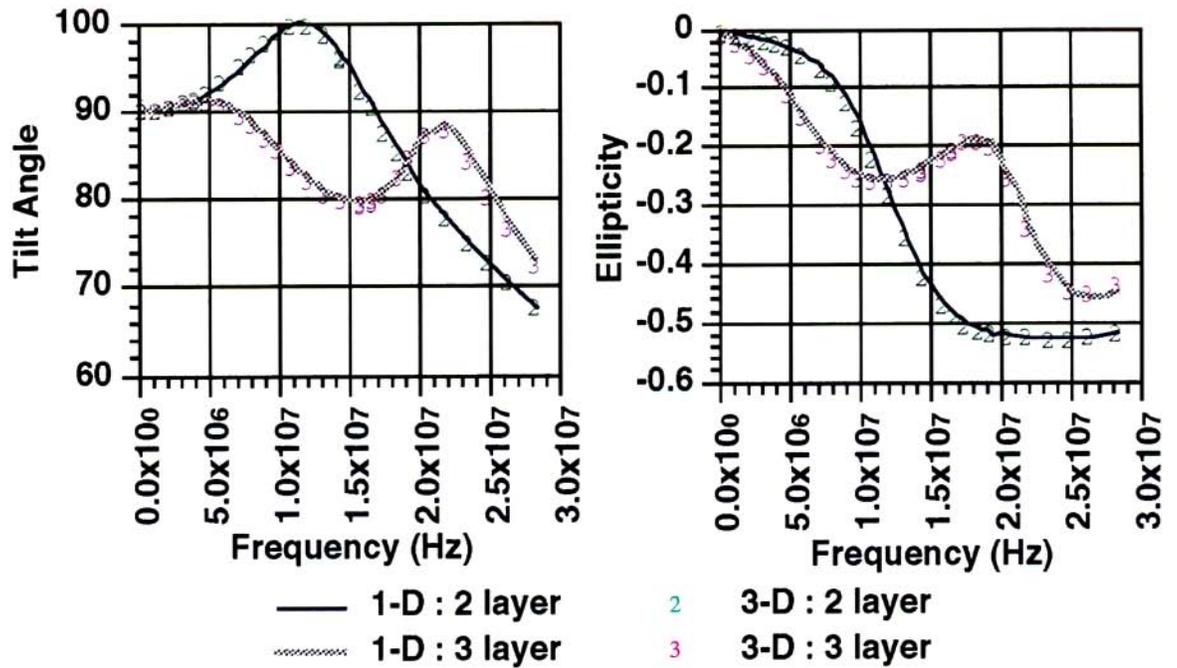
We express our thanks to Dr. Ki Ha Lee of Lawrence Berkeley Laboratory for the use of the 1-D layered earth code used in the model comparisons. This work was performed at Sandia National Laboratories, which is operated for the U.S. Department of Energy. Funding for this project was provided by 1) DOE's office of Basic Energy Sciences, Division of Engineering and Geoscience under contract DE-AC04-94AL85000, 2) the Sandia National Laboratory Director's Research funds and 3) the VETEM project which is supported by the DOE Office of Technology and Development.

## References

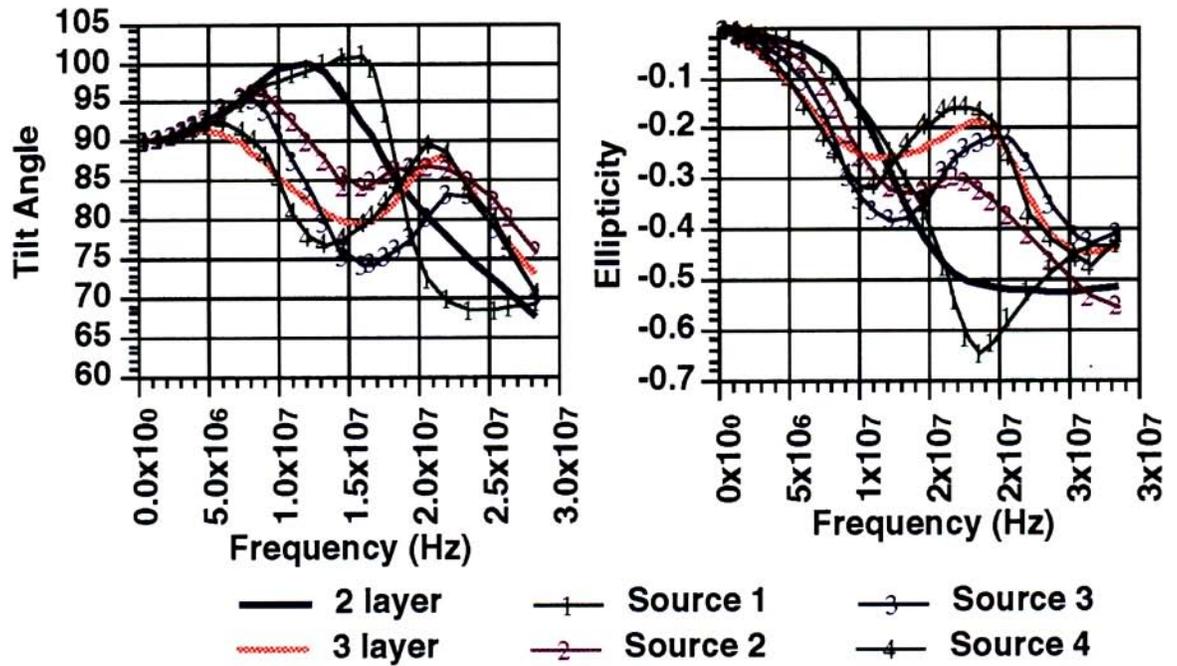
- Alumbaugh, D. L. and Newman G. A., 1994, Fast, frequency domain electromagnetic modeling using finite differences: 64th Ann. Intern. Mtg., Soc. Explor. Geophys., Los Angeles CA, Expanded Abstracts, 369-373.
- Alumbaugh, D.L., and Newman, G.A., Prevost, L., and Shadid, J. N., 1996, Three dimensional wide band electromagnetic modeling on massively parallel computers: Radio Sci.

- Berenger, J., 1993, A perfectly matched layer for the absorption of electromagnetic waves: *J. Comp. Phys.*, **114**, 185-200.
- Chew, W. C., 1990, *Waves and Fields in Inhomogeneous Media*: Van Nostrand Reinhold International Company Limited.
- Chew, W.C., and Weedon, W.H., 1994, A 3-D perfectly matched medium from modified Maxwell's equations with stretched coordinates: *Microwave and Optical Tech. Let.*, **7**, 599-604.
- Druskin, V. and Knizhnerman, L., 1988, A spectral semi-discrete method for the numerical solution of three-dimensional nonstationary problems of electric prospecting: *Izvestiya, Earth Physics*, **24**, 641-648.
- Druskin, V. and Knizhnerman, L., 1994, A spectral approach to solving three-dimensional diffusion Maxwell's equations in the time and frequency domains: *Radio Science*, **29**, 937-953.
- Freund, R., 1992, Conjugate gradient type methods for linear systems with complex symmetric coefficient matrices: *SIAM J. Sci. Statist. Comput.*, **13**, 425-448.
- Hestenes, M. R. and Stiefel, E., 1952, Methods of conjugate gradients for solving linear systems: *J. Res. Nat. Bur. Standards*, **49**, 409-435.
- Katz, D.S., Thiele, E.T., and Taflove, A., 1994, Validation and extension to three dimensions of the Berenger PML absorbing boundary condition for FD-TD meshes: *IEEE Micro. and Guided Wave Let.*, **4**, 268-270.
- McGillivray, P. R., and Oldenburg, D. W., 1990, Methods for calculating Frechet derivatives and sensitivities for the non-linear inverse problem: *Geophys. Prosp.*, **38**, 499-524.
- Mackie, R. L., and Madden, T. R., 1993, Three-dimensional magnetotelluric inversion using conjugate gradients: *Geophys. J. Int.*, **115**, 215-229.
- Newman, G. A., 1995, Cross well electromagnetic inversion using integral and differential equations: *Geophysics*, **60**, 899- 910.
- Newman G. A., and Alumbaugh, D. L., 1995, Frequency-domain modeling of airborne electromagnetic responses using staggered finite differences: *Geophys. Prosp.*, **43**, 1021-1042.
- Newman, G. A., Hohmann, G. W. and Anderson, W. L., 1986, Transient electromagnetic response of the three-dimensional body in a layered earth: *Geophysics*, **51**, 1606-1627.
- Oldenburg, D. W., 1990, Inversion of electromagnetic data: an overview of new techniques: *Surveys in Geophysics*, **11**, 231-270.
- Park, S. K., 1983, Three-dimensional magnetotelluric modeling and inversion: PhD. thesis, Mass. Inst. Tech.
- Pellerin, L., Labson, V. F., and Pfeifer, M. C., 1995, VETEM - a very early time electromagnetic system: presented at SAGEEP '93 (Symposium on the Application of Geophysics to Environmental and Engineering Problems ), Orlando, Florida.
- Skjellum, A., Doss, N.E., and Bangalore, P.V., 1993, Writing libraries in MPI, in A. Skjellum and D. Reese, Ed., *Proceedings of the Scalable Parallel Libraries Conference*, IEEE Comput. Sci. Press, 166-173.
- Smith, T. J., 1992, Conservative modeling of 3-D electromagnetic fields: International Association of Geomagnetism and Aeronomy, 11th Workshop on Electromagnetic Induction in the Earth, Wellington, New Zealand, Meeting Abstracts.
- Smith, B. D., and Ward, S. H., 1974, On the computation of polarization ellipse parameters: *Geophysics*, **39**, 867-869.
- Stewart, D.C., Anderson, W.L., Grover, T.P., and Labson, V.F., 1994, Shallow subsurface mapping by electromagnetic sounding in the 300 KHz to 30 MHz range: model studies and prototype system assessment: *Geophysics*, **59**, 1201-1210.
- Tikhonov, A. N. and Arsenin, V. Y., 1977, *Solutions to ill- posed problems*: John Wiley and Sons, Inc.

- Torres-Verdin, C. and Habashy, T. M., 1995, A two step linear inversion of two dimensional electrical conductivity: *IEEE Trans. Antenna Propagat.*, **43**, 405-415.
- Torres-Verdin, C. and Habashy, T. M., 1994, Rapid 2.5-dimensional forward modeling and inversion via a new nonlinear scattering approximation: *Radio Sci.*, **29**, 1051-1079.
- Wang T. and Hohmann, G. W., 1993, A finite difference time-domain solution for three-dimensional electromagnetic modeling: *Geophysics*, **58**, 797-809.
- Yee, K. S., 1966, Numerical solution of initial boundary problems involving Maxwell's equations in isotropic media: *Inst. of Electr. and Electron. Eng., Trans. Ant. Prop*, **AP-14**, 302-309.
- Zhdanov, M. S. and Fang, S., 1995, 3D electromagnetic inversion based on the quasi-linear approximation: Submitted to *Geophysics*.
- Zhang, J. Mackie, R. L. and Madden, T. R., 1995, Three-dimensional resistivity forward modeling and inversion using conjugate gradients: *Geophysics*, **60**, 1313-1325.



(a)



(b)

Figure 5: Results for the CSM3D model. a) 1-D comparisons for 2 and 3 layer models. b) 3-D comparison for different source positions.

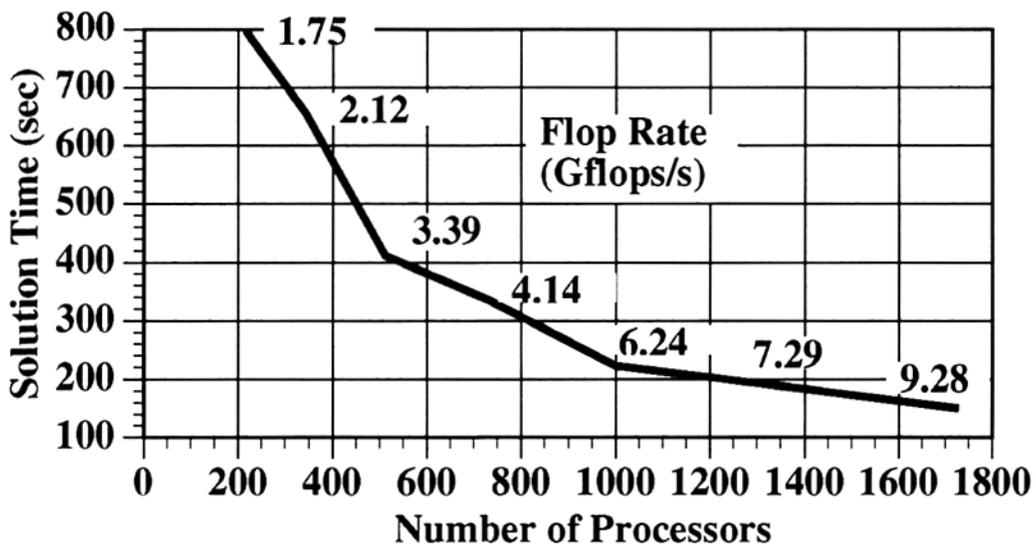


Figure 6: Run time versus number of processors employed for the 3 layer model at 10.1 MHz.

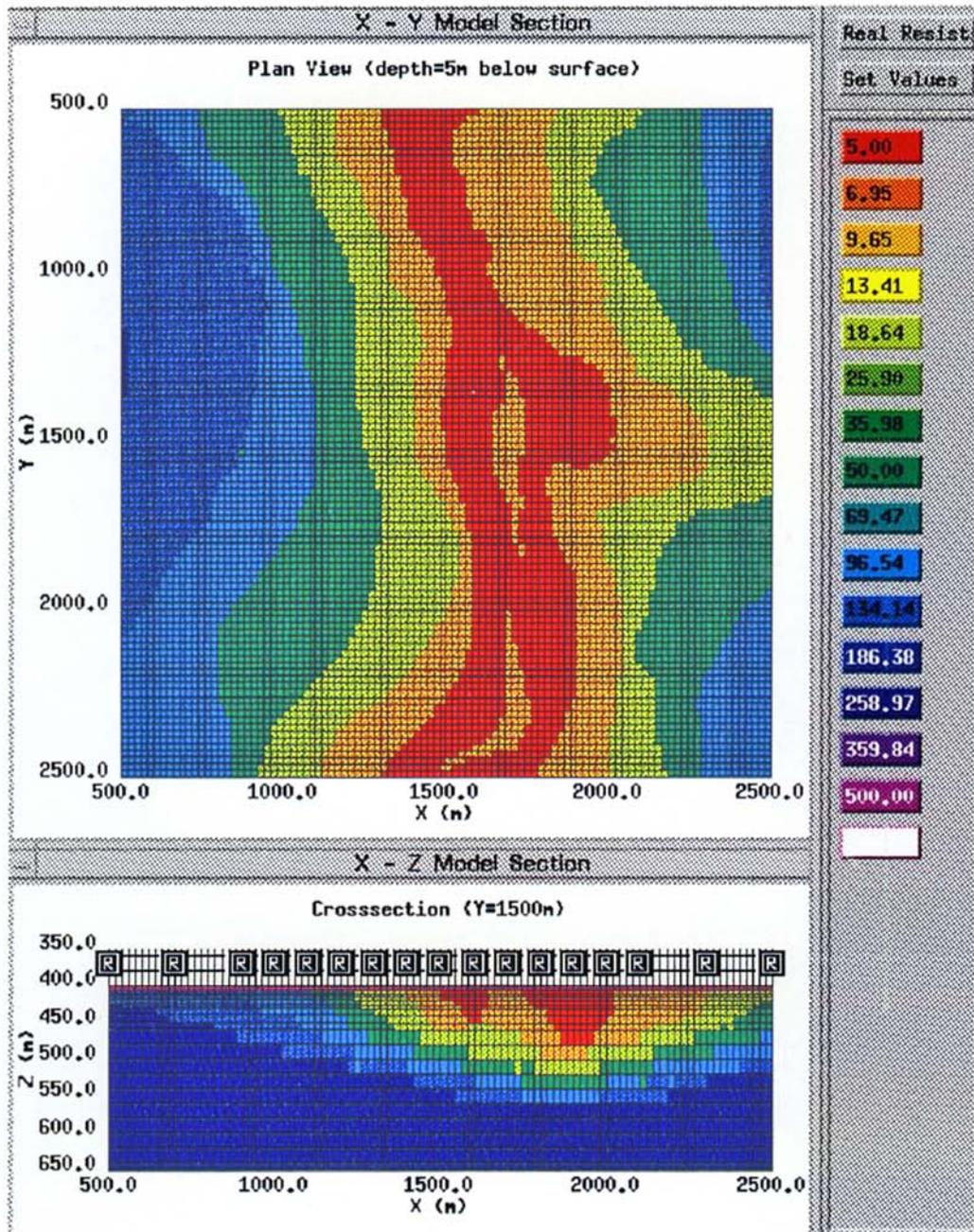


Figure 7: The subsurface channel model employed for the airborne simulation. The upper figure is a plan view at the top of the channel, and the bottom figure is a vertically exaggerated cross-section at  $y=1500$  m. Although it is difficult to see, a 5 m thick-  $500 \Omega\text{m}$  layer exists from the earth's surface down to the top of the channel. The gray scale varies logarithmically from  $5 \Omega\text{m}$  (black) to  $697 \Omega\text{m}$  (white).

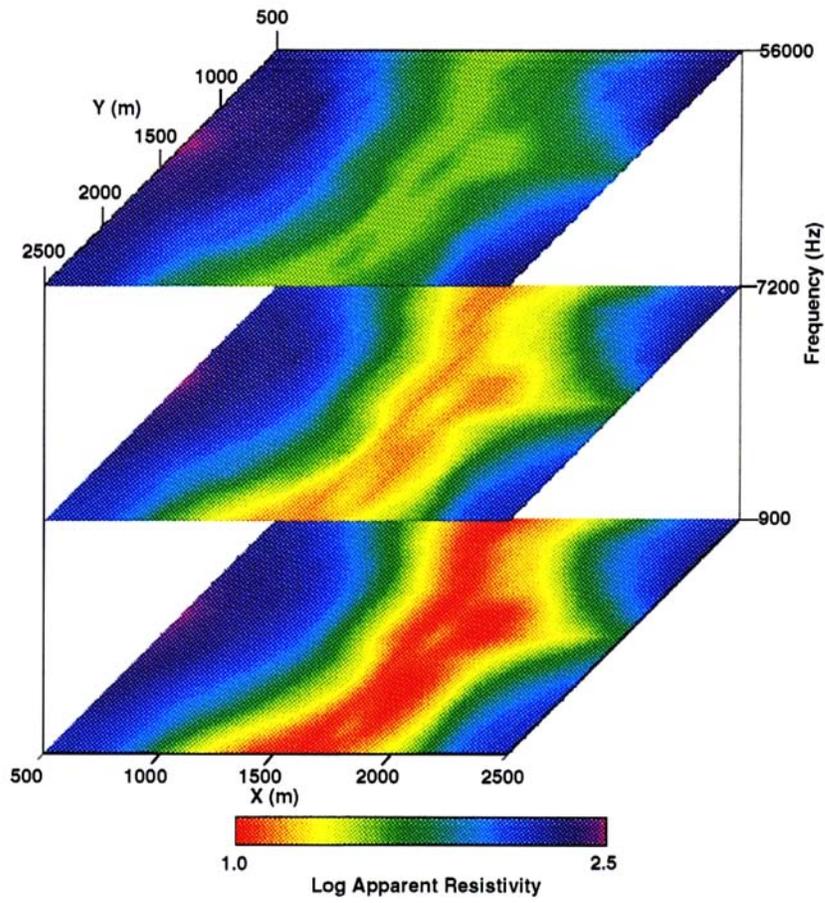


Figure 8: Calculated apparent resistivities for the subsurface channel model.

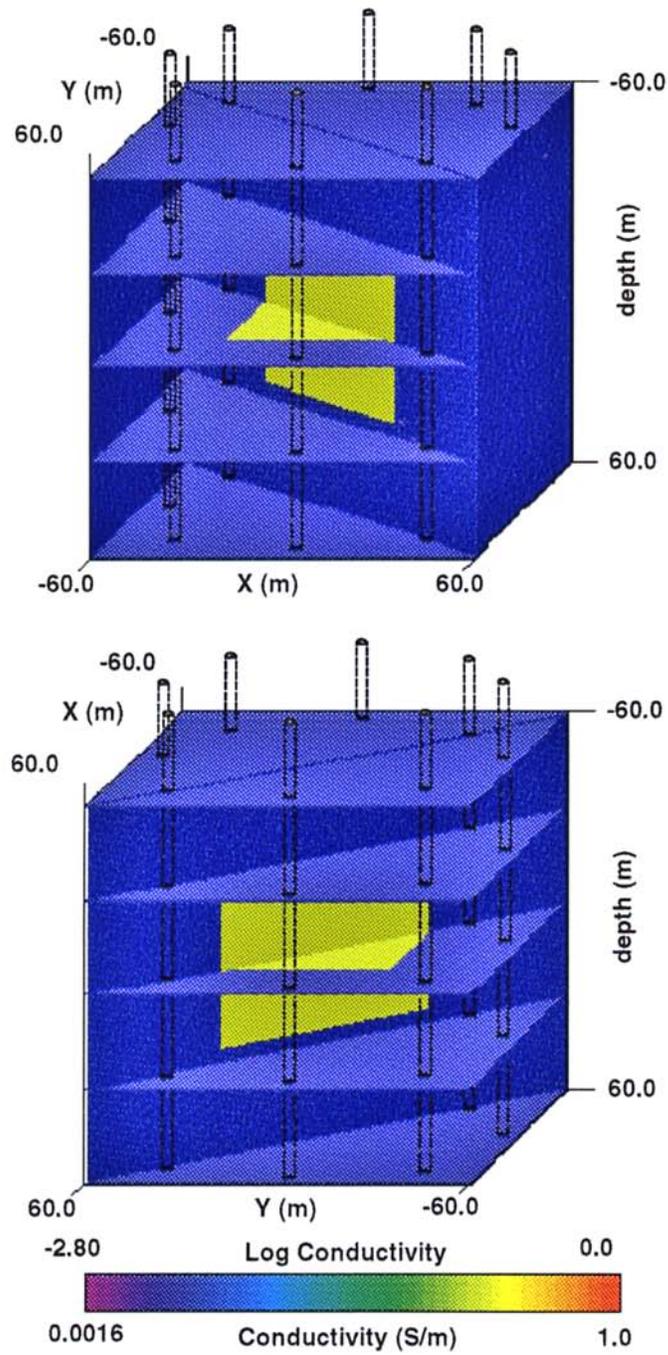


Figure 9: Synthetic example, with wellbores, used to test the inversion algorithm. The data were calculated from this model using an integral equation solution. Different slices of the model are shown from different perspectives.

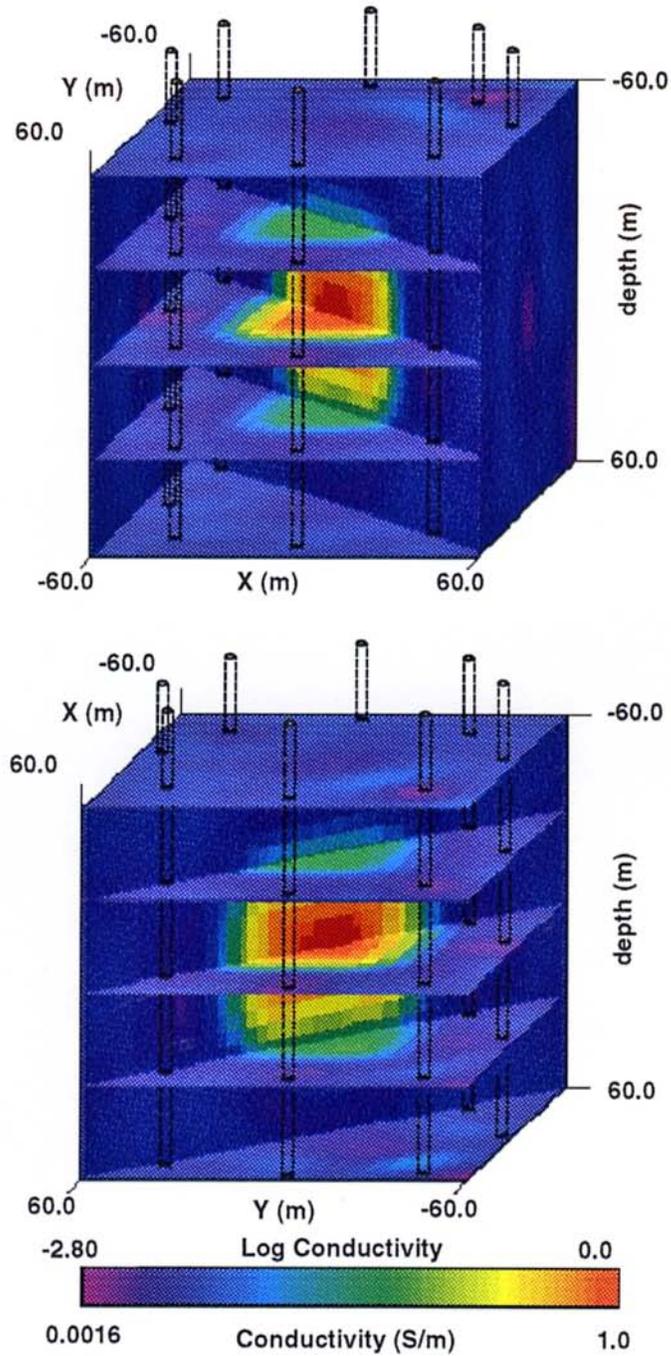


Figure 10: Reconstructed log conductivity and resistivity for the synthetic example illustrated in Figure 9 for different slices from two different perspectives. The wellbores used in the simulation are also indicated.

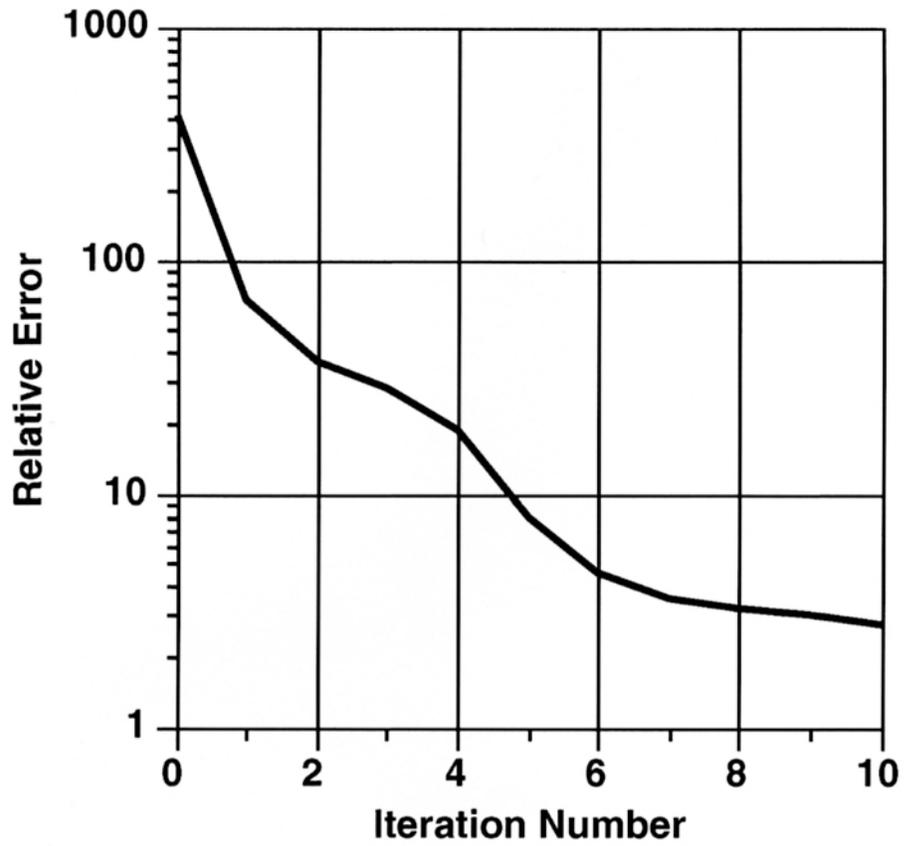


Figure 11: Sum of squared error plotted against iteration number for the 0.2 S/m test body shown in Figure 9. The squared error has been normalized by the variance of the noise.