SANDIA REPORT

4-1-96

RECEIVED
APR 0 8 1996
OSTI

Three-Dimensional Electromagnetic Modeling and Inversion on Massively Parallel Computers

Gregory A. Newman, David L. Alumbaugh

Approved for public release; distribution is unlimited.

MASTER

DISCLAIMER

Portions of this document may be illegible
in electronic image products. Images are
produced from the best available original
document.

THREE-DIMENSIONAL ELECTROMAGNETIC MODELING AND INVERSION ON MASSIVELY PARALLEL COMPUTERS

Gregory A. Newman
David L. Alumbaugh

Geophysics Department
Sandia National Laboratories
Albuquerque, NM 87185-0750

ABSTRACT

A method is presented for modeling the wide-band, frequency domain electromagnetic (EM) response of a three dimensional (3-D) earth to dipole sources operating at frequencies where EM diffusion dominates the response ($<$ 100 kHz) up into the range where propagation dominates ($>$ 10 MHz). The scheme employs the modified form of the vector Helmholtz equation for the scattered electric fields to model variations in electrical conductivity, dielectric permittivity and magnetic permeability. The use of the modified form of the Helmholtz equation allows for perfectly matched layer (PML) absorbing boundary conditions to be employed through the use of complex grid stretching. Applying the finite difference operator to the modified Helmholtz equation produces a linear system of equations for which the matrix is sparse and complex symmetric. The solution is obtained using either the bi-conjugate gradient (BICG) or quasi-minimum residual (QMR) methods with preconditioning. At this point four different

I

preconditioners have been successfully implemented; Jacobi scaling, Block Jacobi, and both a Neumann series and least squares polynomial. The most stable convergence properties are offered by the QMR method, while least squares polynomial preconditioning produces the greatest acceleration in convergence. In order to simulate larger, more realistic models than has been previously possible, the scheme has been modified to run on massively parallel (MP) computer architectures. Comparisons of the 1840 processor Intel Paragon to a high end workstation (an IBM RS6000-590 computer) indicates at least a two order of magnitude increase in both computational speed and maximum model size that can be simulated by moving to the MP machine. Three different geologic models are simulated to demonstrate the use of the code for frequencies ranging from 100 Hz to 30 MHz and for different source types and polarizations. The simulations show that the scheme is correctly able to model the air-earth interface and the jump in the electric current normal to discontinuities. For frequencies greater than 10 MHz, complex grid stretching must be employed to incorporate absorbing boundaries while below this normal (real) grid stretching can be employed.

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 for controlled sources is employed to compute model sensitivities and predicted data in the frequency domain with the 3-D finite difference algorithm. Necessity dictates that the inverse be underdetermined since realistic reconstructions require the solution for tens of thousands of parameters. In addition, large scale 3-D forward modeling is required and this can easily involve the solution of over several million electric-field unknowns per solve. A massively parallel computing platform has therefore been utilized for reasonable execution times and results are given for the 1840 node Intel Paragon. The solution has been demonstrated with a synthetic example with added Gaussian noise, where the data were produced from an integral equation forward modeling code and is different from the one embedded in the inversion algorithm.

The three-dimensional electromagnetic inversion scheme has been successfully applied to crosswell electromagnetic data collected at the Richmond Field Station near Berkeley California. By comparing images of data collected before and after the injection of 50 000 gallons of salt water, a three dimensional image of the plume has been developed which shows the location of zones of maximum permeability surrounding the injection well through which the

II

saltwater has migrated. A resolution analysis has determined that the location of the plume is fairly accurate. However the image of the geology will be distorted due to incomplete data coverage. This latter problem is further complicated by the fact that data residuals indicate that the wells are deviated from the vertical. These deviations have been demonstrated to cause artifacts within the images and thus further reducing the accuracy of the images with respect to the geology.

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

CHAPTER I

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 limitation posed by serial computers are 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 in this report is our approach to solving the 3-D forward and inverse problems on an MP platform. For the forward problem we will examine in Chapter II the implementation of a frequency-domain-finite difference (FD-FD) scheme based on a staggered grid. Building on

1

this we will implement in Chapter III the corresponding inverse. Because the ultimate goal of any inversion scheme is to use it to image field data, in Chapter IV we demonstrate how this scheme can be used to design a 3-D crosswell survey, and then invert a crosswell data set collected at the Richmond field station north of Berkeley California. Images before and after the injection of a salt water plume will be compared to determine the location of the injected plume. In addition, we will also show how the scheme can be employed to analyze the reliability of the images as well as the accuracy and errors in the data.

CHAPTER II

THREE-DIMENSIONAL WIDE BAND ELECTROMAGNETIC MODELING ON MASSIVELY PARALLEL COMPUTERS

The recent advances in the forward modeling of electromagentic fields have been primarily due to the rapid improvements in computer speed and memory. For compact bodies residing in a layered earth, integral equation (IE) solutions (Tripp and Hohmann, 1984, Newman et al., 1986, Xiong, 1992) offer the most efficient way of producing a solution. However, as the complexity and size of the model grows, IE methods become numerically cumbersome. This is due to the fact that a full matrix on the order of 3N needs to be solved, where N is the number of cells representing the structure. The IE method is too time consuming for general models because the solution time for the matrix goes as $(3N)^3$.

Recently, a series of papers have appeared that employ various approximations to the integral equations governing the EM fields (Habashy et al, 1993; Torres-Verdin and Habashy, 1994; and Zhdanov and Fang, 1995b). For general models these methods are much quicker than the full IE solution because they do not require the inversion of a large, dense matrix. However, the approximations often exhibit limitations in terms of the frequency that can be employed and the conductivity contrasts that can be simulated within the model. In addition, these types of approximations have not been implemented in the radar range (> 10 MHz) where wave propagation becomes important. Thus to ensure accurate calculation of the EM fields for general models over a wide range of frequencies and material contrasts, a differential equation solution needs to be employed.

Two different forms of differential equation solutions exist: finite difference (FD) methods and finite element (FE) methods. The advantage of these techniques over integral equations is that a linear system of equations results in which the matrix is very sparse. This property allows for the implementation of iterative Krylov subspace methods to solve the system which are much more efficient than direct matrix inversion. Although finite element techniques are more flexible in terms of the geometry of the mesh that can be employed, we have chosen to employ a FD scheme. With a FD scheme a well structured grid results which is acceptable for the models we plan to consider.

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. However, with the development of massively parallel machines, the rate at which the simulations can proceed has dramatically increased because thousands of processors can operate on the problem simultaneously. In this chapter we examine the implementation of a FD scheme on a parallel platform and demonstrate its usefulness over a wide frequency range for different types of geophysical scenarios. In additional we will examine different types of Krylov

3

solvers and illustrate how the choice of the proper solver can have a dramatic impact on the solution time.

THEORETICAL DEVELOPMENT

In order to simulate the EM response of a 3-D earth, we numerically solve the frequency domain version of Maxwell's equations using a finite difference approximation on a staggered grid (defined below) as outlined by Yee (1966). The use of this type of grid has recently experienced a gain in popularity in EM geophysical applications; Druskin and Knizherman (1988 and 1994), Smith (1992), Wang and Hohmann (1993), Newman (1995) and Newman and Alumbaugh (1995) all employ some type of staggered finite difference grid to solve for the EM fields in both the time and/or frequency domain. As Smith has noted, a staggered finite difference grid is a natural for solving such problems because it explicitly enforces the conservation of electric current and a divergence free magnetic field. These are necessary auxiliary conditions that must be satisfied when modeling the electromagnetic field.

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 an extension of the one outlined in Newman and Alumbaugh (1995) with the major theoretical differences being 1) the incorporation of absorbing boundary conditions (ABCs), and 2) the ability to model variations in magnetic permeability. The ABCs are required to simulate the response of frequencies greater that 10 MHz where wave propagation becomes dominant over lower frequency EM diffusion 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 Maxwell's equations in which the absorption is incorporated through the use of complex grid stretching. Here we discuss the theoretical development of the finite difference solution and leave the discussion of the reasons for using the PML and its properties for a later section and Appendix A.

Assuming a time harmonic dependence of $e^{i\omega t}$ where $i = \text{sqrt}(-1)$, the modified-frequency domain form of Maxwell's equations are given by Chew and Weedon (1994) as

$$\nabla_e \times \mathbf{E}^t = -i\omega\mu\mathbf{H}^t - i\omega\mathbf{M}^p \tag{1}$$

and

$$\nabla_h \times \mathbf{H}^t = (\sigma + i\omega\varepsilon)\mathbf{E}^t + \mathbf{J}^p \tag{2}$$

4

where

$$\nabla_e = \hat{\mathbf{i}} \frac{1}{e_x} \frac{\partial}{\partial x} + \hat{\mathbf{j}} \frac{1}{e_y} \frac{\partial}{\partial y} + \hat{\mathbf{k}} \frac{1}{e_z} \frac{\partial}{\partial z} \qquad (3)$$

and

$$\nabla_h = \hat{\mathbf{i}} \frac{1}{h_x} \frac{\partial}{\partial x} + \hat{\mathbf{j}} \frac{1}{h_y} \frac{\partial}{\partial y} + \hat{\mathbf{k}} \frac{1}{h_z} \frac{\partial}{\partial z}. \qquad (4)$$

In these expressions the electrical conductivity , magnetic permeability and dielectric permittivity are denoted by σ, μ and ε, respectively, the total electric and magnetic fields are defined by the vectors, $\mathbf{E}^t$ and $\mathbf{H}^t$, respectively, and the source vectors, $\mathbf{J}^p$ and $\mathbf{M}^p$, are current densities for impressed electric and magnetic sources. In addition 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 the Appendix A, when e_i and h_i are complex then the medium is perfectly absorbing and the degree of absorption is independent of the angle of incidence.

Because the fields are rapidly varying near the source, a finite difference solution for the total field requires the grid to be finely discretized in this region. Matters are further complicated near the receivers because if too coarse a mesh is used in these locations, the numerical extrapolation and differencing required for accurate output will be prone to inaccuracies. It is easy to comprehend that employing a very fine grid near the source and receiver limits the size of the model to be calculated due to storage overhead, especially if results are going to be calculated for multiple source positions and frequencies. To avoid these problems we have chosen to work with the scattered field versions of the modified Maxwell's equations (see Appendix B for the derivation) which have the form

$$\nabla_e \times \mathbf{E}^s = -i\omega\mu\mathbf{H}^s + \left(\mu - \mu_p\right)\mathbf{H}^p \qquad (5)$$

and

$$\nabla_h \times \mathbf{H}^s = (\sigma + i\omega\varepsilon)\mathbf{E}^s + \left[\left(\sigma - \sigma_p\right) + i\omega\left(\varepsilon - \varepsilon_p\right)\right]\mathbf{E}^p. \qquad (6)$$

In these equations $\mathbf{E}^s$ and $\mathbf{E}^p$ are the scattered and primary electric fields ($\mathbf{E}^t = \mathbf{E}^p + \mathbf{E}^s$), and $\mathbf{H}^s$ and $\mathbf{H}^p$ are the associated magnetic fields with $\mathbf{H}^t = \mathbf{H}^p + \mathbf{H}^s$. Note that $\mathbf{M}^p$ and $\mathbf{J}^p$ have now been replaced by the equivalent sources $\left(\mu - \mu_p\right)\mathbf{H}^p$ and $\left[\left(\sigma - \sigma_p\right) + i\omega\left(\varepsilon - \varepsilon_p\right)\right]\mathbf{E}^p$, respectively where the 'p' superscript designates a primary or background field, and the 'p' subscript the value of the background material properties. These background values can be for that of a whole space, a

5

layered half-space, or some previously run model for which the results were saved to be incorporated as the primary field. For all of the examples presented here, a whole space background has been assumed.

The benefit of this formulation is that if the source is removed from zones of anomalous conductivity, then this equivalent source exhibits a smoother spatial dependence than that of an impressed dipole. Thus the fields are better behaved in the vicinity of the transmitter. It must be mentioned, however, that if the source is within a region of anomalous properties, then problems can arise due to the rapid variation of the primary field within the equivalent source terms. This phenomenon will be addressed below.

Employing the standard finite difference approximations and solvers described below, Newman (1995) has found that solving the coupled equations (5) and (6) works extremely well when both the source and receiver are buried fairly deep within the earth, for example in a crosswell simulation. Unfortunately, when the solution of the fields in the air above the earth is required at frequencies where the dielectric term is unimportant ($\omega \leq 2\pi \times 10$ Mhz), expressions 5 and 6 numerically uncouple because $\sigma + i\omega\varepsilon \approx 0$. In our experience this makes the solution of these first order equations either slowly or non-convergent due to ill conditioning of the matrix. To alleviate this problem, we can work with the vector Helmholtz equation for the scattered electric fields:

$$\nabla_h \times \frac{\mu_p}{\mu} \nabla_e \times \mathbf{E^s} = -i\omega\mu_p(\sigma + i\omega\varepsilon)\mathbf{E^s} - i\omega\mu_p\big[(\sigma - \sigma_p) + i\omega(\varepsilon - \varepsilon_p)\big]\mathbf{E^p} \tag{7}$$

$$-i\omega\mu_p\nabla \times \left[\left(\frac{\mu - \mu_p}{\mu}\right)\mathbf{H^p}\right]$$

This formulation, which is derived in Appendix B, is numerically easier to solve than the coupled equations. In addition, this form of the equation produces a linear system of equations in which the matrix is complex symmetric.

NUMERICAL SOLUTION

The Finite Difference Approximation

To cast expression (7) into a system of linear equations, we discretize the earth (and the air above) into a mesh of rectangular blocks. Again, because we are employing the scattered field formulation, this grid does not need to be finely discretized near the source unless it is located within or near a region of anomalous electromagnetic parameters. To solve for $\mathbf{E^s}$ as given in equation (7) we employ a staggered grid (Yee, 1966) for each component of the field as shown in Figure 1b. For a single cell this corresponds to sampling the electric fields along the edges of the cell, and the magnetic fields on the faces (Figure 1a).

The above formulation requires that the admittivity, defined as $\hat{y} = \sigma + i\omega\varepsilon$, be computed halfway along a given cell edge in Figure 1. Wang and Hohmann (1993) showed that for low frequency calculations, an average conductivity can be evaluated by tracing out a line integral of the magnetic field centered on the midpoint of the cell edge. Using their formulation, the resulting admittivity is simply the weighted sum of the admittivities of the four adjoining cells. For example, in Figure 2 the averaged admittivity in the y direction at $(i+1, j+\frac{1}{2},k+1)$ is given by

$$\hat{y}_{avg} = \frac{\hat{y}_{i,j+\frac{1}{2},k} A_{i,j+\frac{1}{2},k} + \hat{y}_{i+1,j+\frac{1}{2},k} A_{i+1,j+\frac{1}{2},k} + \hat{y}_{i,j+\frac{1}{2},k+1} A_{i,j+\frac{1}{2},k+1} + \hat{y}_{i+1,j+\frac{1}{2},k+1} A_{i+1,j+\frac{1}{2},k+1}}{A_{i,j+\frac{1}{2},k} + A_{i+1,j+\frac{1}{2},k} + A_{i,j+\frac{1}{2},k+1} + A_{i+1,j+\frac{1}{2},k+1}}$$

(8)

where $A_{i,j+\frac{1}{2},k}$ is a weighting function that is based on the cross-sectional area of each cell that is bounded by a line integral. This scheme is a simple application of Ampere's law.

Similarly, the magnetic permeability is averaged across the cell faces to correspond with the location of the magnetic fields. A simple relation can be derived, where this permeability is a geometric average of the two permeabilities of the neighboring cells such that the normal component of the magnetic induction, $\mathbf{B}=\mu\mathbf{H}$, is continuous. For example, in Figure 2 the average permeability corresponding to H_x at the face $(i+1,j+1/2,k+1/2)$ uses the permeability properties of cells $(i+1,j,k)$ and (i,j,k), and is given by

$$\mu_{avg} = \frac{\left(x_{i+3/2} - x_{i+1/2}\right)\mu_{i+1,j,k}\mu_{i,j,k}}{\left(x_{i+3/2} - x_{i+1}\right)\mu_{i,j,k} + \left(x_{i+1} - x_{i+1/2}\right)\mu_{i+1,j,k}}.$$

(9)

Notice in equation (7) that if the first term of the right hand side is moved to the left side, then it contains both the derivatives as well as the unknown scattered field values, while the right hand side consists of the known equivalent source values. Thus we can set up a system of equations which are written in matrix form as

$$\mathbf{KE_s} = \mathbf{s}.$$

(10)

where $\mathbf{K}$ is an NXN complex-symmetric matrix containing the numerical approximation of the spatial derivatives in equation (7) as well as the model electrical properties, $\mathbf{s}$ is the primary field

source vector and $\mathbf{E_s}$ is the scattered field vector we wish to solve for. To assemble a linear system of equations, we employ Dirichlet boundary conditions with the tangential electric fields at the edges of the grid set equal to zero. Unwanted reflections are avoided by either real or complex grid stretching as described below. The explicit formulation for the finite difference solution is given in Appendix C.

The solution vector is obtained by using preconditioned Krylov subspace techniques. These methods are among the fastest solvers currently available for large sparse systems of equations. These methods also efficiently map to a parallel computer architecture. A brief description of the Krylov solvers employed is given in the next sections.

After the scattered fields at the grid points have been determined, the fields at the receiver(s) must be calculated. The electric fields are simply calculated by bi-linear interpolation of the values at the nodal points surrounding the receiver point. The magnetic fields at the receiver are calculated by first taking the numerical approximation of the curl of the electric fields surrounding the receiver, and then interpolating to the point of interest. It must be noted that when receivers lie on or close to an interface where the material properties abruptly change, special care must be exercised in computing the fields because of discontinuities in their normal components across this boundary.

The Krylov Subspace Solver

The best known Krylov subspace solver is the conjugate gradient (CG) technique proposed by Hestenes and Steifel (1952). With this method the iterates proceed to minimize

$$\mathbf{r} = \mathbf{K}\mathbf{E_s} - \mathbf{s} \tag{11}$$

along a series of orthogonal search directions which are generated with variations of the inexpensive recursion relation developed by Lanczos (1950). Unfortunately the CG technique does not work for our problem because $\mathbf{K}$ is complex symmetric and not Hermitian. However, a number of Krylov algorithms have been developed for handling non-Hermitian systems. Though a reduction in the error is not guaranteed at each iteration and thus the convergence is generally erratic, these techniques have proven successful in reducing the error to a predetermined level within an acceptable number of iterations. The most widely used of the techniques is the bi-conjugate gradient (BICG) method first employed for electromagnetic modeling by Sarkar (1987) and more recently for magnetotelluric modeling by Smith (1992). Here we have examine both this method as well as the quasi minimum residual (QMR) technique proposed by Freund (1992), who has determined that these two methods offer the best trade off between solution accuracy and speed for matrix systems that are complex symmetric. In order for the reader to better understand the

8

parallelization of the FD scheme as discussed below, a general description of these routines is given here. For a more explicit description, the reader is referred to Freund (1992).

To illustrate the computational nature of the BICG and QMR routines, the algorithms have been decomposed into four mathematical operations which dominate the run time. These operations are listed in the left column of Table 1, with the most computationally intensive operation at the top of the column, and the least involved at the bottom. This table has been designed to show the number of times each operation is employed within the main body of the two Krylov schemes.

Operation	Number of times used within the Lanzcos recursion	
	BICG	QMR
Vector-Matrix Multiply	1	1
Vector Dot Product	3	4
Vector Addition/ Subtraction	3	6
Vector Constant Multiply	3	9

Table 1 - The type and number of major mathmatical operations incolved in the BICG and QMR schemes.

To initialize both routines, a starting solution vector E_0 is chosen, the residual given by equation 11 computed, and then various vector-matrix and vector dot products calculated. After this initialization process has been completed, the Lanczos recursion begins. As it can be seen, the QMR method is more computationally intensive than the BICG, and thus will take more time per iteration to complete the necessary calculations. However, as explained by Freund (1992) this added complexity is offset by stability which will be demonstrated below. Note that when using this method, s has to be reconstructed for each new source, and both K and s for each frequency, i.e., the process has to be reinitialized for each new source and frequency. However, if successive source positions and/or frequencies do not exhibit large differences from one another, then a bootstrapping technique can be performed where the previous solution vector is used as the initial guess (E_0) for the new model. Newman and Alumbaugh (1995) have found that for airborne simulations this process can offer a time savings of up to 15%.

Preconditioning

In order to accelerate the convergence of the BICG , QMR and other Krylov methods, preconditioning can be applied to the linear system. This process reduces the condition number of the system matrix by finding some approximation (M^{-1}) to K^{-1}, and then multiplying through the system by this matrix (Barrett et al, 1994). To this point several different preconditioning schemes have been implemented and tested including modified incomplete Cholesky methods, two different polynomial preconditioners, a block Jacobi method, and a simple diagonal or Jacobi scaling. An excellent overview of all these methods can be found in Barrett et al. (1994).

The Jacobi scaling is the easiest to implement as it only has to be applied once before the Krylov routine is called. In addition we have found that this diagonal scaling can accelerate the convergence rate of other preconditioners if it is applied prior to solution. This routine involves modifying equation (10) such that the linear system looks like

$$\left[J^{-1/2} K J^{-1/2} \right] J^{1/2} E_s = J^{-1/2} s \tag{12}$$

where $J_{ij} = 0$ for $i \neq j$, and $J_{ij} = \sqrt{K_{ii}}$ for $i=j$. The modified matrix $\left[J^{-1/2} K J^{-1/2} \right]$ is still complex symmetric but with unity on its main diagonal. Because this scaling is always applied, the scaled fields ($J^{1/2} E_s$) must be rescaled back to their true values after an acceptable convergence has been obtained.

Unlike Jacobi scaling, the other preconditioners work by modifying the residual at each iteration in order to obtain a better estimate of the optimal search direction, i.e., they must be recalculated and applied at each iteration within the BICG and QMR routines. For more detail the reader is again referred to Barrett et al. (1994).

Smith (1992) showed excellent convergence acceleration in magnetotelluric modeling using the incomplete Choelesky/DKR preconditioner, and thus our first attempts at finding a more robust preconditioner for our system employed this factorization technique. However, we have found that this method requires additional memory that severely limits the size of the problem that can be simulated. This is due to the fact that any incomplete decomposition method requires the storage of a matrix with equal size to K. Because of this extra memory requirement, we have not pursued this method any further than an initial implementation.

The remaining preconditioners that have been implemented operate without any limitations of memory. The block Jacobi method works by partitioning the variables into subsets, forming a square matrix from each subset, and then inverting each of these sub matrices. The resulting preconditioning matrix is then block-diagonal. Here we have chosen to employ a subset composed of the three unknowns assigned to a given nodal point.

Two different polynomial preconditioners have also been implemented. A Neumann series polynomial (Dubois et al.,1979) employs a truncated form of a Neumann series expansion of $\mathbf{K}$ to approximate $\mathbf{K}^{-1}$, i.e..

$$\mathbf{M}^{-1} = \frac{1}{\xi}(\mathbf{I} - \mathbf{G})^{-1} \approx \frac{1}{\xi}\sum_{i=0}^{n}\mathbf{G}^i .\tag{13}$$

where

$$\mathbf{G} \equiv I - \frac{1}{\xi}\mathbf{K} .\tag{14}$$

The idea here is that the series will converge as long as ξ is chosen to be equal to one half the spectral radius of $\mathbf{K}$ (Shadid and Tuminaro, 1994). A least squares polynomial (Saad, 1985) uses a different approach to find a polynomial, $\mathbf{M}^{-1}$, that minimizes the function

$$\|\mathbf{I} - \mathbf{M}^{-1}\mathbf{K}\|^2 .\tag{15}$$

Here, the expression

$$\mathbf{M}^{-1} = \sum_{i=0}^{nt}\tau_i\mathbf{K}^i\tag{16}$$

leads to the least squares polynomial. Because we are employing a library of preconditioners that was originally designed for real systems and has since been adapted for complex systems, the coefficients (τ_i) are approximated by assuming that all eigenvalues of $\mathbf{K}$ lie on the real axis between 0 and the infinity norm of $\mathbf{K}$ (Shadid and Tuminaro, 1994). While our problem violates this assumption, it is still possible to accelerate convergence, and the convergence properties for all of the preconditioners are discussed below in the section entitled "Numerical Characteristics of the Krylov Solvers".

PROPERTIES OF THE PML ABSORBING BOUNDARY CONDITION

The PML absorbing boundary condition developed by Berenger (1993) was chosen over other ABC's for two reasons: 1) it is naturally parallelizable as opposed to an integral boundary conditions such as the one give in Duskin and Knizerman (1994) and 2) it leaves the matrix complex symmetric which is not true of many other ABC's such as the Bayliss-Turkel and Liao conditions (Chew, 1990). In addition the incorporation of complex grid stretching involves only slight modifications to the existing serial version of the code making it very easy to implement, and

11

unlike the time-domain solution the frequency domain version does not require the amount of memory to be doubled.

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. Using their type of analysis we demonstrate that these same properties hold for the scattered electric field Helmholtz equation in Appendix A. Here we explain the physical characteristics of the complex stretching variables, how they are implemented, and briefly describe how they affect our linear system.

The complex stretching parameters are assigned a value of the form $1+a+ib$, and the method in which they are incorporated into the discrete finite difference equation is given in Appendix C. 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_j and h_j for j=x,y, and z are allowed to vary over several cells, but only in the direction that is perpendicular to the boundary. For example along the +z boundary $e_x = h_x = e_y = h_y = 1$, 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 the matrix **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 in the model that has employed a 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

$$\kappa = \sqrt{i\omega\mu_p\left(\sigma_p + i\omega\varepsilon_p\right)} = \alpha - i\beta \qquad (17)$$

where α and β 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

12

$$\delta^{ps} = \frac{1}{\alpha b + \beta a} \tag{18}$$

and a pseudo-wavelength defined as

$$\lambda^{ps} = \frac{1}{\alpha a - \beta b}. \tag{19}$$

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 depths 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.

PARALLEL IMPLEMENTATION ON MULTIPLE INSTRUCTION MULTIPLE DATA (MIMD) COMPUTERS

As mentioned in the introduction, the original serial version of the code has been modified to run on massively parallel MIMD (multiple instruction multiple data) machines which can have thousands of processors. This was a necessity in order to simulate more realistic models than has previously been available. These parallel machines are employed by assigning a given number of processors in each direction (nx in x, ny in y and nz in z) such that the total number employed is equal to nx*ny*nz. The model is then broken up across the processor bank such that each individual processor is in charge of a 3-D subset of the model. Because each processor needs only to make the necessary calculations for this subset, and because all of the processors are making the their appropriate calculations simultaneously, the solution time is reduced by a factor which is approximately equal to the number of processors employed.

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 unknowns as possible for which to

13

solve. In the ideal case where the number of nodes is equally divisible by the number of processors in all three directions, then each processor will contain the same number of unknowns. However if the number of nodes in a direction is not equally divisible by the number of processors such that there is a remainder r, then the model is divided up such that the first r processors contain $q+1$ nodes in that direction where q is the integer quotient. The remaining processors then contain only q nodes.

The second issue that needs to be addressed is the manner in which the model is input. To accomplish this, 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 permitivity and magnetic permeability) that are assigned to each cell within the model. These data are input differently then the global data for two reasons; 1) with increasing model size, the input file containing the material properties can easily exceed a hundred megabytes, and 2) because each processor is in charge of a small subset of the model, the portion of local data needed by a given processor will be different from any other processor. Because each processor contains only a small amount of local memory, a single processor can not read in this large file and then distribute the data across the machine. To get around this problem the local data is broken up into multiple files, one for each processor, which are then read in individually. Additional time savings can be incorporated by distributing these multiple files across a parallel disk system which allows several files to be read in simultaneously.

After the data have been accessed, each processor constructs its own portion of the stiffness matrix $\mathbf{K}$ and source vector $\mathbf{s}$, and then proceeds to solve for its portion of the solution vector. However as Table 1 indicates, each iteration within the solver routines 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 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 solution and residual vectors 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 contain adjacent portions of the model. This is deduced by closely examining the finite difference stencil in Figure 1b. For example, in order for the processor containing node (i,j,k) to complete its vector-matrix multiply in a given iteration, it will need to know the current values of the appropriate unknowns assigned to

14

adjacent nodes. However, if (i,j,k) lies on a boundary such that nodes (i,j+1,k-1) and (i,j+1,k) are assigned to a different processor, than the processor containing (i,j,k) will need to receive the updated values of those particular unknowns from the processor containing (i,j+1,k-1) and (i,j+1,k), and vice-versa. Thus the next and most difficult step in implementing the code on a parallel machine is to determine the stencil which defines the neighboring processors that each processor must communicate with, and which unknowns will be communicated.

This 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 1b. 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). In addition, node (i,j,k) will receive two unknowns from each node located in a positive direction ((i+1,j,k), (i,j+1,k), (i,j,k+1)) while sending each of these nodes all three unknowns assigned to it. The situation is reversed for those nodes in the negative direction as (i,j,k) receives 3 unknowns from each of these while sending only two. The second type of communication occurs across certain "corners" 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 3. Thus if the processor we are interested in contains a certain block of the model and is represented by a cube at the center of the stencil in Figure 3, then the processors, or portions of the model that it needs to communicate with are arranged about it in the manner illustrated.

The last issue to be addressed is the 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.

To this point the code has been implemented on two different MIMD machines available at Sandia National Laboratories, the 1840 processor Intel Paragon and 1024 processor NCUBE, and run time characteristics for the Paragon are given below. To provide for the required message

15

passing on these two machines 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.

DEMONSTRATION OF THE FINITE DIFFERENCE SOLUTION

To validate the numerical solution and show its versatility, we have simulated three different models which represent measurement configurations that might be employed in the field. Two of the simulations involve one dimensional geometries, i.e. layers, and are compared against a one-dimensional modeling code developed by Ki Ha Lee at Lawrence Berkeley National Laboratory. This code can incorporate layers of varying thickness, conductivity, dielectric permitivity and magnetic permeability, and can calculate the response for both electric and magnetic dipole sources oriented in any direction for normal induction frequencies up into the radar range. The third model involves a comparison against a three dimensional IE solution given in Newman et al. (1986). The run time characteristics for the different models will be mentioned here, however, a more thorough analysis of this property is given for two of the models in a subsequent section. In all of the cases presented, the background dielectric constant and magnetic permeability were both assumed to be that of free space, and only the background conductivity varies from one model to the next. In addition, the Krylov solver was assumed to have converged to an adequate error level when $\|\mathbf{r}\|^2/\|\mathbf{s}\|^2$ was found to be less than or equal to 1.0×10^{-8}. This error level is empirical and is based on extensive comparisons of the solution with other numerical solutions and scale model experiments (Alumbaugh and Newman, 1994).

Airborne Simulation

The first example crudely simulates an airborne experiment where both a VMD and x directed HMD are located 20m in the air over a 100 Ωm half space of varying magnetic permeability (Figures 4a and 5a). The sources are operating at 0.9 kHz, 7.2 kHz and 56 kHz, and eight receivers are located at the same height as these sources at five meter intervals in x. To calculate this with the 3-D finite difference code, the earth and air were divided into a 54 x 43 x 53 cell grid which yields a total of 3.7×10^5 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 $\pm$ 320m in x, $\pm$ 300m in y, and -310m and +240m in z where the air-earth interface is at 0m. The smallest cell size employed was 5m x 5m by 2.5m and was employed at the air-earth interface underneath the source-receiver array. The largest cell size employed was in the corners of the mesh and was 20m x·20m x 20m. A background conductivity of $\sigma=1.0 \times 10^{-16}$ S/m was assumed to simulate the electrical properties of the air.

16

Figure 4 shows a comparison of the magnetic fields when the permeability of the half space is set equal to μ_0. These results have had the free space response removed such that only the fields which have been back-scattered off the earth are plotted. Although the comparison for both sources is excellent, the results for the HMD are especially exciting as the correct solution for this polarization requires all vertical electric fields within the earth to go to zero; it appears that the FD solution is capable of doing so. However, the solution for the HMD takes longer than that of the VMD which indicates that this is a harder problem to solve.

It must be noted that in order to get the good comparison in the real components at 0.9 kHz, the edges of the grid had to be moved further away from the central region of the model to mitigate reflections. This was accomplished by setting $a=1.0$ and $b=-0.5$ for 10 cells along each boundary such that the additional stretching applied to each cell in this region was $2.0-0.5i$. The solution took 21% longer to converge but the results were much more accurate. In addition we found that using a combination of real and imaginary grid stretching worked better that using either alone; if only real grid stretching was employed the results were not as accurate while if only imaginary grid stretching was employed the solution would not converge. This again emphasizes the need to complete a more thorough analysis of these absorbing conditions to determine how they affect linear system at lower frequencies, and to define a more rigorous approach in choosing them.

The solution convergence times for the VMD source using 100 processors and the QMR scheme with Jacobi preconditioning was 113 seconds at 0.9 kHz, 73 sec at 7.2 kHz and 59 sec at 56 kHz. Even though this is a very small problem that uses only a limited number of processors, the quick turn around time clearly illustrates the advantage of using the parallel processors; running this same model on our high end workstation (an IBM RS600-590) took over an order of magnitude longer. In addition, these results illustrate the convergence pattern that has been generally observed for frequencies below 1 MHz with the higher frequency simulations converging quicker than the lower. A more thorough description of the convergence properties for the HMD is given in the section on the runtime properties below.

In Figure 5 the results have been plotted for a half space in which the permeability is set equal to $5*\mu_0$. Notice that once again the comparison between the 3-D and 1-D results is excellent. Also note the large differences between these curves and those for the non-permeable halfspace (Figure 4). This demonstrates that the 3-D code is correctly modeling variations in this property. Finally , due to the increased scattering caused by the high magnetic permeability of the half space, the 0.9 kHz results did not require complex grid stretching in contrast to the corresponding example given in Figure 4.

17

Crosswell Simulation

The second example is a crosswell example which simulates the EM monitoring of an enhanced oil recovery process such as the one examined in Wilt et al. (1995b). As shown in Figure 6a, a conductive block representing either a hot water or steam stimulation is placed in a 100 Ωm layer representing an oil bearing layer. The reservoir is bounded above and below by 20 Ωm layers which was assumed to be the resistivity of the background whole-space. Two wells are located 100m apart on each side of the block and in the first simulation a VMD source is located 40m above the layer (Figure 6a), while in the second case (Figure 7a) the source is within the layer.

The grid size of 82 x 62 x 63 cells yields a total of 9.6×10^5 unknowns for which to solve. (Note: this is too large of a problem to solve on our IBM workstation.) To avoid reflections off the mesh boundaries, the grid was 800 m long in x, 600 m long in y and 600 m in in depth (z). The largest cell size employed at the boundaries was 20 m x 20 m x 20 m, while within the body a minimum cell size of 2.5 m x 2.5 m x 2 m was employed. Again Jacobi scaling was used to precondition the system and in this case 252 processors were employed on the Paragon. The total run time for both sources and all three frequencies was 35 minutes, with the 0.1 kHz simulations taking approximately 5000 iterations to converge, the 1.0 kHz simulations taking 2000 iterations, and the 10 kHz runs taking approximately 700 iterations.

To check the finite difference results we calculated the reponse of the block in a layered half space using the integral equation scheme described in Newman et al. (1986). To discretize the block, cubic cells 5 m on a side were employed, which resulted in a model that took a few minutes to run for each frequency on our IBM RS6000 workstation. The comparisons are shown for the horizontal magnetic fields in Figures 6b and 7b, and for the vertical fields in Figures 6c and 7c. Notice that in all cases the comparison between the two solutions is excellent. It is especially promising that even when the source is located within a zone of anomalous conductivity (Figure 7) the comparison is very good even without fine discretization about the transmitter. This property will be addressed again in the next example.

Figures 8 and 9 show results for the same exact geologic model as given above with the VMD source replaced by a vertical electric dipole (VED) source. Thus here we are comparing the horizontal and vertical electric fields for the two solutions. Figure 8 shows that for the source located above the layer the results are in excellent agreement. Even the rapid discontinuity at the layer boundaries in the z component of the field is mapped, although it is done so in a slightly smoother fashion. This smoother transition for the FD calculations is to be expected due to the manner in which the conductivity is averaged between adjacent cells. The solution time for this model was slightly longer than that of the VMD with the 0.1 kHz simulation taking approximately 6000 iterations, the 1.0 kHz 2300 iterations, and the 10 kHz run taking 780 iterations.

The comparisons shown in Figure 9 with the VED source within the layer, though very good, were much more difficult to obtain compared to the three other crosswell examples given above. The most serious problem presented to the FD solution was that the VED source was located in a zone of anomalous conductivity. When the background wholespace was assumed to be 20 Ωm the results did not compare very well. In fact the amplitudes were off by almost an order of magnitude. To get the good comparison shown in Figure 9, a background of 100 Ωm, i.e., equal to that of the layer was assumed. This result clearly demonstrates that when the source is located within or near a region of anomolous electromagnetic properties, caution must be exercised depending on the source type and polarization as the results will not always be as accurate as they were for the VMD source in Figure 7.

It must also be mentioned here that although the results in Figure 9 compare very well, the comparison is not nearly as good as it was for the other cases, especially in the amplitudes. However, we believe that this is not the fault of the FD scheme, but rather a discretization problem with the IE solution. We have arrived at this conclusion because in order to obtain the results presented above, the descretization within the block of the IE model had to be cut in half such that the cubic cells were 2.5m on a side; if a 5m cubic cell was employed the comparison was much worse. This produced a model which took approximately 3 1/2 hours per frequency to run on our IBM workstation. Thus we could not try to further improve the comparison by more finely discretizing the IE solution as this would have resulted in a model which was too large to simulate.

High Frequency Simulation and the Need for Absorbing Boundary Conditions

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). The system will be useful in locating buried hazardous waste as well as the boundaries of trenches and landfills where the waste has been dumped. To help in the design of the system, various forward modeling codes are under development which should yield valuable information about the type of source and polarization to employ, what components of the fields should be measured, what source-receiver offsets should be employed, etc. It was the need for modeling at these high frequencies that initiated a search for absorbing boundary conditions that would both preserve matrix symmetry as well as map appropriately to a massively parallel platform.

To illustrate the accuracy of the code at these frequencies and the need for absorbing boundary conditions, the model shown in figure 10a 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" (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 second layer 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. In addition, as demonstrated below, simply enlarging the cells at the boundaries as we have done in the previous examples does not work and actually can make conditions worse due to grid dispersion. Thus absorbing boundary conditions are needed to solve the problem.

Throughout most of the following examples, 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 million 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 $\sigma=1.0\times10^{-16}$ S/m was assumed. The solution convergence times for the models with different types of boundary conditions will be examined below.

In Figure 10b the horizontal and vertical magnetic fields calculated with the 3-D code and no absorption have been plotted against Lee's 1-D solution. It is immediately evident that the 3-D solution begins to break down at about 15 MHz, and we can assume that this is due to reflections of the mesh boundaries contaminating the solution. Doubling the size of cells along the mesh boundaries , i.e. using normal grid stretching, does not help matters. Figure 10c shows that poor results occur when a real stretching parameter $a=1.0$ is employed along 25 cells of each boundary. However , when complex grid stretching is employed the results are much better. Figure 10d shows that when a stretching parameter of $b=-0.6$ is employed along 25 cells of each boundary, the fields calculated with the 3-D solution match those of 1-D solution almost exactly.

In Figure 10e we demonstrate how the absorbing boundary conditions can be employed to shrink the size of the mesh. In this case a 72 x 72 x72 mesh was employed, and again a maximum cell size of 0.15m x 0.15m x 0.15m was used. This small mesh size places the boundaries only 10.8m apart. In Figure 10e we have plotted the calculated fields that result from employing a complex grid stretching parameter of $b=-2.0$ along 10 cells of each boundary. Notice that the 3-D calculations again agree almost exactly with the 1-D solution. This example fully illustrates the utility of these absorbing boundary conditions as not only do they allow one to accurately model high frequency where propagation rather than diffusion dominates, but they also allow the mesh size to be significantly reduced which results in much quicker run times.

Because the rate of convergence of the Krylov solvers is dependent on the spectral properties of the linear system, an analysis of how the PML boundary condition affects the system can be accomplished by observing how different stretching parameters alter this convergence. Thus in Figure 11 the number of iterations it takes for solution convergence has been plotted for those results shown in Figures 10b through 10d. Notice that at low frequencies, the unstretched system converges very quickly. However as the frequency increases, the number of iterations needed for convergence also increases. The rapid increase in solution time is even more dramatic for a system that has had real grid stretching applied. However, the system with the complex grid stretching behaves much differently. Although it takes longer to converge when compared to either of the other methods at lower frequencies, it does not experience the rapid increase in solution time as the frequency increases. This indicates that unlike the other two boundary conditions, the spectral properties of the system that employs complex grid stretching is almost independent of frequency, at least for frequencies greater than 10 MHz.

In this section we have compared the 3-D finite difference code to other solutions to verify its accuracy and demonstrate its usefulness for simulating a wide range of different models. However, we have not really demonstrated the advantages of modifying the original serial version of the code to operate on a massively parallel platform except that the second and third examples were too large to run on our IBM workstation. Thus in the next section we examine some of the run time characteristics of the solution on the Intel Paragon, and then discuss some of the numerical properties of the Krylov solvers that are employed.

RUN TIME PROPERTIES OF THE SOLUTION ON THE INTEL PARAGON –

To fully demonstrate the power of the adapting the code to a parallel platform, and also demonstrate some of the questions that must be answered when using these machines, the solution time has been plotted against the number of processors employed for two of the models examined above. In the first case, we use the model shown in Figure 4a with the HMD source operating at 7.2 kHz. For this comparison we have employed the QMR solver with simple Jacobi scaling for a preconditioner, and the results are plotted in Figure 12. (Note: the time obtained for one processor is that of our IBM RS6000-590 workstation.) The resulting curve falls off exponentially with an increasing number of processors. The results clearly show that: a) as expected the Paragon converges to the solution much more quickly than the IBM and b) as the number of processors is increased past 700, the interprocessor communication begins to dominate the solution time. This is deduced from the fact that there is no decrease in solution time with increasing number of processors beyond this point. Thus we are left with a decision to make. If we wish to use the machine most efficiently, we would employ less than 200 processors such that the internal computations are dominating the solution time. We could then run several jobs

21

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.

To estimate the increase in computational efficiency of the Paragon over our IBM workstation we can perform the following exercise. We know that maximum efficiency will be obtained when the calculations rather than communication are dominating the processing time, and that this will occur when as few processors as possible are employed. In this particular case that number was 48. For these 48 processors alone the Paragon is 3.5 times faster than the IBM. However, if we were to utilize all 1840 processors the machine has to offer and run 38 jobs simultaneously, than the efficiency scales accordingly such that Paragon is 136 times faster than the IBM. This two order of magnitude plus increase in speed again demonstrates the power of these types of machines.

To verify the point at which the inter-processor communication becomes a problem we can perform the same test for a model which is too large (the 10.1 MHz VETEM calculations examined in Figure 10d above) to run on the IBM workstation. In this case we not only plotted the solution times versus the number of processors but also determined the flop rate at which the Paragon is operating. In Figure 13, the large decrease in run time with an increasing number of processors up to 512 indicates that the processors are spending the majority of their time performing calculations rather than communicating. This corresponds to solving for 10,000 to 24,000 unknowns per processor. Medium processor efficiency is obtained when between 512 and 1000 processors are employed, which corresponds to 10,000 to 3,000 unknowns being assigned to each processor. The relatively small decrease in run time with increasing number of processors over 1000 indicates that if less then 3000 unknowns are being solved for on each processor, the code will be making inefficient use of the resources due to message passing.

An additional topic that has not been addressed in these exercises is the maximum size of the model that we can simulate on the Paragon. Because we must assign a certain number of processors to each of the x,y, and z directions, we have found most complete use of all 1840 processors occurs when 14 are assigned to one direction, 13 to the second and 10 to the third. This yields a total of 1820 processors being accessed. Each node on the Paragon is equipped with 16 MBytes of RAM. If we store the matrix K in single precision to make maximum use of the memory, we have found that the maximum problem that a single processor can accommodate is 20 x 20 x 20 cells, or 24000 unknowns. Thus the maximum problem size is 280 x 260 x 200 cells which yields a total of 43.6 million unknowns for which to solve. However if for accuracy we wish to store the matrix in double precision, then the maximum number of cells each processor can accommodate is 15 x 15 x 15 cells which is equivalent to 10,125 unknowns. In this case the maximum model size is 210 x 195 x 150 which yields a total of 18.4 million unknowns being

solved for. Note, the example running on 216 processors in Figure 13 provides that each processor is running at maximum capacity. Taking the flop rate for this example (1.75 Gflops) and scaling it upward for the maximum number of processors that we can access at one time (1820) yields a theoretical maximum flop rate of 14.7 Gflops.

NUMERICAL CHARACTERISTICS OF THE KRYLOV SOLVERS

In this last section, we examine some of the properties of the Krylov solver library that we are currently using. This package has been modified for the complex symmetric system from a library that was originally developed for solving real systems (Shadid and Tuminaro,1994). Thus it does not take into account any of the special properties that our matrix system exhibits other than its symmetric nature. First we will briefly examine the properties of the two different solvers themselves, and then we will look at the effect that four different preconditioners have on the solution convergence rate.

Properties of the BICG and QMR solvers

In his paper Freund (1992) states that the QMR solver is a more appropriate solver than the BICG for solving systems that are complex symmetric. This is due to the fact that the convergence of the QMR scheme is much more stable than that of the BICG. However, because the QMR method requires more calculations per iteration, it can actually be slower. To illustrate these concepts, and to illustrate why in general we employ the QMR method, we have plotted the residuals produced by the two methods as a function of both time and iteration number for the 7.2 and 56 kHz HMD results given in Figure 4.

Figures 14a and 14b show the results for the 56 kHz example. In this case the QMR converges in fewer iterations but takes slightly longer than the BICG. Two other interesting points to notice are 1) the much more erratic behavior of the BICG residual as the solution converges and 2) the "flattening" out of the convergence rate for both methods after 500 iterations. This flattening phenomenon is most often observed at lower frequencies rather than higher, and Newman and Alumbaugh (1995) have determined that this is most likely caused by a loss in orthogonality of the Lanczos vectors due to round off errors.

These results imply that we should be employing the BICG routine because it is slightly faster. However we have chosen to regularly employ the QMR for the reason illustrated in Figures 14c and 14d. In this case (7.2 kHz) both methods converge rather quickly up to approximately 900 iterations. However the BICG method loses stability beyond this point and the residual begins to oscillate. The net result is that it takes over twice as long to converge to the desired level when compared to the QMR method. We feel that this is most likely due to the fact that the BICG

23

scheme is experiencing problems associated with round off errors before that of the QMR. Thus, due to its more stable nature we generally employ the QMR routine.

Properties of the Jacobi, Block Jacobi, and Polynomial Preconditioners

As was mentioned earlier, the Krylov preconditioning schemes that we are employing were quickly modified for complex arithmetic from a library developed for real systems. Thus a thorough analysis of the best way to implement the preconditioners, especially the polynomials, has not been undertaken. However, we are presenting results here in the hopes that it will motivate future research. For all of the following results, the model is the HMD over a 100 Ωm half-space as presented in Figure 4b. In addition the computation times that are listed resulted from using 180 processors on the Paragon.

In most of the results presented above, we have used simple Jacobi scaling as a preconditioner. In Figure 15 we present both the convergence time and the number of iterations needed for convergence of the simulation in Figure 4b when this simple preconditioner is employed. Notice that for this number of processors, the time taken to converge is about one order of magnitude less than the number of iterations. In addition it must be mentioned that the rate of convergence at 56 kHz for this model does not fit the normally observed pattern. It is our experience that for simulations below 1 MHz, higher frequency implies quicker convergence as was the case for the VMD source in Figure 4a. At this time we have no explanation for this phenomenon.

In Figure 16 the results for a block Jacobi preconditioner are plotted and are much worse both in the number of iterations and the convergence time when compared to Jacobi Scaling. Recall that this method is implemented by sub-partitioning the unknowns assigned to each node into a 3 by 3 submatrix. The fact that this sub-partitioning only takes into account a small fraction of the coupling between the various unknowns on the staggered grid may explain the poor performance of this preconditioner. However at this point we have not thoroughly diagnosed the cause of this.

The first polynomial preconditioner to be implemented is the Neumann polynomial, the results of which are shown in Figure 17 for the three different frequencies. Here the convergence times and number of iterations are plotted as a function of the number of terms employed in the polynomial. In general as the number of terms is increased, the number of iterations needed for solution convergence decreases. However, the rate at which the number of iterations decreases does not overcome the increased work at each iteration, and therefore the time needed for convergence rises. Thus the Neumann series works best if only a single term, or at maximum 2 terms in the polynomial are employed.

24

Lastly we examine the least squares polynomial which has demonstrated the most promise of any preconditioner implemented so far. The results plotted in Figure 18 show that for this method there are substantial differences on how the preconditioner works at different frequencies. At 0.9 kHz, the minimum convergence time occurs when 9 terms are employed in the polynomial, at 7.2 kHz when 3 to 6 terms are employed, and finally at 56 kHz when 2 to 3 terms are used. The maximum time savings over simple Jacobi scaling in each of these three cases is 24%, 18% and 13% respectively. From these results it appears that this preconditioner works better at low frequencies. We believe that this observation may be a function of how the coefficients of the polynomial are chosen. Remember from an earlier section that these are determined according to the distribution of eigenvalues along the real axis. However, because our system is not positive definite there will exist imaginary eigenvalues. It is possible that the real eigenvalue approximation works better at lower frequencies than at higher, and thus to get better performance, the polynomial preconditioner needs to be redesigned such that these imaginary values are taken into consideration.

CONCLUSIONS AND DISCUSSION

In this chapter we have presented a scheme to solve for the frequency domain electromagnetic response of a 3-D earth over a wide band of frequencies using massively parallel computers. The problems associated with porting the serial version of the scheme to a parallel machine have been outlined, and a variety of comparisons have been demonstrated to prove the validity of the code. Implementing the code on the 1840 processor Intel Paragon has demonstrated a decrease in computing time of over two orders of magnitude when compared to a high end IBM workstation and a similar magnitude increase in the maximum model size that can be simulated. In addition a maximum theoretical flop rate of 14.9 Gflops has been established. Finally we have demonstrated the use of different Krylov solvers and preconditioners and found the QMR scheme coupled with a least squares polynomial and simple Jacobi scaling to be the most efficient yet stable method of solution that we have available.

Currently we are using the scheme in a variety of projects, for example to assist in the design of geophysical instruments (Pellerin et al, 1995) as well as simulating airborne EM surveys (Alumbaugh and Newman, 1995; Newman and Alumbaugh, 1994). The simulations that we are running for these projects would have been impossible prior to the parallel implementation due to the size of the models and/or the number of frequencies and sources involved. We believe, however, that there is still much research to be done with regards to the implementation of this type of scheme. The most notable location for improvement is in the area of preconditioners. Techniques being considered are multigrid preconditioners, and methods to separately treat the real and imaginary components of the matrix system. A thorough study of the grid stretching

parameters at frequencies lower than 1 MHz also needs to be undertaken. This type of study will hopefully yield either an analytical or empirical method of choosing them based on the frequency, conductivity of the medium, etc., similar to the scheme employed at higher frequencies. A method 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, methods of dealing with the air-earth interface need to be more closely examined. We have found that this interface tremendously complicates the numerical problem, especially when electric dipole sources are employed on the surface.
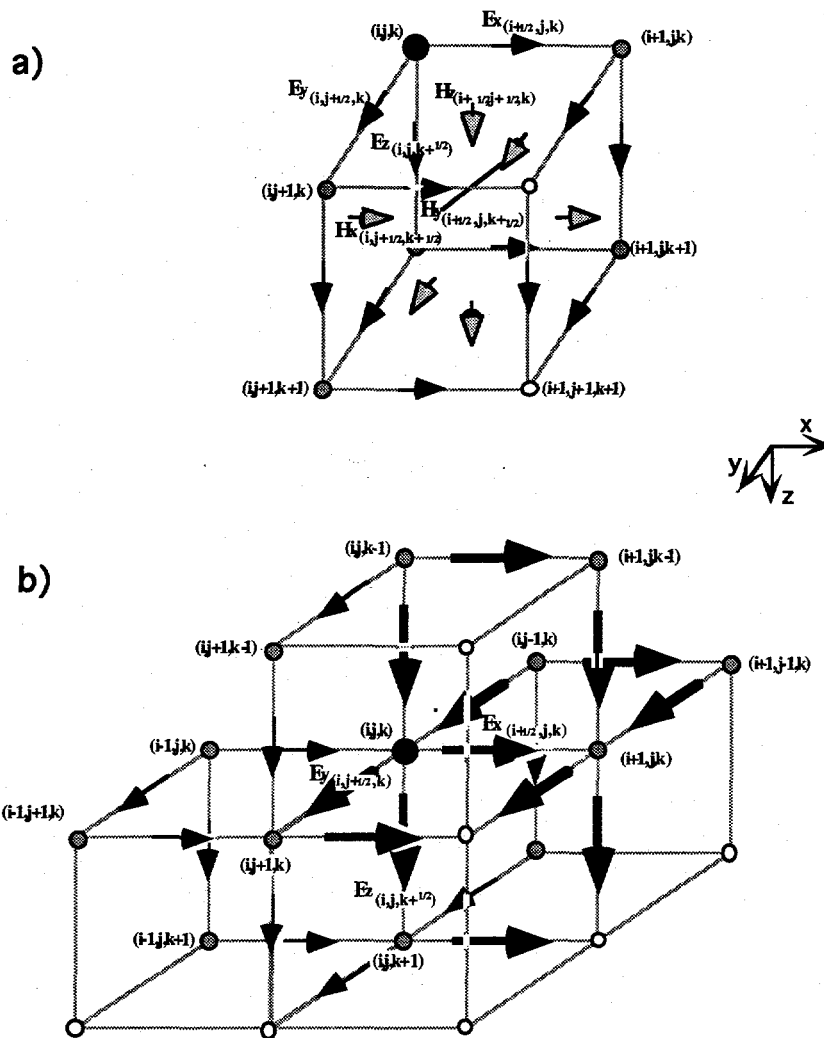
Figure 1 - a) The staggered grid for the coupled Maxwell's equations (after Yee, 1966). The electric field is sampled at the center of the cell edges, and the magnetic fields at the center of the cell faces. Node (i,j,k) is the large dark circle in the upper-back-left hand corner of cell (i,j,k), and has the six unknown electric and magnetic fields illustrated assigned to it. Cell (i,j,k) has a conductivity $\sigma_{i,j,k}$, a dielectric permitivity $\varepsilon_{i,j,k}$, and magnetic permeability $\mu_{i,j,k}$ assigned to it. b) The staggered grid for the Helmholtz equation for the electric fields. The dark circle at the center represents node (i,j,k) which has the three hi-lighted components of the electric field assigned to it. The large arrows represent the 13 unknown electric field values needed to form the equation for Ex at node (i,j,k), with the other arrows representing the additional fields needed to form the equations for Ey and Ez. The gray circles represent nodal points to which unknown electric field values are assigned that are needed to complete the three equations at node (i,j,k); the open circles represent nodal points from which no information is needed for these equations.
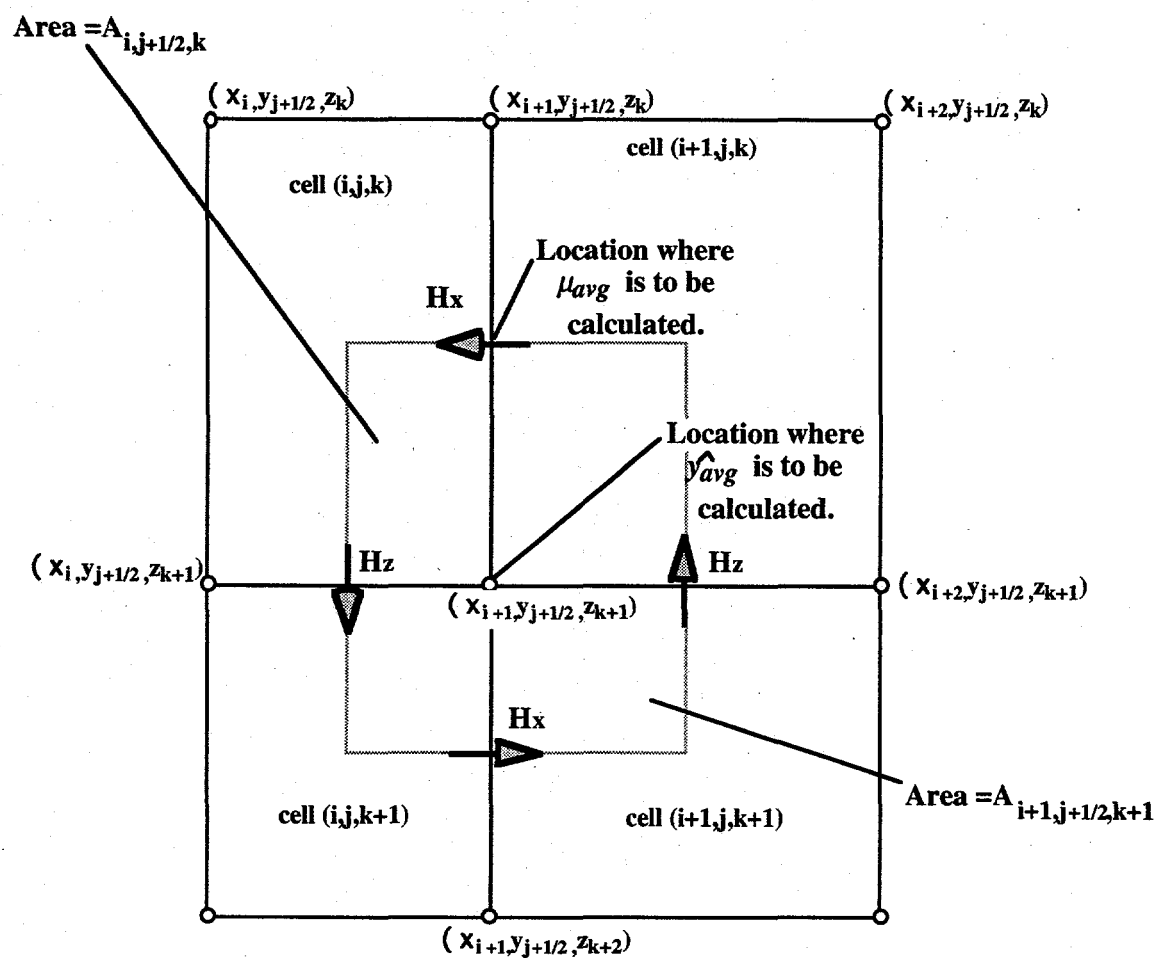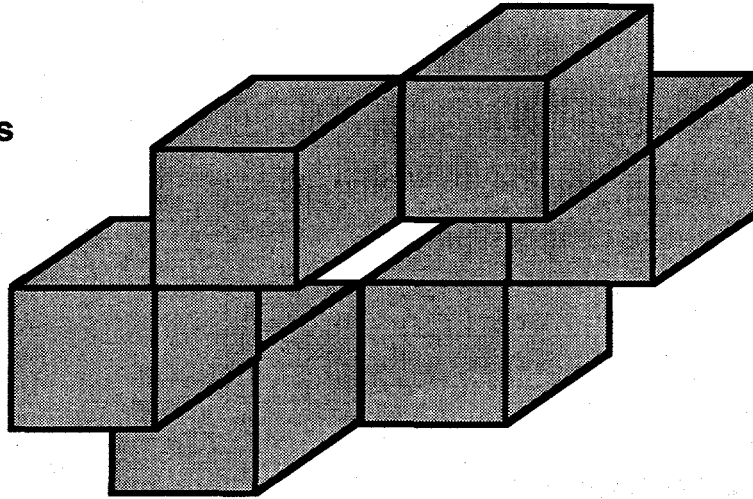
Figure 2 - Illustration to demonstrate how the average admittivity is calculated halfway along a cell edge in the y direction, and the average magnetic permeability is calculated at the center of the cell face.

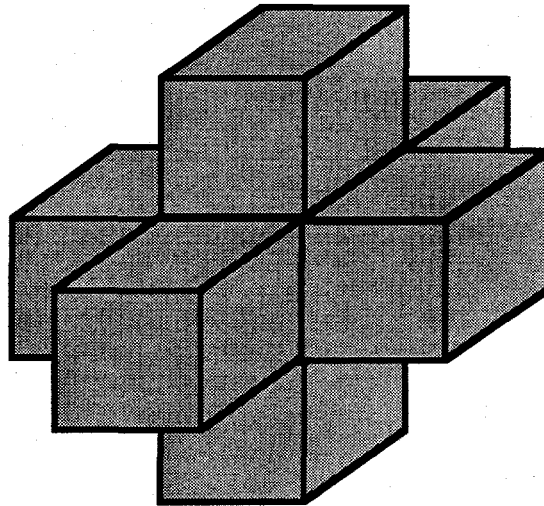**Edge Contributions**

+

Face Contributions

Figure 3 - The processor communication stencil that provides for proper message passing in the solution phase of the program. Each cube represents a neighboring processor with which a processor located at the center of the "face contributions" cluster would need to exchange information through message passing.
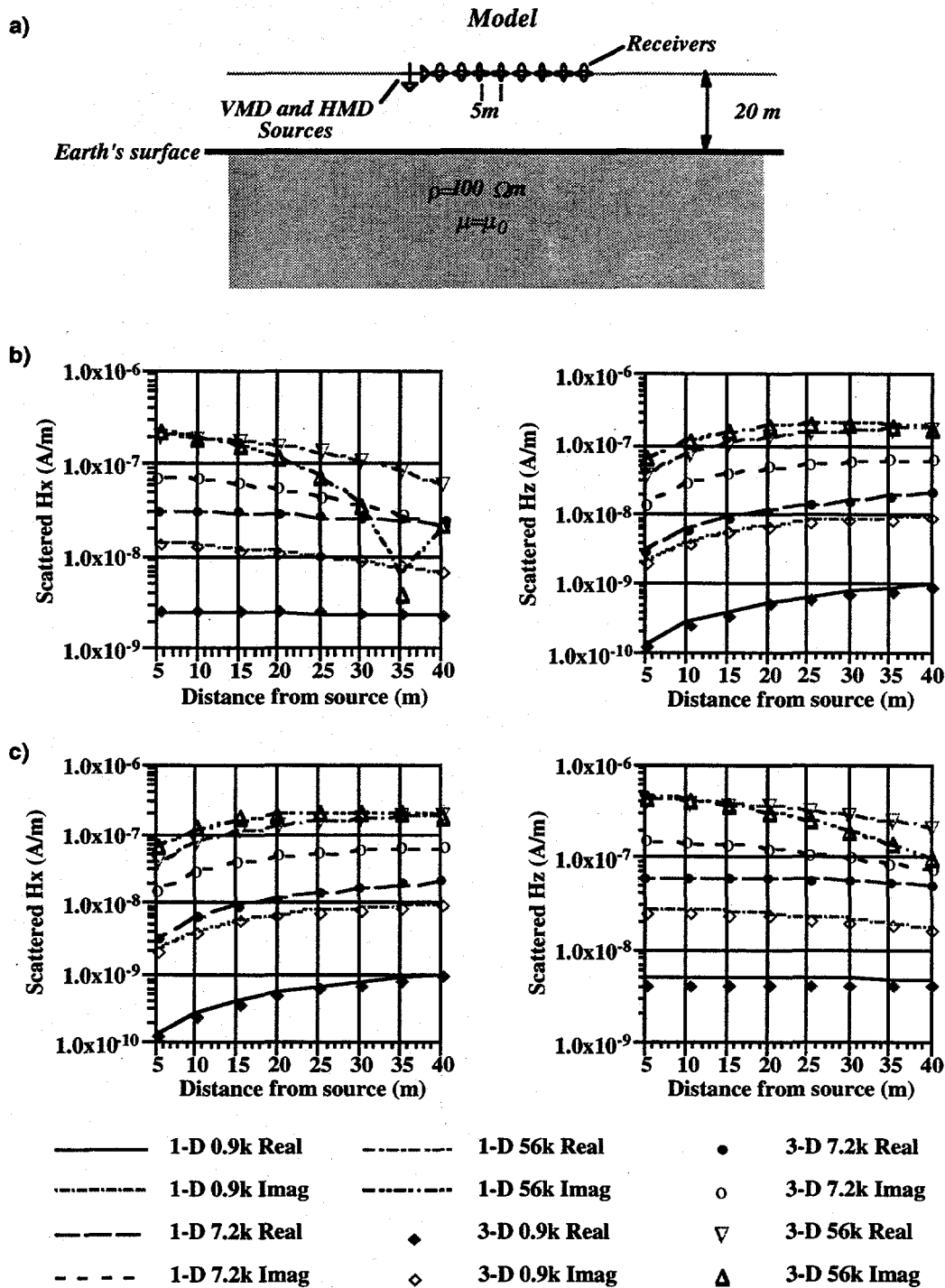
Figure 4 - a) Airborne simulation with sources and receivers above a non-magnetically permeable halfspace. The three frequencies employed are 0.9 kHz, 7.2 kHz, and 56 kHz. The comparisons are between the 3-D finite difference scheme and Lee's 1-D solution. b) Horizontal and vertical magnetic field results for a HMD source. c) Horizontal and vertical magnetic field results for a VMD source.
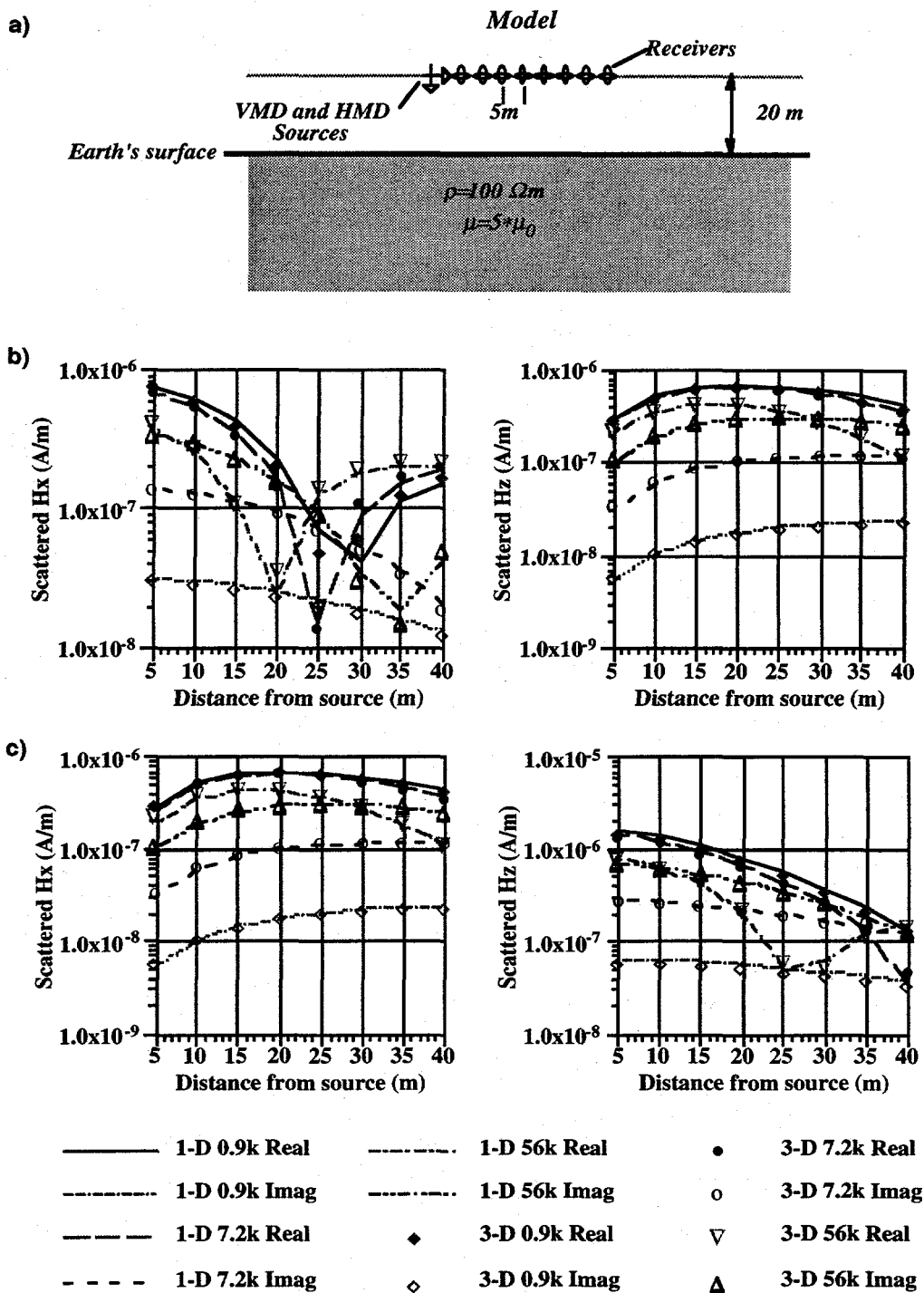
Figure 5 - a) Airborne simulation with sources and receivers above a magnetically permeable halfspace. The three frequencies employed are 0.9 kHz, 7.2 kHz, and 56 kHz. The comparisons are between the 3-D finite difference scheme and Lee's 1-D solution. b) Horizontal and vertical magnetic field results for a HMD source. c) Horizontal and vertical magnetic field results for a VMD source.
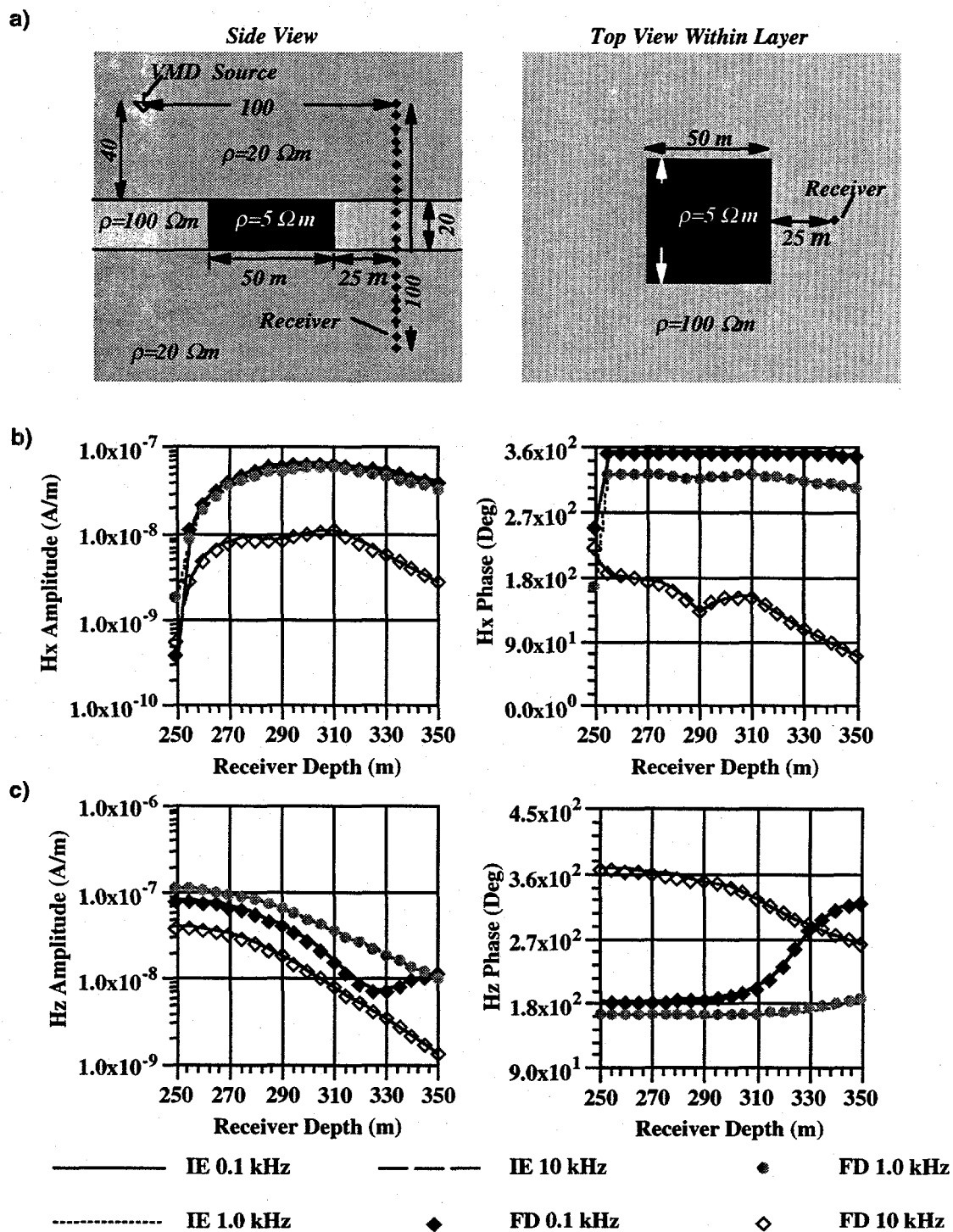
31

Figure 6 - a) Crosswell model with VMD source located 40m above the resistive layer. The receiver depths range from 250 m to 350 m, and the frequencies employed are 0.1 kHz, 1 kHz and 10 kHz. The comparisons are between the 3-D finite difference scheme and the integral equation solution of Newman et al. (1986). b) Horizontal magnetic field results. c) Vertical magnetic field results.
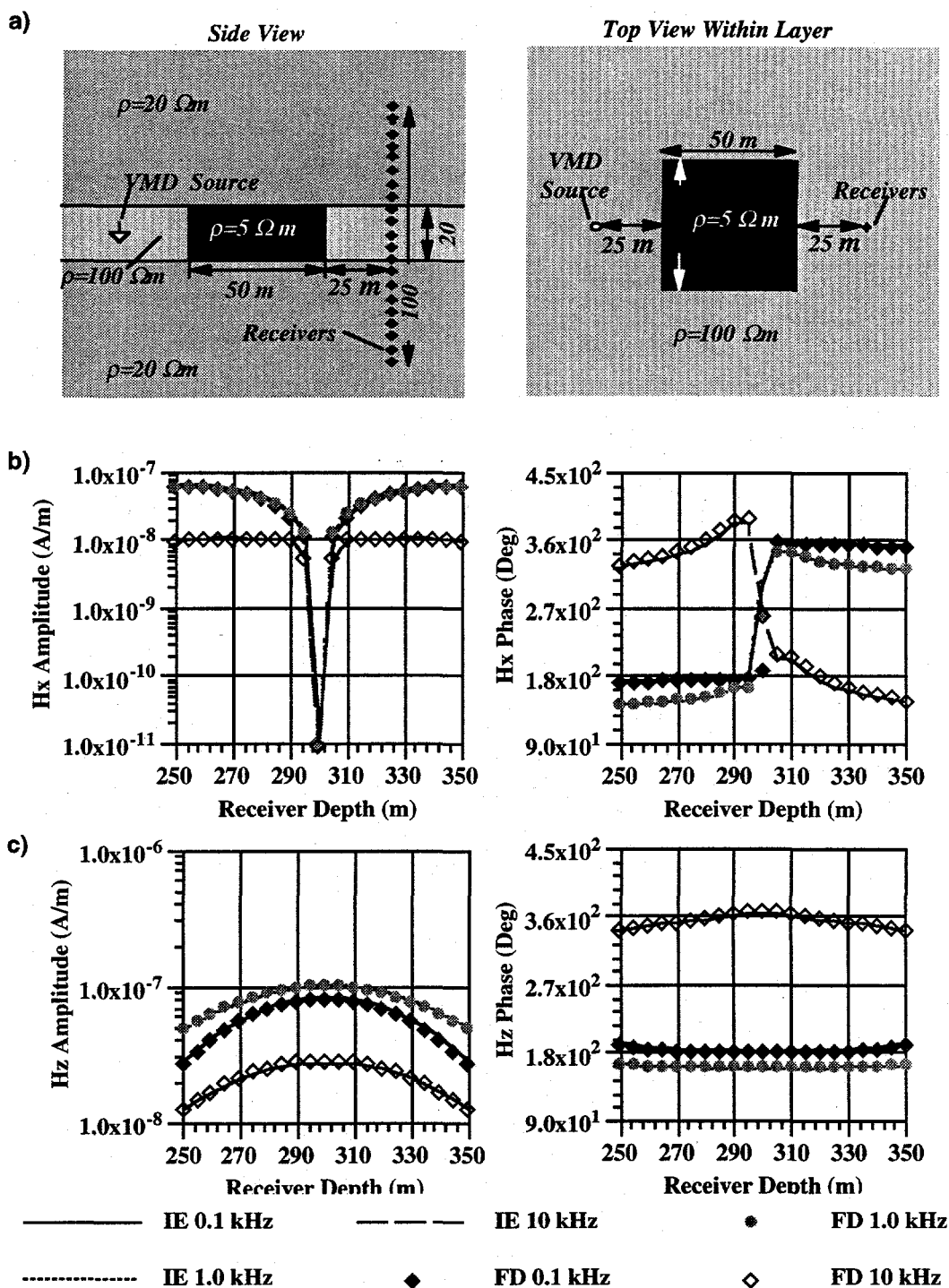
Figure 7 - a) Crosswell model with VMD source located within the resistive layer. The receiver depths range from 250 m to 350 m, and the frequencies employed are 0.1 kHz, 1 kHz and 10 kHz. The comparisons are between the 3-D finite difference scheme and the integral equation solution of Newman et al. (1986). b) Horizontal magnetic field results. c) Vertical magnetic field results.
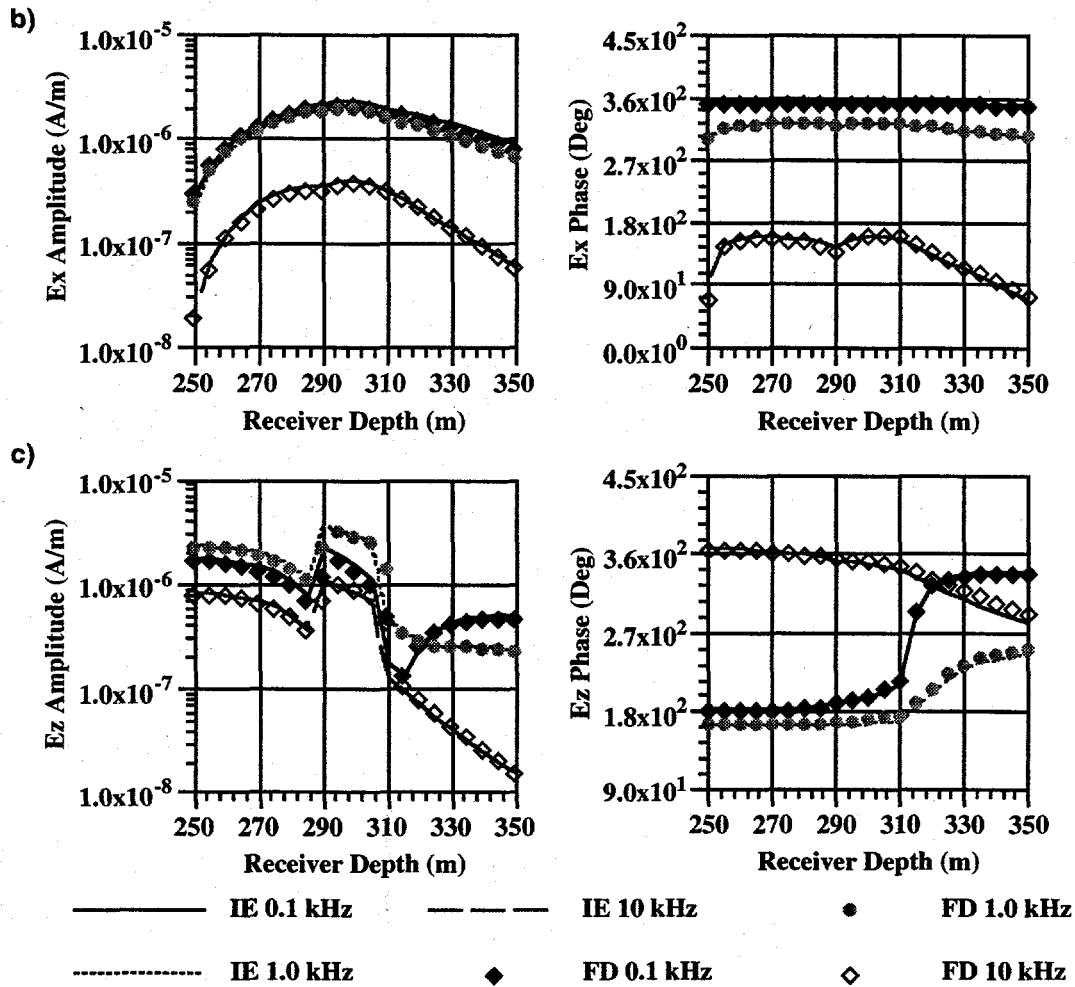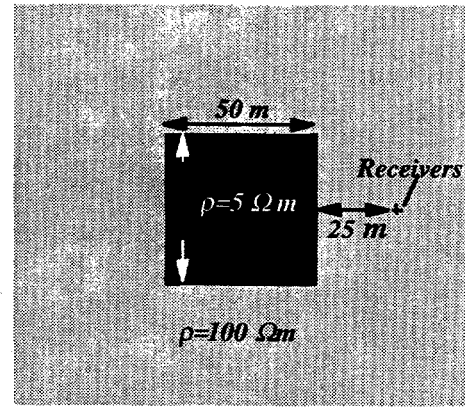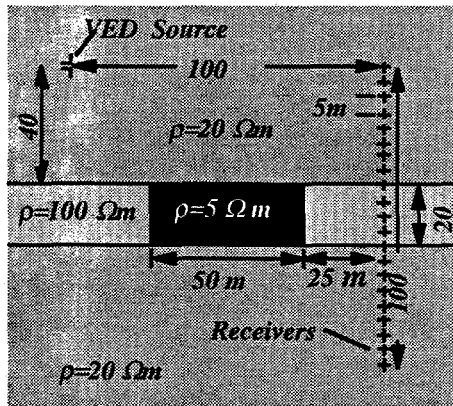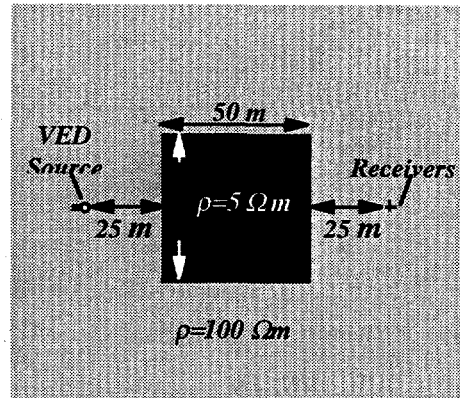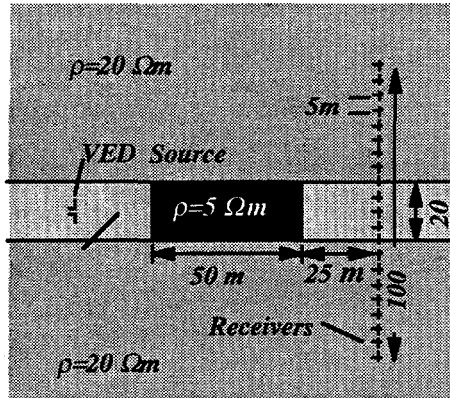
Figure 8 - a) Crosswell model with VED source located 40m above the resistive layer. The receiver depths range from 250 m to 350 m, and the frequencies employed are 0.1 kHz, 1 kHz and 10 kHz. The comparisons are between the 3-D finite difference scheme and the integral equation solution of Newman et al. (1986). b) Horizontal electric field results. c) Vertical electric field results.

34
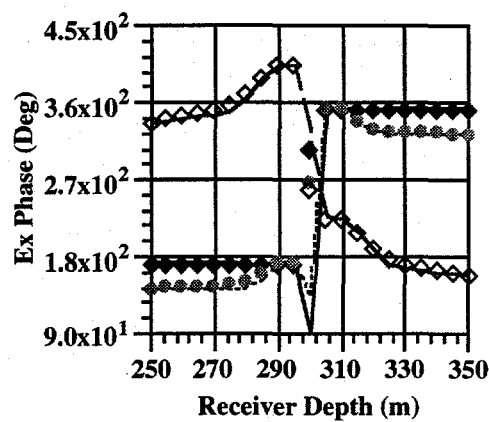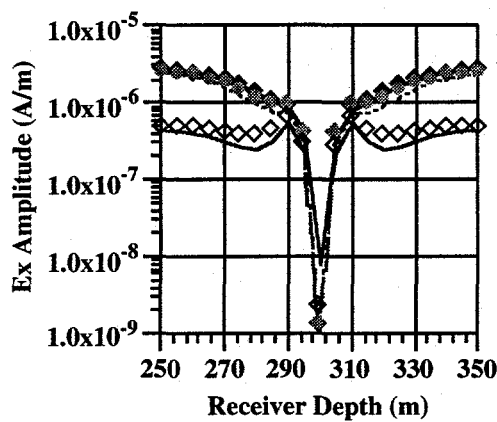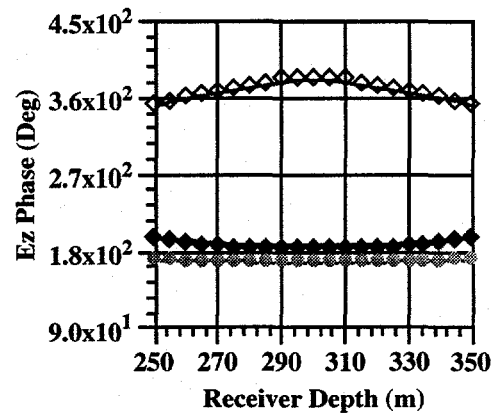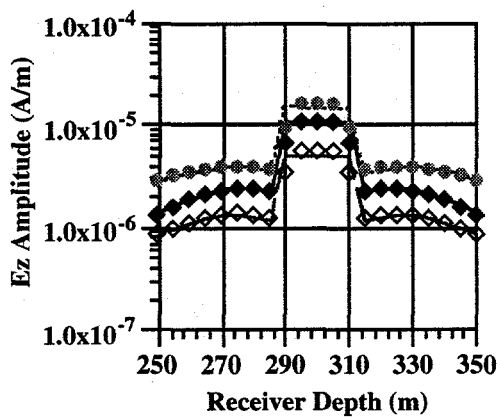
Figure 9 - a) Crosswell model with VED source located within the resistive layer. The receiver depths range from 250 m to 350 m, and the frequencies employed are 0.1 kHz, 1 kHz and 10 kHz. The comparisons are between the 3-D finite difference scheme and the integral equation solution of Newman et al. (1986). b) Horizontal electric field results. c) Vertical electric field results.

35
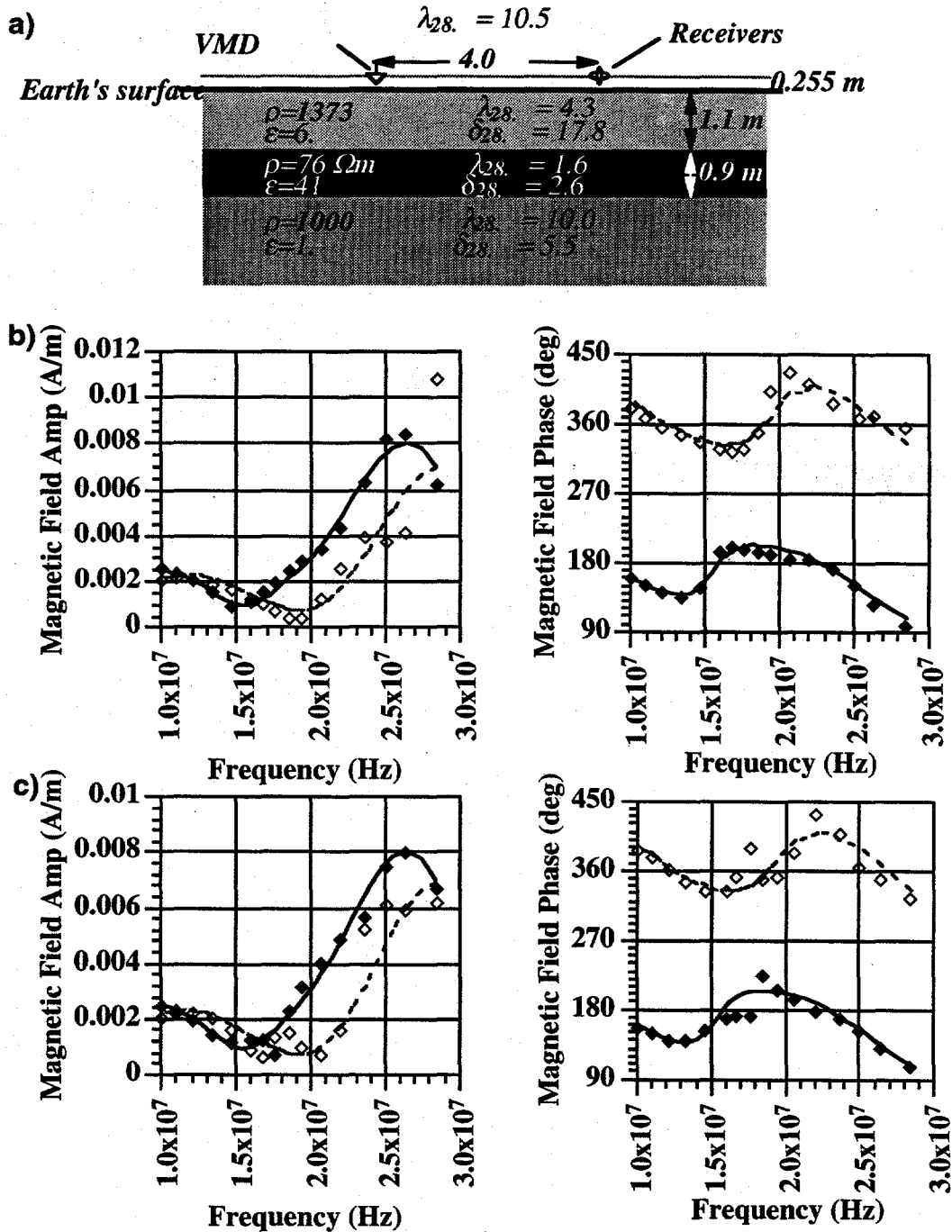
Figure 10 - a) Colorado School of Mines 1-D model. The magnetic permeability in all three layers has been set to that of free space. Unless otherwise noted a 120 x 120 x 120 cell mesh was employed with a maximum cell dimension of 0.15m. The comparisons are between the 3-D finite difference scheme (symbols) and Lee's 1-D solution (lines). b) Horizontal and vertical magnetic field results obtained with no grid stretching. c) Horizontal and vertical magnetic field results obtained with real grid stretching (a = 1.0,b = 0.0) for 25 cells along each boundary.

d) Horizontal and vertical magnetic field results obtained with complex grid stretching (a = 0.0,b = -0.6)

e) Horizontal and vertical magnetic field results obtained on a 72 x 72 x 72 cell mesh with increased complex grid stretching (a = 0.0,b = -2.0) for 10 cells along each boundary.

Legend:

—— 1D Hz Results ◆ 3D Hz results

- - - 1D Hx Results ◇ 3D Hx results
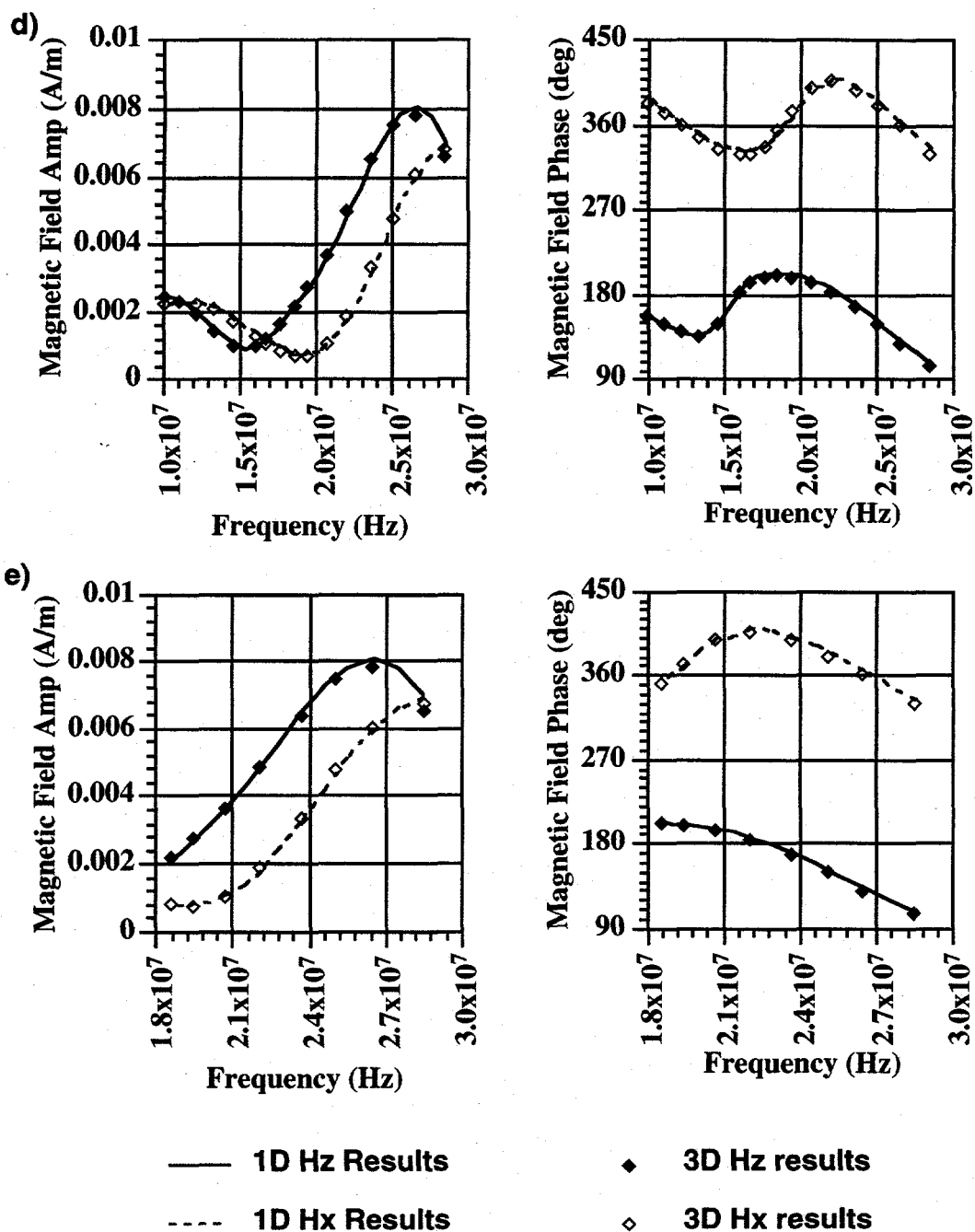
Figure 10 (cont.) d) Horizontal and vertical magnetic field results obtained with complex grid stretching (a = 0.0,b = -0.6) for 25 cells along each boundary. e) Horizontal and vertical magnetic field results obtained on a 72 x 72 x 72 cell mesh with increased complex grid stretching (a = 0.0,b = -2.0) for 10 cells along each boundary.
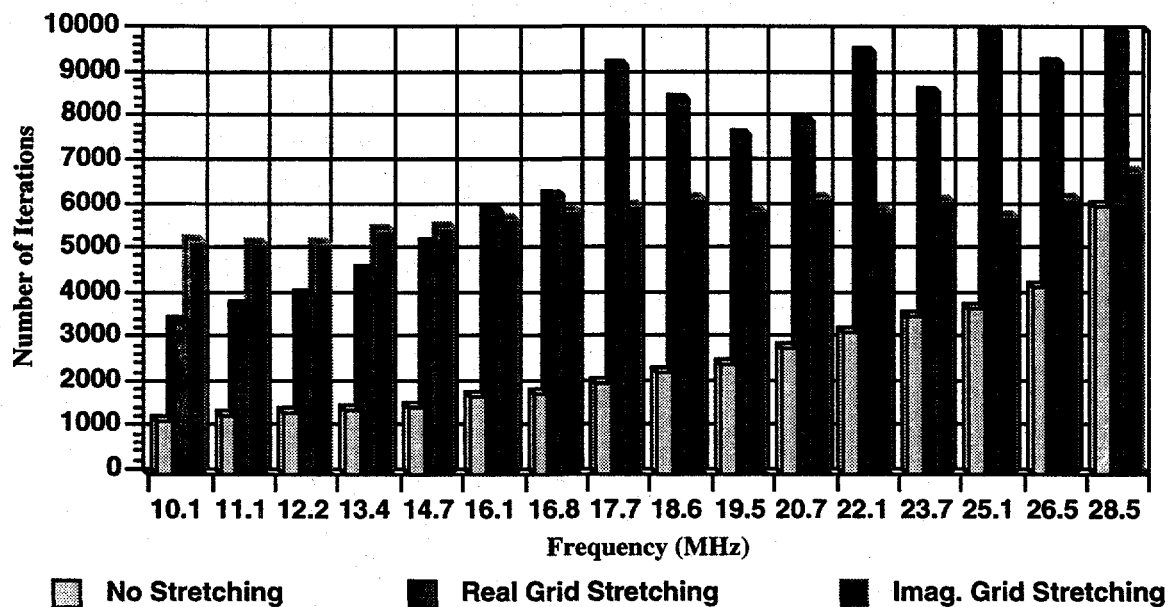
Figure 11 - The solution times of the QMR method for different frequencies and the different grid stretching parameters employed for Figures 10b through 10d.
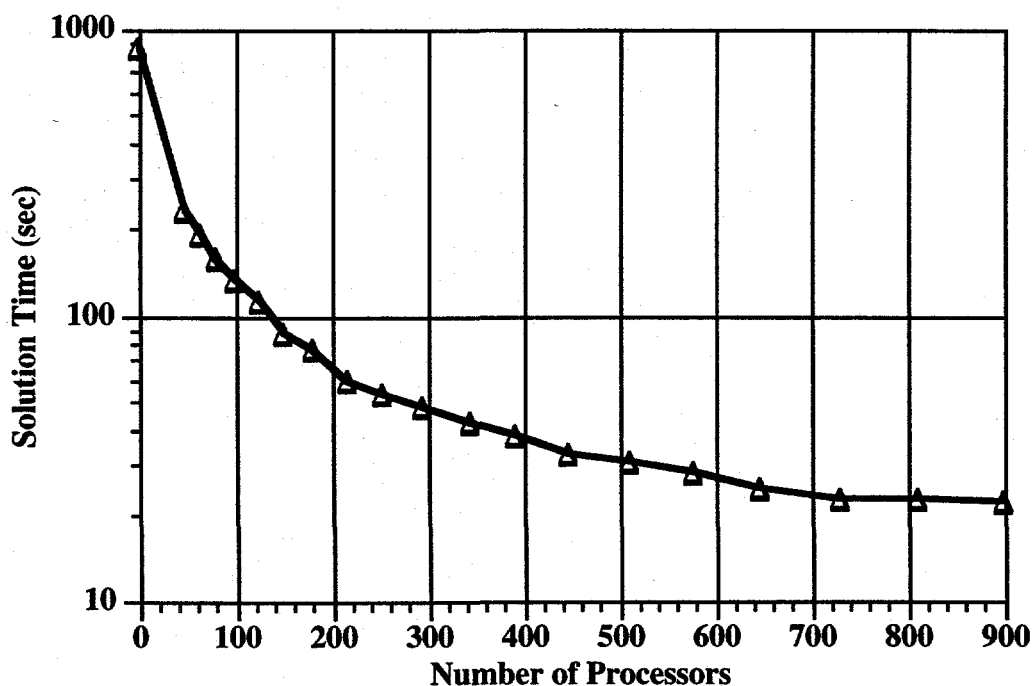
Figure 12 - The solution time versus the number of processors employed on the Intel Paragon for the 7.2 kHz HMD simulation in Figure 4. Note: the result for one processor is for an IBM RS6000-590 workstation.
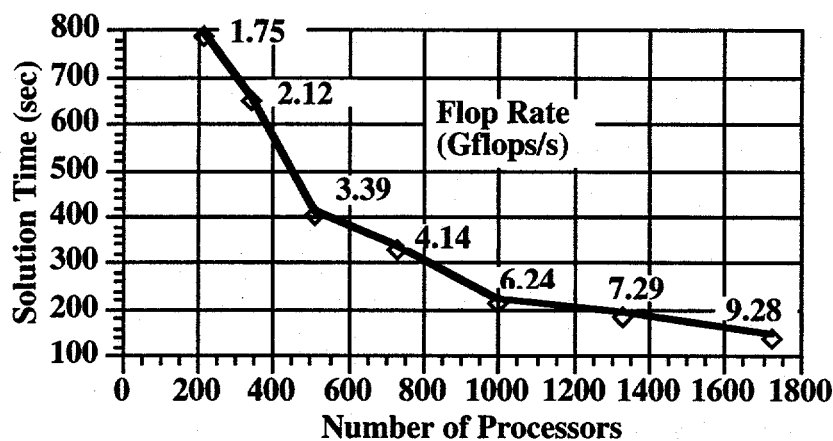
Figure 13 - The solution time and flop rate versus the number of processors employed on the Intel Paragon for the 10.1 MHz simulation plotted in Figure 10d.
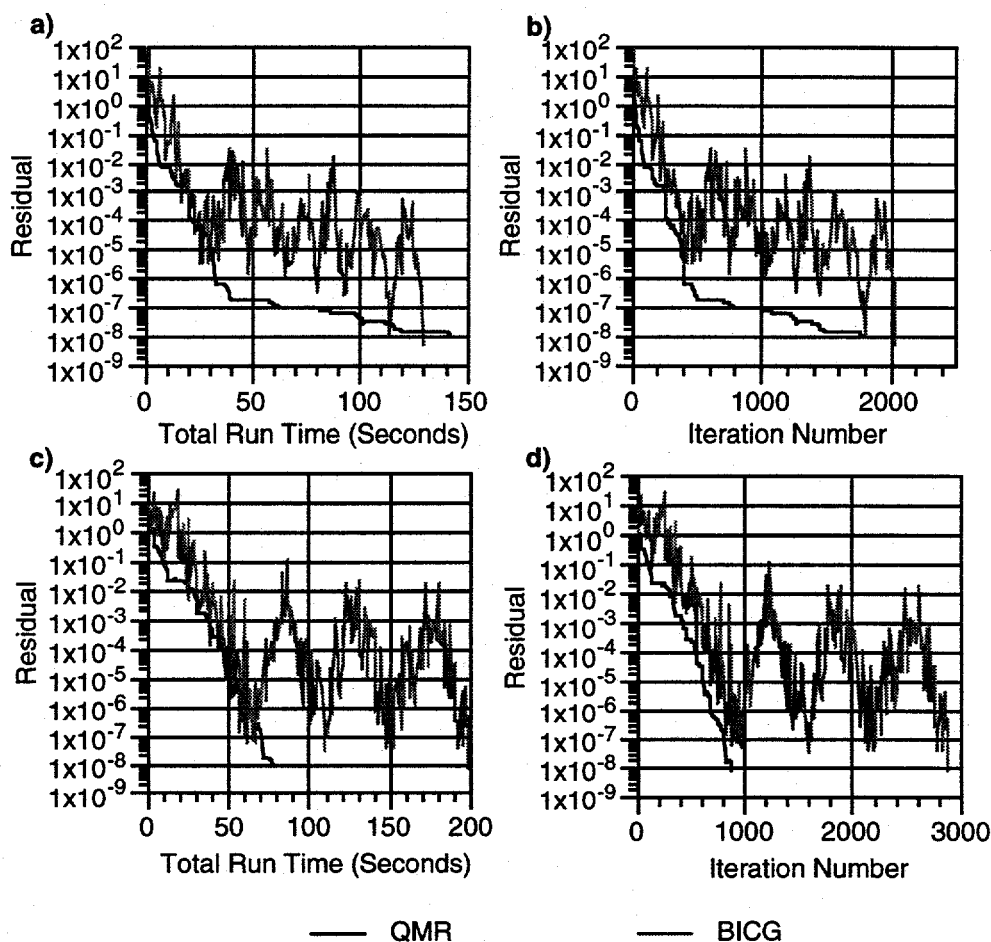
Figure 14 - Convergence of the BICG and QMR methods for the HMD simulation in Figure 4. a) Normalized residual versus run time for 56 kHz simulation. b) Normalized residual versus iteration number for 56 kHz simulation. c) Normalized residual versus run time for 7.2 kHz simulation. b) Normalized residual versus iteration number for 7.2 kHz simulation.

39
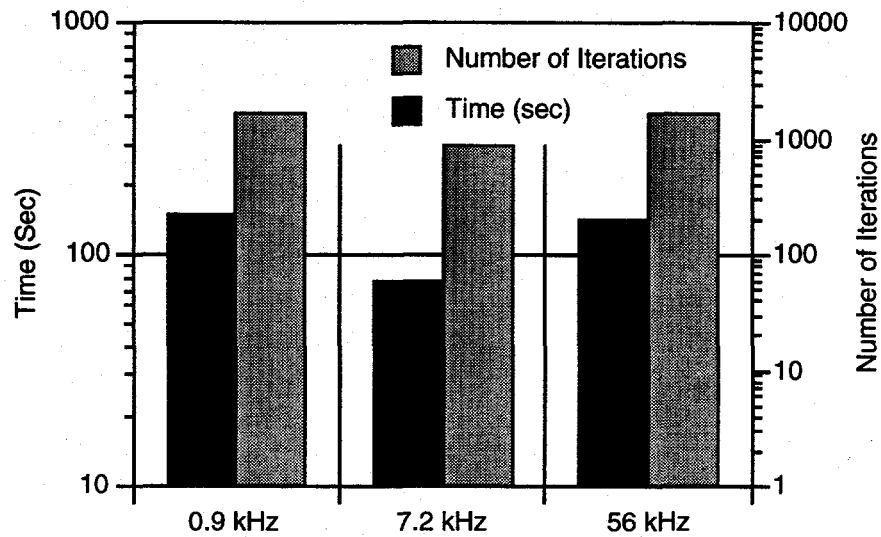
Figure 15 - Time of convergence, and number of iterations needed for convergence of the HMD simulation in Figure 4 for all three frequencies using the QMR solver with Jacobi scaling.
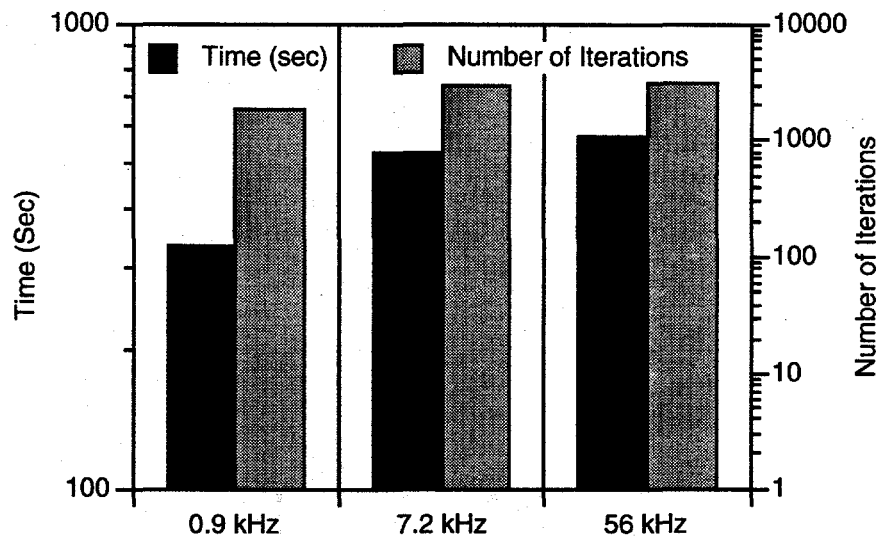
Figure 16 - Time of convergence, and number of iterations needed for convergence of the HMD simulation in Figure 4 for all three frequencies using the QMR solver with Block Jacobi preconditioning.
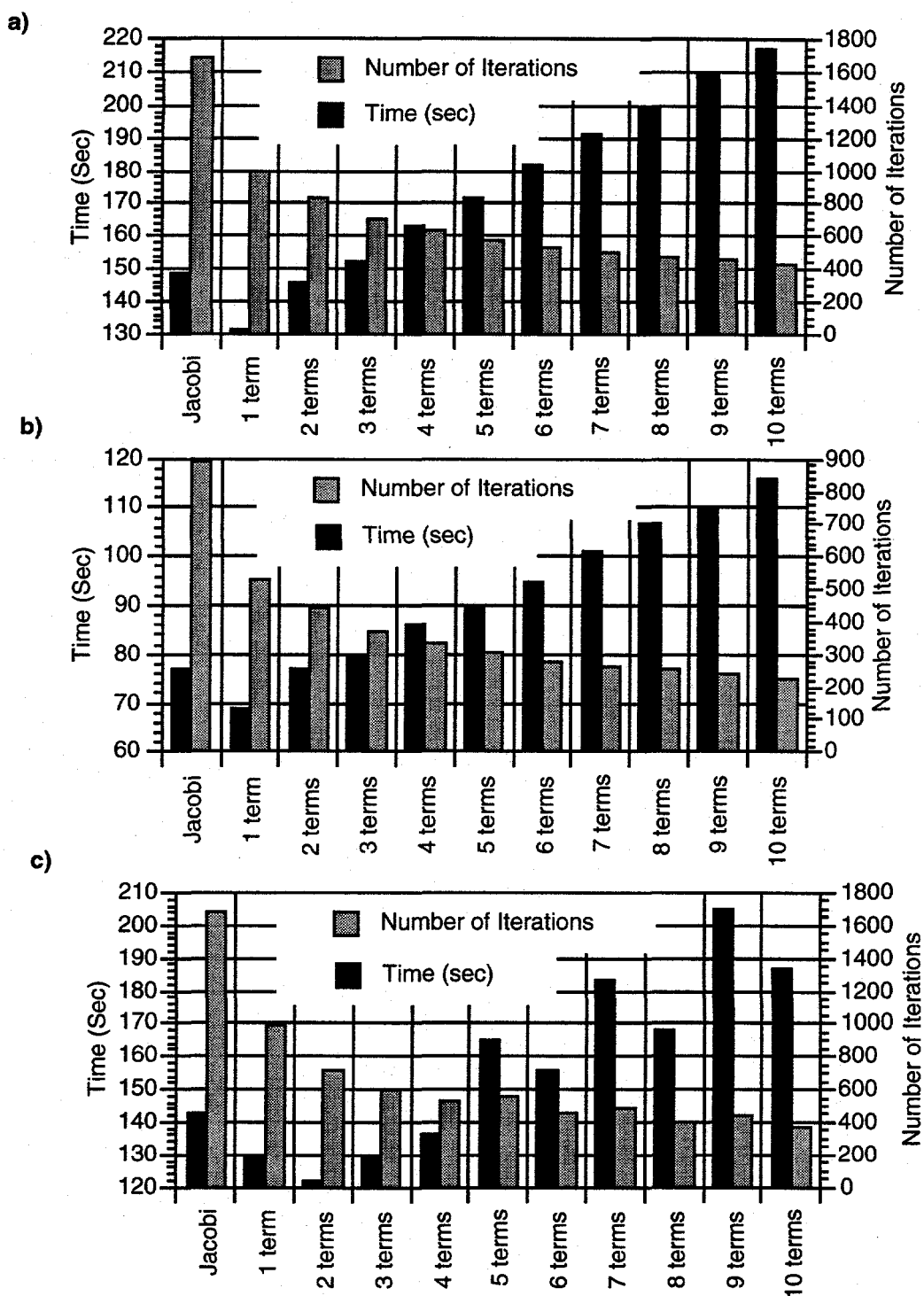
Figure 17 - Time of convergence, and number of iterations needed for convergence of the HMD simulation in Figure 4 using the QMR solver with Neumann polynomial preconditioning. The results are plotted as a function of the number of terms employed in the polynomial with the first entry representing simple Jacobi scaling. a) 0.9 kHz. b) 7.2 kHz. c) 56 kHz.
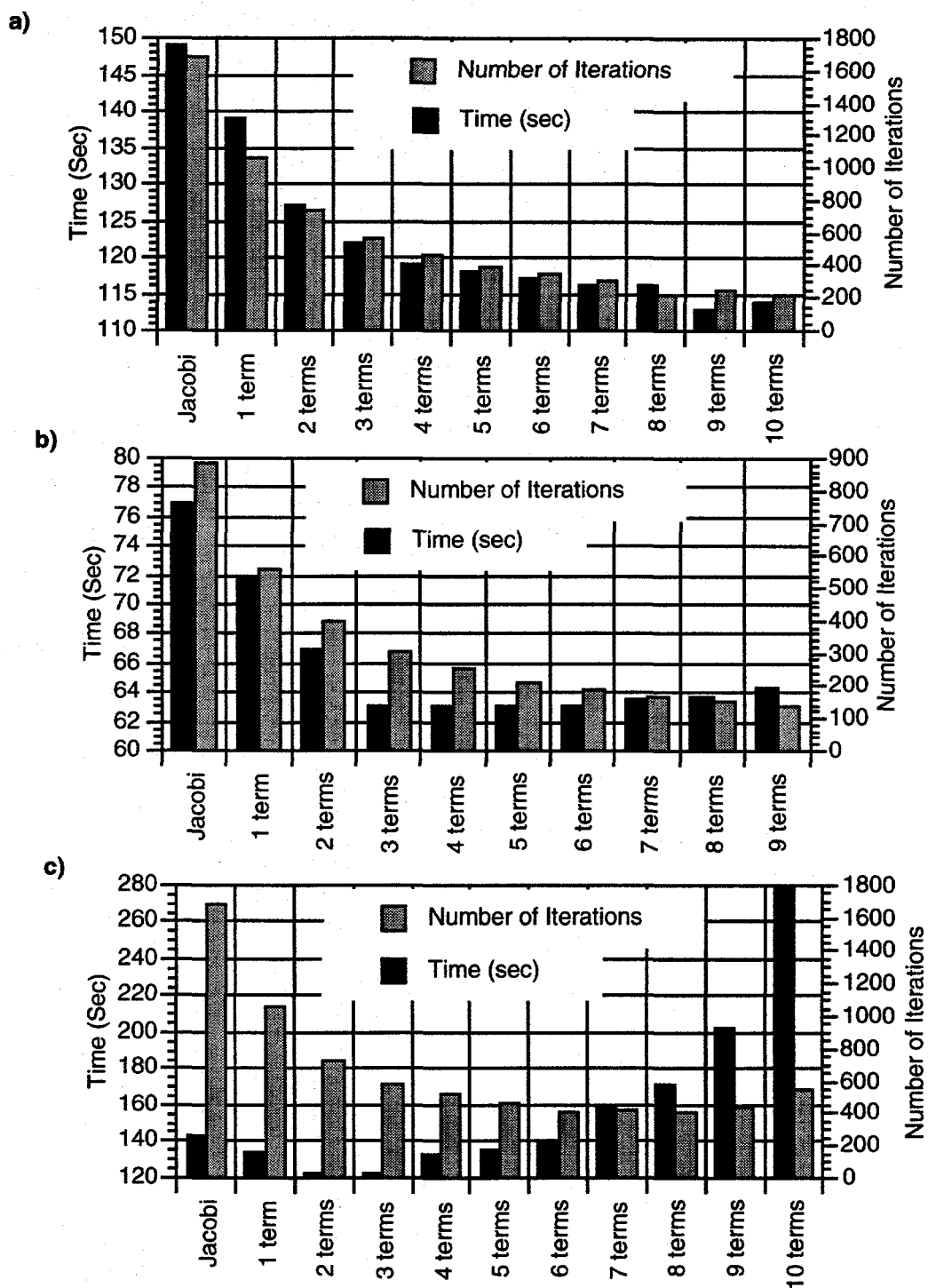
Figure 18 - Time of convergence, and number of iterations needed for convergence of the HMD simulation in Figure 4 using the QMR solver with least squares polynomial preconditioning. The results are plotted as a function of the number of terms employed in the polynomial with the first entry representing simple Jacobi scaling. a) 0.9 kHz. b) 7.2 kHz. c) 56 kHz.

CHAPTER III

3-D MASSIVELY PARALLEL ELECTROMAGNETIC INVERSION -- THEORY

To this point, a complete solution to the 3-D inverse problem has been hindered by insufficient computing resources. Realistic 3-D reconstructions require tens of thousands of unknown electrical parameters to be estimated. This demand coupled with forward modeling overhead, where up to several million field unknowns may need to be calculated to determine model sensitivities and predicted data, make the solution of the 3-D inverse problem non trivial. Attempts to circumvent this difficulty have included the use of quasi-linear approximations in both the forward as well as the inverse modeling (cf. Torres-Verdin and Habashy, 1995 and 1994, Habashy et al., 1995, Zhandnov and Fang, 1995a) and the use of approximate model sensitivities (Farquharson and Oldenburg, 1995). Unfortunately even these approaches suffer when the number of parameters being estimated exceeds several thousand. Only with the advent of massively parallel (MP) computers can a realistic attack to the problem be proposed.

Even with an MP platform one must be careful when implementing a solution to the inverse problem. Foremost is to avoid directly inverting large matrix systems that are either sparse or full. Rigorous modeling of 3-D EM fields can be carried out efficiently using staggered finite differences, which produces a sparse linear system. On an MP platform this system, if properly preconditioned, can be quickly solved using iterative Krylov subspace methods (refer to Chapter II for examples). On the other hand, the solution of the least-squares inverse problem requires dealing with a full linear system. However, since this system satisfies the normal equations it can also be efficiently solved iteratively with conjugate gradient (CG) methods. Mackie and Madden (1993) and Zhang et al. (1995) used this approach to attack the 3-D magnetotelluric (MT) and direct current (DC) inverse problems, respectively, on scalar platforms. Here we will apply the approach to the 3-D EM inverse problem for frequency-domain dipolar source fields, where the source strengths and locations are known. Because the controlled source EM problem is far more computationally demanding than both the DC problem due to its vector nature, and the MT problem due to the sheer number of source fields to be considered (upwards of several hundred), an MP platform is a necessity. As will be demonstrated below such a platform allows large models to be reconstructed, which are not underparameterized, in a reasonable amount of time.

A key consideration in developing any inverse solution is efficient computation of model sensitivities. Because we will solve the inverse problem from an underdetermined point of view, we can efficiently carry out calculations involving model sensitivities using reciprocity, which is known as the adjoint solution to the problem. The use of reciprocity, where the receivers act as sources, can be used to limit forward modeling to the number of the transmitter and receiver positions at a given frequency. The traditional approach requires the number of forward solves to be equal to the number of parameters used in the inverse. When the number of parameters far exceeds the number of transmitters and receivers, the adjoint approach is obviously most efficient (cf. McGillivray and Oldenburg, 1990). In fact using the adjoint approach coupled with the CG solution of the normal equations one can even avoid forming individual components of the model sensitivity matrix, hence resulting in a significant savings of computational memory.

In this chapter we present the theory behind the 3-D inversion scheme, including how the scheme must be modified to run on a parallel computer. Next synthetic data generated by an integral equation code will be inverted. This provides an independent check on the solution as the data are produced by a code that is very different in nature from the finite difference code used in the inversion routine and are thus prone to different numerical errors.

THE INVERSE SOLUTION

Regularized Least Squares

As already mentioned 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 (m) may be possible if the 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 error between the measured and predicted data.

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

$$S = [(D((d-d^{P(i)})-A^{P(i)}(m-m^{(i)})))^t (D((d-d^{P(i)})-A^{P(i)}(m-m^{(i)}))) - \chi^2] + \lambda(Wm)^t(Wm). \quad (20)$$

The terms in equation (20) that control how well the data are fit by the model are as follows: 1) the observed data, represented by the vector d, 2) the predicted data arising from the reference model $m^{(i)}$ denoted by $d^{P(i)}$, 3) a data weighting matrix D, which is diagonal and consists 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, 4) the Jacobian or model sensitivities matrix given by $A^{P(i)}$ and 5) χ^2 the estimated square-error in the observed data. In addition t represents the transpose operator instead of the Hermitian operator because the data, predicted data, data weighting matrix and the Jacobian matrix have been split into real and imaginary parts, where we assume the model parameters, m, to be always real valued. The parameters that control model smoothness are 1) the regularization matrix W, which consists of a finite difference approximation to the Laplacian (∇^2) operator and is sparse and 2) the tradeoff parameter λ, which is used to control the amount of model smoothness in the reconstruction. Its selection requires special care if the inverse solution is to provide acceptable results. Selecting tradeoff parameters that are too small can produce models that are physically unreasonable; although the models produce superior data fits they are unreasonably rough. Selecting tradeoff parameters that are too large produce highly smoothed models, however these models show poor dependence on the data. We shall defer further discussion of this parameter until we discuss the iterative nature of equation (20).

Minimization of equation (20) with respect to m yields the model update,

$$m = [(DA^{P(i)})^t (DA^{P(i)}) + \lambda(W)^t(W)]^{-1} (DA^{P(i)})^t (D\delta d^{(i)}) \quad (21)$$

with

$$\delta d^{(i)} = (d - d^{P(i)} + A^{P(i)}m^{(i)}). \quad (22)$$

Because negative values of m are an admissible solution arising from equation (21), it is advisable that before minimizing equation (20) it should be reformulated so one can invert for the natural logarithm of the parameters instead of the parameters themselves (Appendix D). This

45

causes the imaged properties to be always positive which is a physical requirement. By using a log parameterization, it is also possible to incorporate a lower bound positivity constraint in the inverse solution.

Derivation of the Jacobian Matrix Elements

Deriving a computationally efficient form of the Jacobian matrix elements is critical for a robust inverse solution, since calculation and manipulation of these elements is the bottleneck within the inversion. To derive these elements consider a single predicted data point, d_j, defined for a given transmitter-receiver pair as

$$d_j = d^b_j + g_j^t E_s. \tag{23}$$

In this equation d^b_j is a field arising from some specified whole space or layered half space background model at location j and E_s is the scattered electric field vector arising due to 3-D changes within this background. E_s has dimension of NTx1 and is determined from the finite difference forward solver, (discussed in Chapter II), where NT represents the number of field unknowns. The vector g_j^t is an interpolator vector for the jth measurement point and is of dimension 1xNT. This vector will interpolate fields on a staggered grid to the measurement point and can also be used to numerically approximate magnetic field measurements through the curl of the electric field. With this definition an element of the Jacobian matrix is written as

$$\partial d_j/\partial m_k = g_j^t \, \partial E_s/\partial m_k. \tag{24}$$

From the forward problem the scattered electric fields are determined from the linear system,

$$KE_s = s, \tag{25}$$

where K is the sparse finite-difference stiffness matrix with 13 non-zero entries per row and depends linearly on the electrical parameters we desire to estimate. Because the forward problem is formulated for the scattered fields, the source vector, s, for a given transmitter also depends linearly on the model parameters. It is related to the difference between the model parameters

46

and the background model, weighted by the background electric field, E^b; refer to Chapter II for the details. Thus differentiating equation (25) with respect to m_k yields,

$$\partial E_s/\partial m_k = K^{-1}(\partial s/\partial m_k - \partial K/\partial m_k E_s),\qquad(26)$$

and an element of the Jacobian matrix in complex form can be written as

$$\partial d_j/\partial m_k = g_j^t\, K^{-1}(\partial s/\partial m_k - \partial K/\partial m_k E_s).\qquad(27)$$

Model Step via Conjugate Gradients

As the number of unknowns increases beyond several thousand, using direct matrix inversion to compute the updated model, **m**, in equation (21) is not feasible, even with an MP platform. Instead we opt for an iterative solution. Since equation (20) satisfies the normal equations, the conjugate gradient method of Hestenes and Stiefel (1952) can be used to get the solution. This method offers a benefit over direct inversion in two ways: 1) following Mackie and Madden (1993) and Zhang et al., (1995) it is possible to avoid explicitly forming the Jacobian Matrix, $A^{P(i)}$ and its transpose altogether, thus saving considerable computer storage, and 2) as the number of unknowns, n, increases the solution for direct inverse goes as n^3 compared to n^2 with the iterative approach. Finally, it is much easier to implement a CG routine on a parallel platform when compared to a full matrix inversion.

In the conjugate gradient method all one needs is one matrix-vector multiply per relaxation step. However, because the matrix given by this operation is $[(DA^{P(i)})^t\,(DA^{P(i)}) + \lambda(W)^t(W)]$, there are several other matrix-vector multiplies to be considered. First, the matrix product of $(DA^{P(i)})^t$ with $DA^{P(i)}$, requires two matrix-vector multiplies. In addition the regularization-matrix product with its transpose requires two more matrix-vector multiplies. Since the latter matrix-vector multiplies are easy to implement and compute, no further elaboration will be given to them until the MP implementation of the 3-D inverse.

For the Jacobian matrix-vector multiplies, $DA^{P(i)}$ and $(DA^{P(i)})^t$, we have

$$y = DA^{P(i)}u\qquad(28)$$

and

$$z = (DA^{p(i)})^t y, \qquad (29)$$

where u is an arbitrary real vector, known as a CG search direction vector. The vectors y and z are also assumed to be real in the above expressions. We now determine compact and computationally efficient forms for the two matrix vector multiplies. These forms will also be used to treat the matrix-vector multiplies given in equations (21) and (22), i.e. $A^{p(i)}m^{(i)}$ and $(DA^{p(i)})^t (D\delta d^{(i)})$, which are needed to initialize the CG solver at each iteration of the inversion. For compact expressions, let the vector y in equation (29), the observed and predicted data, as well as the data weighting matrix be redefined as complex. Using the results from Appendix E and equation (28), we have for the jth element of the first matrix vector multiply

$$y_j = \text{Cmplx}(\text{Re}(g_j^t\ K^{-1} \sum_{k=1}^{M} u_k(\partial s/\partial m_k - \partial K/\partial m_k E_s))\text{Re}(D_{jj}),$$

$$\text{Im}(g_j^t\ K^{-1} \sum_{k=1}^{M} u_k(\partial s/\partial m_k - \partial K/\partial m_k E_s))\text{Im}(D_{jj})), \qquad (30)$$

where M is the total number of parameters to be estimated and D_{jj} is the jth diagonal entry of the matrix D. E_s here denotes the scattered electric field arising from a given transmitter at a specific frequency used to determine the model sensitivities and predicted data at location j. Using the same approach one can also show that for the second matrix-vector multiply

$$z_k = \text{Re}(\sum_{j=1}^{N} \text{Cmplx}(\text{Re}(D_{jj})\ \text{Re}(y_j),\ \text{Im}(D_{jj})\ \text{Im}(y_j))^*\ g_i^t\ K^{-1}\ (\partial s/\partial m_k - \partial K/\partial m_k E_s)), \qquad (31)$$

where N is the number of complex data points used in the inversion and the symbol '*' stands for complex conjugation. Note that even though the summation in equation (31) is over all the data points, parts of the sum could be over different transmitters and/or frequencies, hence E_s will change. Lastly, the derivatives $\partial s/\partial m_k$ and $\partial K/\partial m_k$ in equations (30) and (31) are rapid to

48

compute analytically; it is shown in Appendix F that the vector $\partial s/\partial m_k$ and matrix $\partial K/\partial m_k$ each have 12 non-zero entries when m_k represents either the conductivity or permittivity.

In addition to the forward solves necessary to determine E_a for each source, the matrix-vector multiplies in equations (30) and (31) require solving a series of forward problems corresponding to the total number of unique data measurements locations, where

$$v_j^t = g_j^t \, K^{-1}, \tag{32}$$

or since $K^t = K$ (refer to Chapter II),

$$Kv_j = g_j \tag{33}$$

(note: The fact that K is symmetric is simply a statement of reciprocity). A unique measurement location comprises the measurement of a specific field component made at a site. Thus the total number of forward solves needed for each model update is given by $N_{tx} + N_{rx}$, where N_{tx} and N_{rx} are the total number of transmitters and unique receiver positions used in the inversion.

Handling the Jacobian matrix-vector multiplies in this manner is much more efficient then attempting to explicitly solve equation (26) and using the results to form the matrix-vector multiplies. For example if we are estimating over 30,000 parameters, this would 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 thousand forward solves per model update. Limiting the number of forward solves has also been recommended by McGillivray and Oldenburg (1990) and Oldenburg (1990) because of its efficiency and has been used by Park (1983), Mackie and Madden (1993) and Zhang et al. (1995) in their constructions of the inverse solution.

An Iterative Solution and Selecting the Tradeoff Parameter

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 tradeoff parameter 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 (21). This model is determined once the tradeoff parameter, λ, is selected. To assure a smoothed solution at the first iteration, we select the tradeoff parameter as

$$\lambda = \text{Max Row Sum}(DA^{p(i)} [DA^{p(i)}]^t)/2^{(i-1)}, \tag{34}$$

where i=1 for the first iteration. We have selected this method of choosing λ because it is an estimate of the largest eigenvalue of the non-regularized least squares system matrix. Thus weighting $(W)^t(W)$ by this amount allows only the largest eigenvalues to influence the solution. The maximum row sum is easy to compute and follows from equations (28) and (29) with u selected to be the unit vector.

To digress for the moment, we note that the CG method is designed for linear systems that are symmetric positive definite. While the normal equations in equation (21) are symmetric, both $(DA^{p(i)})^t (DA^{p(i)})$ and $(W)^t(W)$ posses a zero eigenvalue. Thus it appears that the matrix describing the normal equations may be semi-definite. However when $(DA^{p(i)})^t (DA^{p(i)})$ and $(W)^t(W)$ are summed as $(DA^{p(i)})^t (DA^{p(i)}) + \lambda(W)^t(W)$, experience shows the CG algorithm converges provided the tradeoff parameter is reasonably selected. One must avoid selecting λ too large such that non zero elements of $(W)^t(W)$ are much greater than the corresponding elements of $(DA^{p(i)})^t (DA^{p(i)})$ as this will cause a degradation of the convergence rate within the CG algorithm.

We proceed to the next iteration if the data error (sum of square errors) is above χ^2. If this is true the model is linearized again about the new model m, new predicted data and electric fields are computed from the updated background model, and the new model update determined with the tradeoff parameter specified with equation (34). 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 matches χ^2, convergence of the data error 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,

50

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 the scheme (Zhang et al., 1995). For the first and second iterations, 20 and 40 relaxation steps are used, respectively. Subsequent iterations use 60 steps.

MASSIVELY PARALLEL IMPLEMENTATION

EM inversion in 3-D can easily require the solution of at least several hundred forward solves per iteration. In Chapter II we demonstrate how these forward solves can be efficiently computed on an MP machine, where each solve could constitute over five million field unknowns. A significant portion of the storage required to preform the inversion is taken up by the electric field solution vectors produced by these solves and are needed to complete matrix-vector multiplies in the CG routine. Fortunately on the 1840 node Intel Paragon at Sandia National Laboratories it is possible to execute and store all solves without writing to disk; the Paragon has approximately 30 Gbytes of accessible memory.

As determined in Chapter II, the most efficient use of the processors is to divide the problem as close to an equal number of unknowns for which to solve. Because each processor needs only to make calculations for a subset of the forward and inverse problems, and because the processors are making their calculations in parallel, the solution time is reduced by a factor which is approximately equal to the number of processors employed.

The parallelization of the inverse problem is achieved by assigning a given number of

51

processors in each direction of the forward modeling domain (nx in x, ny in y and nz in z). Hence the number of processors dedicated to the problem is determined by nx*ny*nz. The actual estimation of the earth's electrical properties is carried out on the same sets of processors dedicated to the forward problem, with all the processors sharing the same data, but storing different parts of the inversion and forward modeling domain. However, it is possible that some of the processors may not contain portions of the inversion domain and thus will be idle during the CG solve. The reason for this is that cells outside the inversion domain are necessary to keep the boundary of the forward modeling domain at distance (Figure 19). We desire parameter estimates that are not adversely affected by grid truncation errors in the forward modeling.

We now need to address the manner in which the model is input into the parallel machine. The input could constitute a starting model needed to launch the inverse or a restart model in the event of a system crash or if excessive model structure was being incorporated in the inversion. To accomplish this input, we have decomposed the data into two different sets, a global data set and a local data set as in Chapter II.

Communication or message passing amongst the processors will be needed to complete calculations in the inverse problem as well as for the forward problem. Communication amongst processors consists of both the global and local variety. Global communication will be required to treat the five dot products within a generic CG routine and an additional one in equation (30). On the other hand, calculations involving the matrix-vector multiplies require local communication.

Within the inversion three types of local communication will be needed. The first will involve communication of electric field values on processor boundaries such that matrix-vector products in equations (30) and (31) can be completed. This communication will occur before the CG routine is called for efficiency. The second type of communication will involve matrix-vector products of the CG search direction vectors with the regularization matrix times its transpose. This occurs within the CG routine at every relaxation step because 1) we have explicitly formulated the regularization matrix and 2) the CG vectors are constantly updated. The final type of communication occurs after exiting the CG routine. Electrical properties of cells along processor boundaries must be communicated with neighboring processors for proper averaging of electrical properties at cell edges; these averages are needed in subsequent forward-model calculations. After this message passing, calculations with the forward solver can proceed with

the next iteration, given the convergence criteria outlined above.

To deduce the communication pattern of the first type, consider eight nodes located at the corners of a cell whose properties we wish to estimate (Figure 20). Consider the simplest case where each processor is in charge of only one node and cell. For example, node (i,j,k) has the cell in Figure 20 assigned to it as well as the three components of the electric field at $(i+\frac{1}{2},j,k)$, $(i,j+\frac{1}{2},k,)$ and $(i,j,k+\frac{1}{2})$. To complete its calculations, the processor that owns this node and cell also needs the electric fields on the cell edges assigned to other nodes on different processors. These processors will thus need to supply the field components. Furthermore the processor that owns the node (i,j,k) may also have to send its electric field components to nodes on other processors. For example, node $(i-1,j,k)$ will require the y component of electric field assigned to node (i,j,k).

The pattern for the second type of communication can be obtained from Figure 21. The stencil shows the required coupling between the center cell and its neighbors arising from the Laplacian operator, as applied in the regularization matrix-vector multiplies. Again consider the case where each processor contains only a single cell. To complete its local version of the matrix-vector multiply, the center processor needs components of the CG search direction vector which are assigned to the other cells and hence processors. In addition to this, the processor holding the center cell will also be required to send components to the neighboring processors so that they can complete their corresponding computations.

From Figure 22 the final communication pattern can be inferred. Consider the computation of the average electrical properties at cell edges $(i+\frac{1}{2},j,k)$, $(i,j+\frac{1}{2},k)$ and $(i,j,k+\frac{1}{2})$, which are assigned to node (i,j,k). The electrical properties of the four cells that form each edge will be needed and the computation at these positions will be carried out on the processor that holds the solid cell also assigned to node (i,j,k); additional cells that are required are indicated by the dashed outlines. Let us now consider that each node, cell, and its associated electrical properties belong to a different processor. Since the dashed cells belong to different processors, their electrical properties need to be passed to the processor (indicated by solid cell) that will compute the averages. In addition this processor will be required to send its electrical properties. Consider computing average electrical properties at location $(i+\frac{1}{2},j+1,k)$. Since this computation is carried out on a different processor, the electrical properties assigned to the solid cell in Figure 22 will be needed.

The local communication pattern for the inverse problem can now be summarized in Figure 23, where each cube represents a different processor with subsets of nodes and cells assigned to it. For the matrix-vector multiplies involving the Jacobian matrix and its transpose, communication consists along the faces of processors as well as along edges. Specifically information is passed from the central processor (marked by the heavy outline) to those neighbors that are dashed in Figure 23. Likewise those neighboring processors with solid boundaries 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 23, where all the processors send the required elements of the CG vectors to the central processor as well as receive from it. Finally, the communication needed for averaging electrical properties of the cells at adjacent processor boundaries is an opposite sense compared with the communication of the Jacobian matrix vector multiplies. Those face and edge processors marked with a dashed outline send to the central processor, while those that are solid receive information from it. To provide for the required message passing we have chosen as in Chapter II to employ Massage Passing Interface (MPI, Skjellum et al., 1993), instead of using machine specific commands.

As previously mentioned the solution time will decrease with the number of processors employed. This is demonstrated in Figure 24 for an example described in Chapter IV of this report. A significant speed up is observed starting from eighty processors for a single iteration of the inverse algorithm. However as the number of processors continue to increase inter-processor communication becomes more of a factor, resulting in an asymptotic behavior in the solution time with increasing number of processors. Here the amount of message passing will eventually limit the speed at which the computation can proceed. Put simply, increased message passing implies more time communicating and less time computing. Thus optimal use of the machine may entail running the example in Figure 24 using 200 processors and launching several of such jobs simultaneously. On the other hand, if turn around time is an issue, then one would want to operate near the far right end of the curve.

SYNTHETIC EXAMPLE

Figure 25 shows two different perspectives of 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 fully independent. 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, with 15 vertical magnetic dipole (VMD) transmitters at 10 m intervals straddling the target. The vertical magnetic fields were calculated in all other wells at 10 m intervals, excluding the transmitter well. 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 throughout the model and set to that of free space. Gaussian noise equal to two percent of the data amplitude was added to the data set. The data were then weighted by this percentage before inversion. In total, the data consist of 12 600 transmitter-receiver pairs.

The inversion domain consists of 29 791 cells, but only 13 824 cells are shown in the interwell region in Figures 25 and 26; cells outside this region are used to keep the boundary of the inversion domain at distance so as to not affect the conductivity estimates within the interwell region. The inversion, which was launched assuming a 0.005 S/m whole space, has recovered fairly well the location and geometry of the cube, but a smeared version of its conductivity within the cube boundary; the estimates vary from 0.1 to 1.0 S/m. The conductivity estimates of the background range as low as 0.0016 S/m. It has been our experience that improved resolution of the background and cube 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.

Eleven iterations were needed to obtain this reconstruction, where the reduction in relative error against iteration count is illustrated in Figure 27. 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 27 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. These data were produced from a forward modeling algorithm that is different from the one used in the inverse. Finally the processing time needed to produce the image in Figure 26 was approximately 24 hours on the Paragon, with 512 processors utilized.

DISCUSSION

The MP inversion scheme we have presented has been demonstrated on a data set that would be impossible to invert on scalar workstation platforms due to memory and processor speeds (refer to Chapter IV for additional examples). An important question to ask is what is the largest model the MP inversion can handle? Certainly the maximum model size (both forward and inverse) will be related to the number of transmitters and receivers specified in the data set because this will determine the number of electric field vectors, E_s, that need to be computed and stored. Given the maximum memory on the Intel Paragon of 16 Mbytes per processor, and considering a problem divided amongst 1728 processors (this corresponds to 12 processors assigned along each coordinate direction), Table 2 illustrates a range of problem sizes that can be effectively handled. If 120^3 nodes are used to describe the forward and inverse modeling domain, the number of transmitters and receivers that can be used is 700. To increase the number of transmitters and receivers it appears necessary to reduce the number of nodes.

Problem Size(Nodes):	120^3	96^3	72^3
#Tx's and Rx's:	700	1300	3000

Table 2. Maximum problem size that can be treated by the Intel Paragon assuming 1728 processors. Problem size is determined by the number of nodes or cells used in the forward modeling and inversion and the number of transmitters (Tx's) and unique receivers (Rx's) specifying a data set. Each Tx and Rx position is for a unique frequency.

One way to increase the size of inverse problems that can be tackled is to skeletonize the inversion domain, but retain the same parameterization level in the forward modeling domain. The key idea here is to reduce the storage of the electric field vectors needed in the inverse. For a given source, the electric field and predicted data are computed at the parameterization level specified in the forward modeling. The electric field is then interpolated to the skeletonized grid corresponding to the inverse and stored in memory. Hence the forward modeling accuracy is still retained in the inverse. Note that the coarser grid can still produce smooth images since it can

involve tens of thousands to hundreds of thousands of cells.

The skeletonized electric field vectors allow for the number of transmitters and receivers to increase dramatically. 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 skeletonized inversion grid, which still comprises 216 000 cells, allows for the number of transmitters and receivers to increase from 700 to over 3000.

CONCLUSIONS

A 3-D EM inversion code has been successfully implemented and tested on an MP platform. Reasonable, overnight to full day processing times have been obtained. Because of the MP platform, reconstructions have been produced that do not underparameterize the earth; these are reconstructions that 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 (cf. Torres-Verdin and Habashy, 1995 and 1994; Zhandnov and Fang, 1995a; Habashy et al., 1995; Farquharson and Oldenburg, 1995).

In this chapter, we have presented the theory and demonstrated the 3-D inversion capability on synthetic data. Because the ultimate goal of any inversion scheme is to use it to image field data, in Chapter IV we demonstrate how this scheme can be used to design a 3-D crosswell survey and invert a crosswell data set collected at the Richmond field station north of Berkeley California. Images before and after the injection of a salt water plume will be compared to determine the location of the injected plume. In addition, we will also show how the scheme can be employed to analyze the reliability of the images as well as the accuracy and errors in the data.
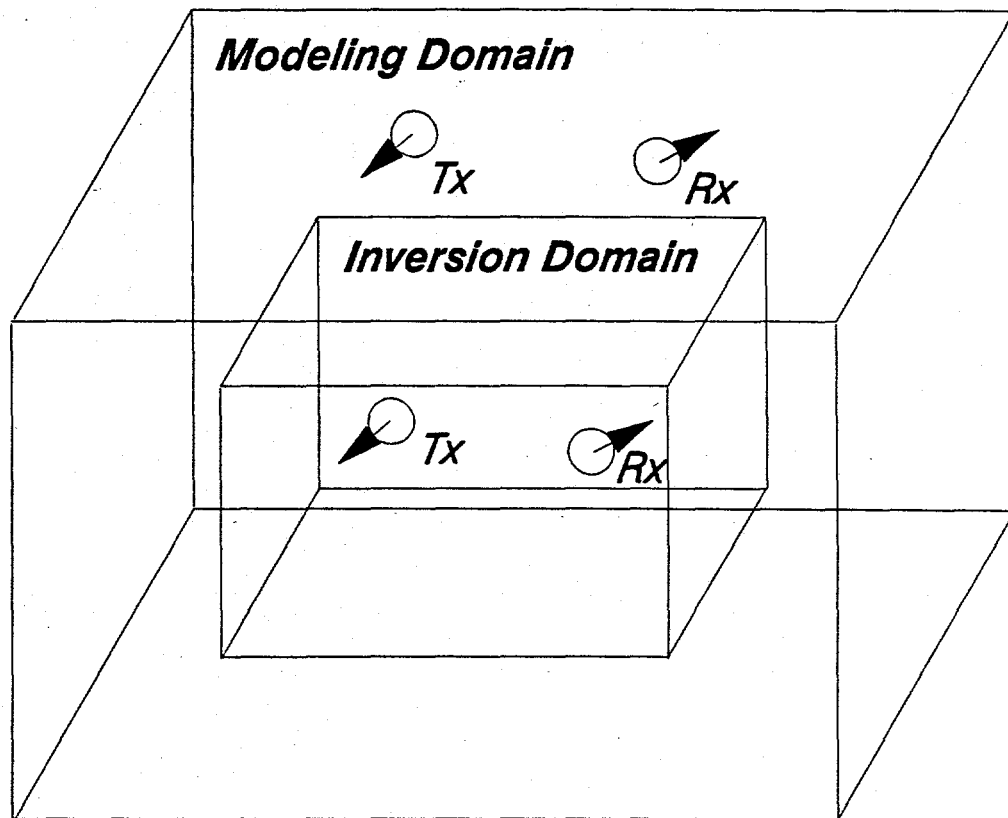
Figure 19. The inversion domain is a subset of the forward modeling domain because of forward-modeling errors near grid boundaries. Transmitters and receivers can be placed either inside or outside the inversion domain. External transmitters and receivers could correspond to surface or airborne configurations, while internal sources and receivers could correspond to cross-well configurations.
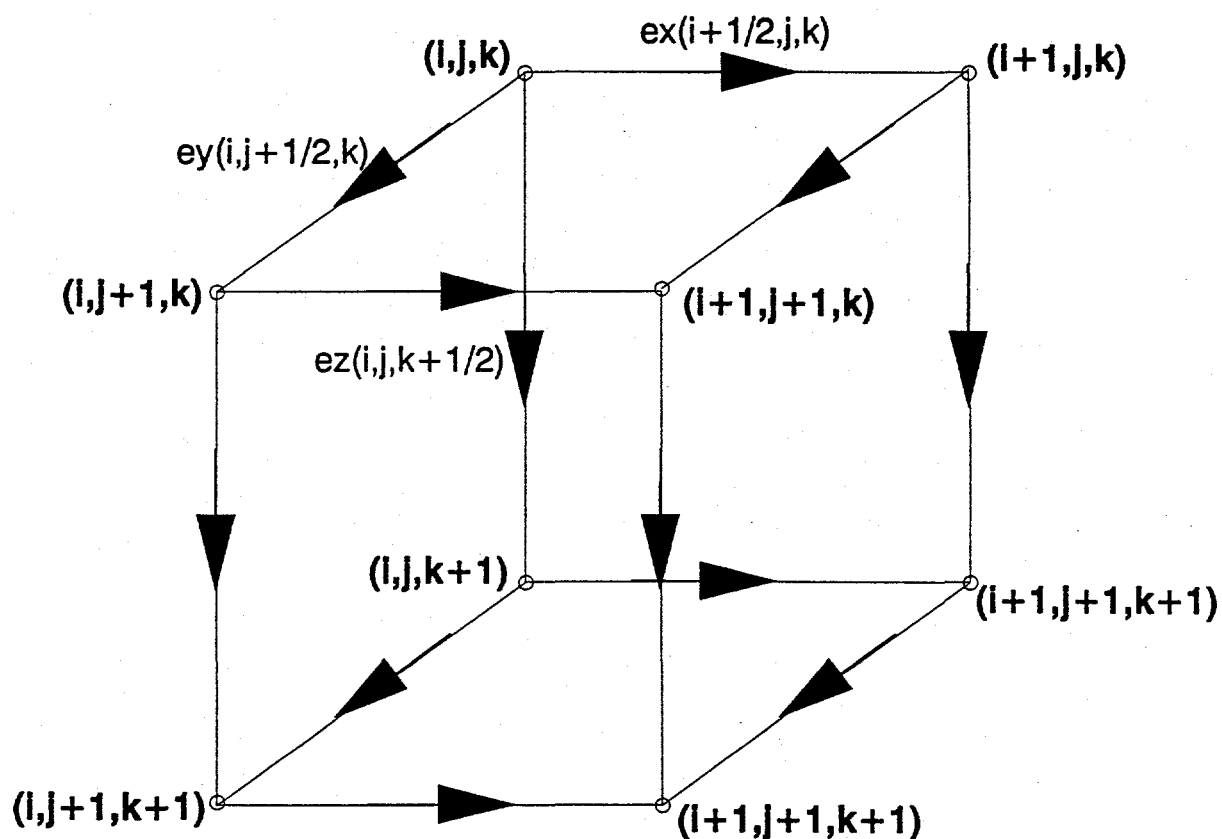
Figure 20. The electric field stencil needed to complete the Jacobian matrix-vector multiplies in the inverse for a single cell. Node (i,j,k) has the cell and the x, y and z electric field components assigned at (i+½,j,k), (i,j+½,k) and (i,j,k+½), respectively. Assignment of other electric field components to other nodes as shown in the figure follows analogously. Using results for the single cell, a processor map can be developed to carry out the required local communication amongst the processors.
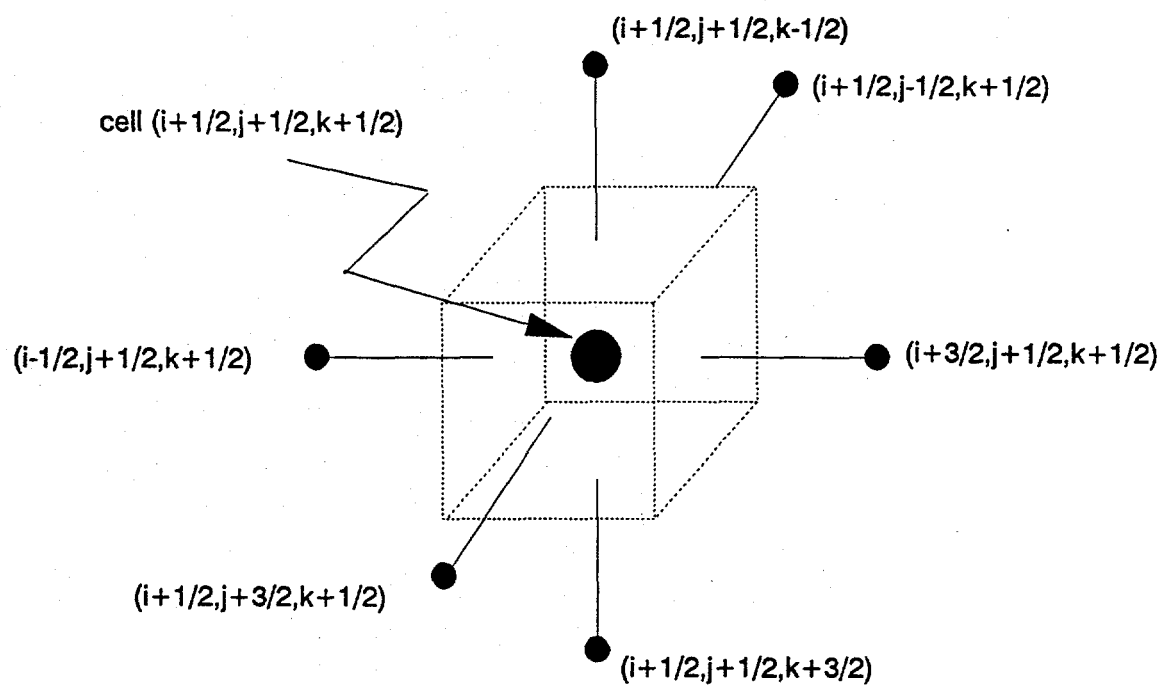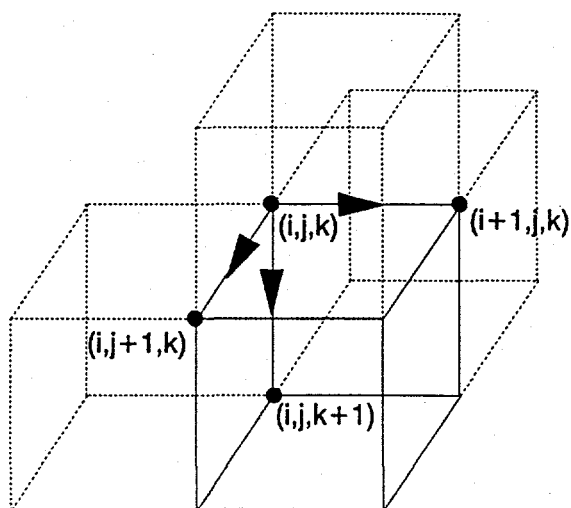
Figure 21. The stencil needed to complete local regularization matrix-vector multiples in the CG routine. Using results for the single cell assigned to a single processor, a processor map can be developed to carry out the required local communication amongst processors.

Face Cells

Edge Cells

Figure 22. The different cells needed to compute average electrical properties at edges $(i+\frac{1}{2},j,k)$, $(i,j+\frac{1}{2},k)$ and $(i,j,k+\frac{1}{2})$. These edges, as well as the solid cell are assigned to node (i,j,k). The additional face and edge cells needed to compute average electrical properties are indicated by the dashed outlines. Using results for the single processor and cell, a processor map can be developed to carry out the required local communication amongst processors, necessary for subsequent forward model calculations.
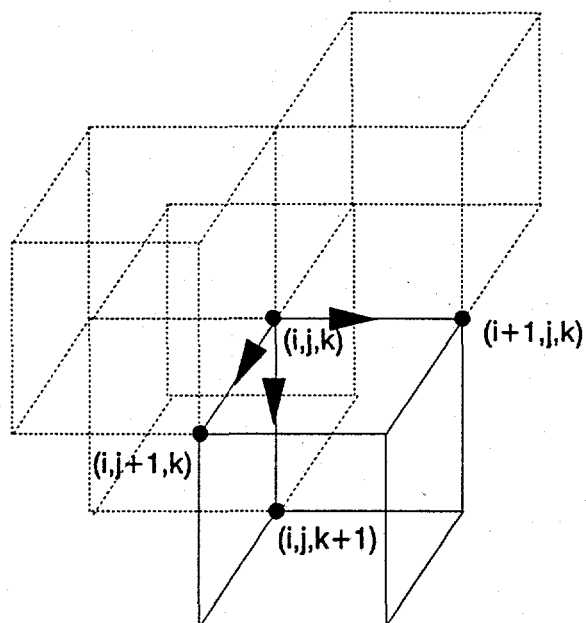
61

Face Communication
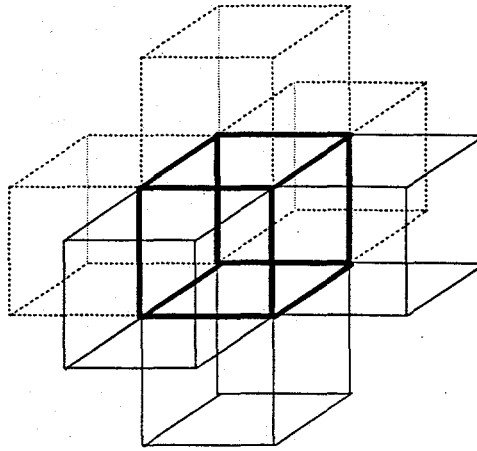
Edge Communication
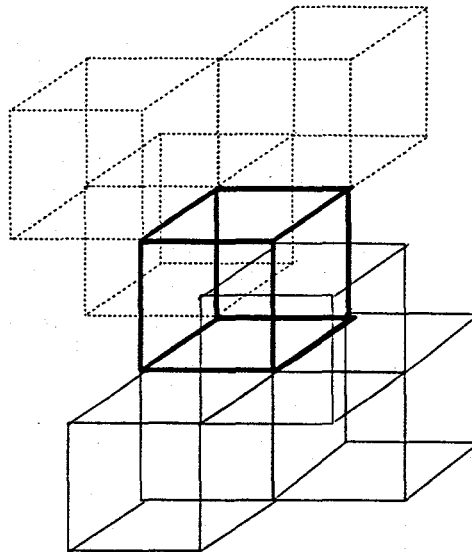
Figure 23. Local processor communication scheme used in the 3-D MP inverse. The solid cube depicts the central processor that is sending and receiving from its neighbors. Both face and edge communication patterns are indicated.
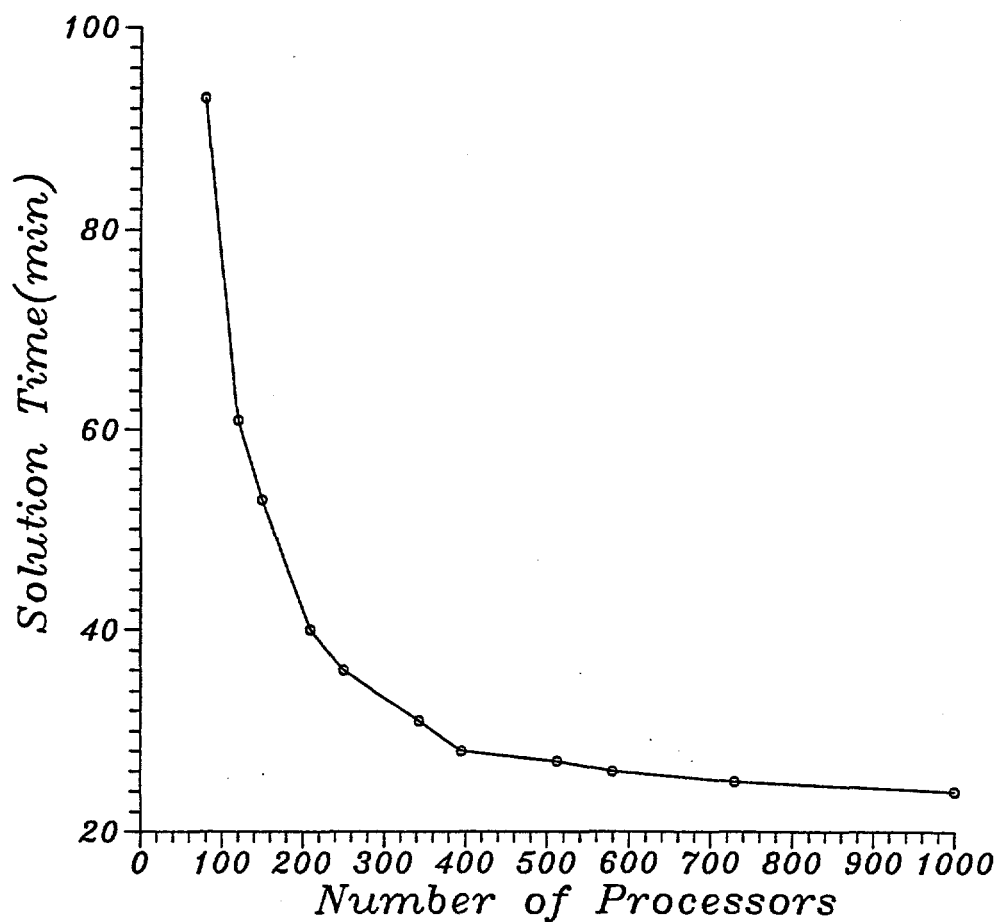
Figure 24. Solution time for one iteration of the inversion versus number of processors employed. Results are for the 8 well Richmond model used in the design experiment discussed in Chapter IV of the report. The model is discretized at 114 000 cells for the forward model calculations with the inverse parameterization using 88 200 cells. The total number of transmitters-receiver pairs used in the inversion is 1 848.
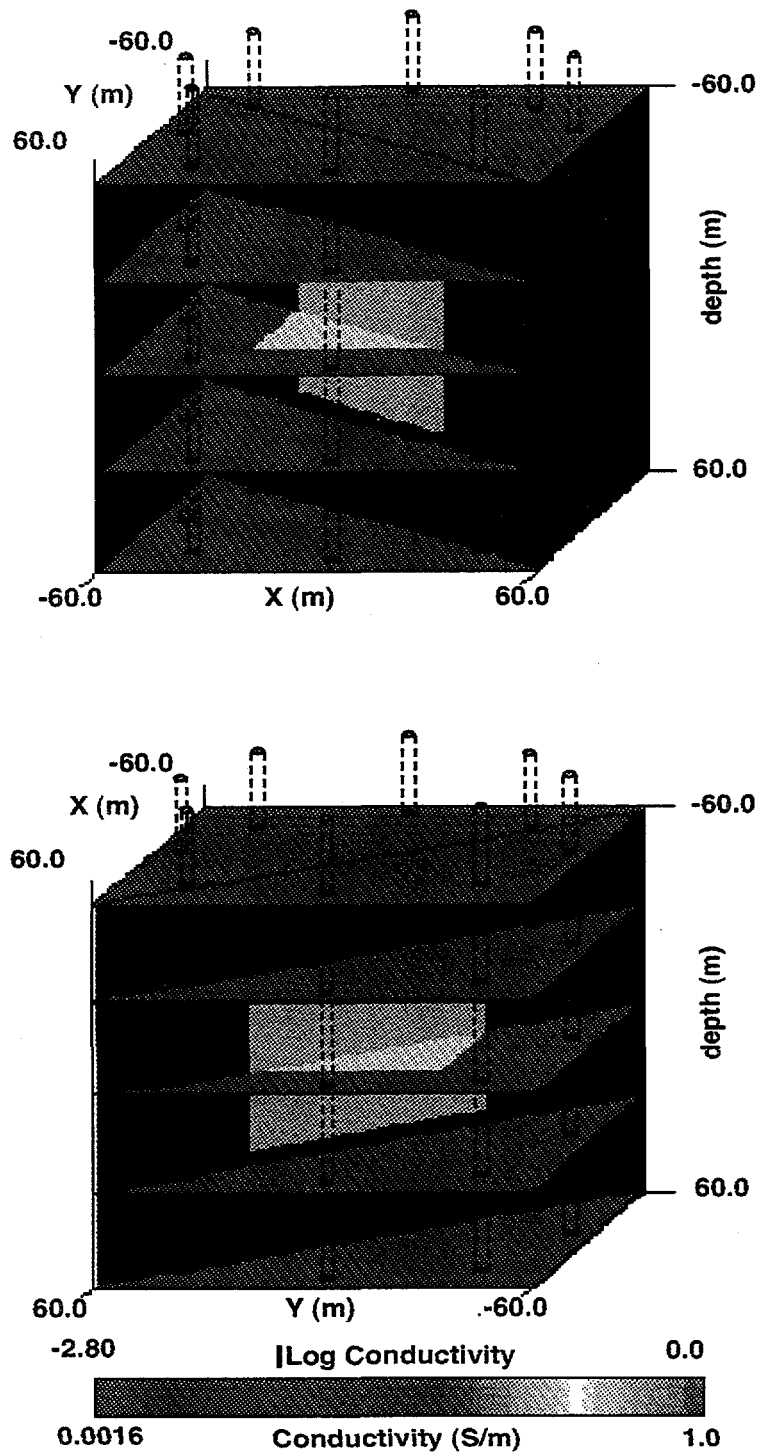
Figure 25. Synthetic example, with wellbores, used to test the inversion algorithm. The data were calculated from this model using an integral equation solution. Model is shown for different slices from two different perspectives.
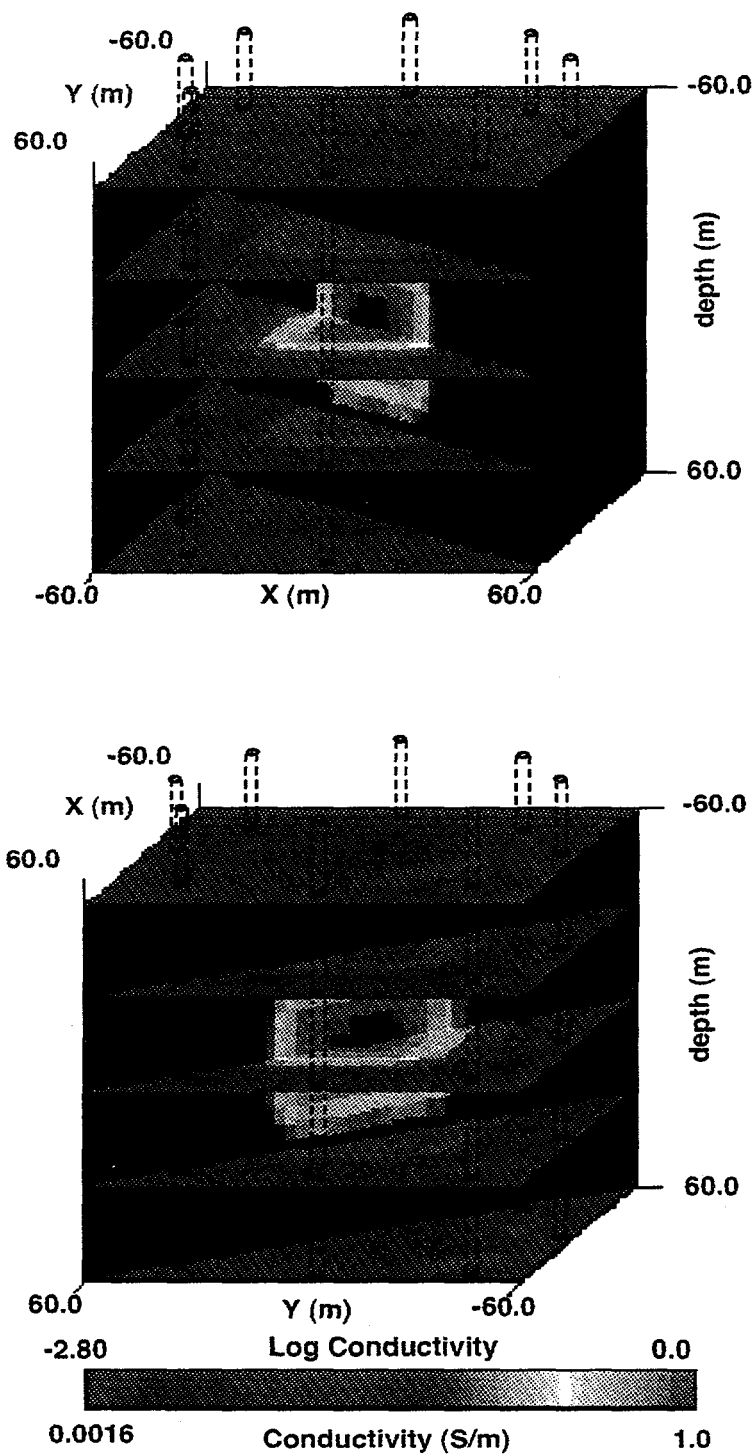
Figure 26. Reconstructed conductivity for the synthetic example illustrated in Figure 25. The wellbores used in the simulation are also indicated.
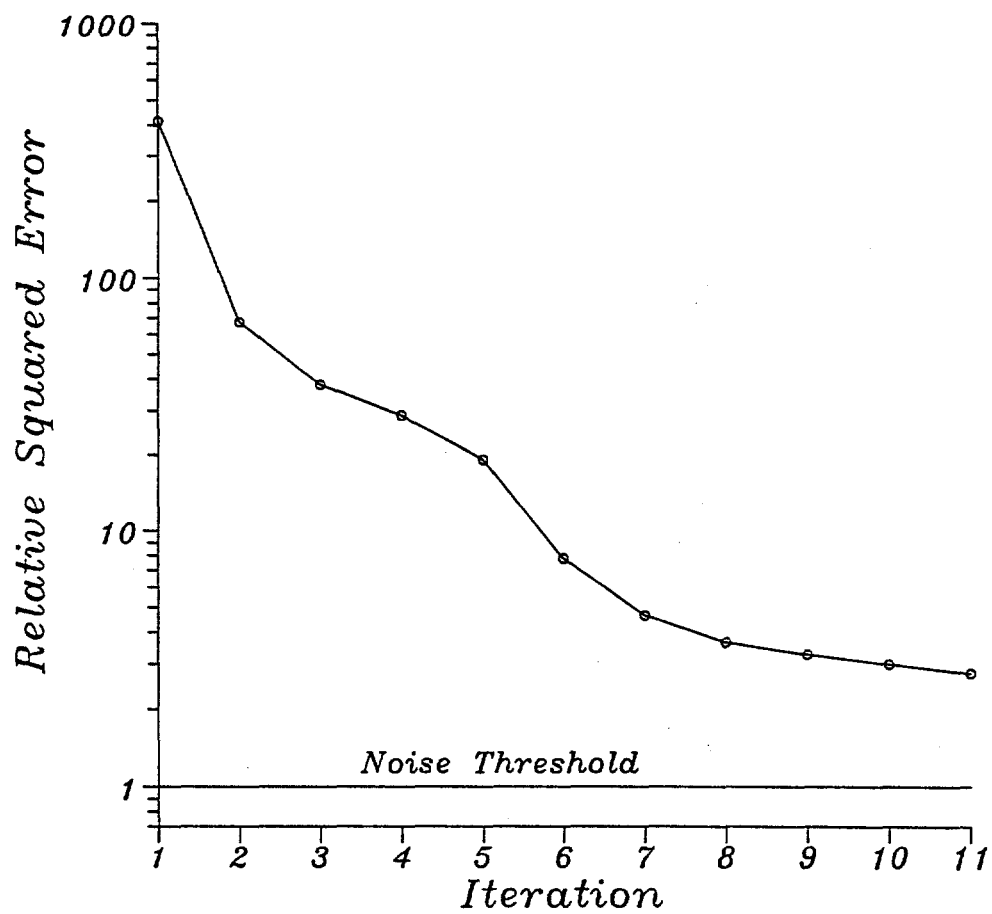
Figure 27. Sum of squared error plotted against iteration number for the 0.2 S/m test body shown in Figure 25. The squared error has been normalized by the variance of the noise.

66

CHAPTER IV

ANALYSIS OF A 3-D CROSS WELL EM EXPERIMENT

Although synthetic examples such as the one presented in the last chapter illustrate the theoretical accuracy and limitations of a geophysical inversion algorithm, the real test of a scheme's usefulness, versatility and robustness comes when data collected at a field site are inverted to produce an image of the subsurface. In general the characteristics of the noise are much different than the Gaussian noise assumed in synthetic tests and the data tend to be more sparsely sampled due to survey time constraints than one would like. In addition with certain types of surveys, such as a crosswell EM survey, the unknown accuracy of the source and receiver locations may provide additional sources of error.

To illustrate the benefits provided by the full 3-D inversion code, as well as how these problems affect the images, we have inverted a crosswell data set that was collected to monitor the injection of a volume of salt water at depth. The versatility of the scheme is demonstrated by not only producing 3-D images of the subsurface, but by also employing it for pre-imaging experimental design and resolution analysis, as well as post-imaging error analysis. In addition, the benefits of the massively parallel computer platform is demonstrated by the quick turn around time for the images as well as the number of imaging experiments that can be accomplished within a given time period.

THE RICHMOND FIELD STATION EXPERIMENT

The University of California's Richmond Field Station has been the location of a series of salt water injection monitoring experiments since 1988, the purpose of which have been to simulate both an enhanced oil recovery water flood as well as the injection of a contaminants and/or tracers into an aquifer. The site contains several monitoring wells (Figure 28) as deep as 80 m which show the geology to consist of unconsolidated alluvium to depths of 30 to 35 m overlying a basement of sandstone and shale. The alluvium consists of muds and silts interbedded with layers of sand and gravel and well logs show this upper section to have conductivities ranging from 0.2 to 0.02 S/m. The basement tends to be more resistive with conductivities as low as 0.001 S/m. In addition the logs indicate that an unconformity of some type exists in the basement between the Inj and NW wells. This unconformity is likely due to either steeply tilting stratigraphy or a fault.

Although more thorough descriptions of the experiment can be found in Alumbaugh and Morrison (1995) and Wilt et al. (1995a), for completeness a brief description is given here. In May of 1992, crosswell electromagnetic measurements were made by placing a vertical magnetic dipole (VMD) source operating at 18.5 kHz in the central well, and making vertical magnetic field measurements in the four surrounding wells (Figure 28) using the system described by Wilt et al.

67

(1995a). The measurements were made at 5m receiver intervals from 5m to 60m depth which yields 11 receiver positions per well. A similar range of source depths was employed using a sampling interval of 0.5m.

After this baseline set of measurements had been completed, 50 000 gallons of 1 S/m salt water was injected into a gravel aquifer at 30 m depth through the center well in Figure 28. A second set of measurements were then completed using the same parameters as in the baseline survey. In addition, post-injection repeat data sets were collected between the injector and the NW well, and the injector and the SW well at a time interval from the original surveys of one week and two weeks, respectively. The purpose of these repeat measurements was to better quantify the noise characteristics of the data. The experiment ended in June of 1992 prior to pumping out the salt water.

The purpose of this particular experiment was to analyze how well the plume location could be determined through crosswell EM tomography. Alumbaugh and Morrison (1995a), Liu et al. (1995) and Wilt et al. (1995a) all employ EM inversion schemes that assume a 2-D cylindrical geometry in which the geology is symmetric about the borehole containing the source. In all three cases the plume was clearly shown to be migrating in a northerly direction. In addition, Newman (1995) employs a 2.5D geometry in his inversion scheme which yields similar conclusions for the data collected between the injector and the NW well. These conclusions all agree with surface-to-borehole dc resistivity measurements made by Bevc and Morrison (1992).

Although these images were very successful in mapping the general migration route of the plume, questions still remain about the 3-D shape of the plume. In addition, as demonstrated by Alumbaugh and Morrison (1995a), assuming a 2-D geometry can impose artifacts in the images if the geology does not fit the 2-D assumption. Thus to better describe the shape of the plume as well as more accurately resolve the geology, a 3-D inversion scheme is needed.

EXPERIMENT DESIGN AND RESOLUTION ANALYSIS

One of the critical questions when trying to image geologic structure is what acquisition parameters are needed to adequately resolve structures in the subsurface? If the data have already been collected then the question becomes given the survey acquisition parameters, what resolution can we expect to achieve from the data? Thus before inverting the Richmond Field Station data we shall employ the 3D inverse to try and answer these questions. However, because the experiment actually took place before these simulations could be conducted, the focus here will be to predict how well the plume and assumed electrical structure of the site can be recovered, and what improvements could be made to the survey configuration to improve the resolution.

Figure 29 shows the synthetic model employed in this simulation. Horizontal slices of the electrical conductivity from the zero to 60 m depth illustrate conductive sediments overlying

resistive basement. The finer structural features that we are trying to resolve include the conductive salt water plume at 30m depth within a thin (4m thick) aquifer, a vertical contact between resistive units within the basement, and a thin (4m thick) conductive channel at 22m depth within the overburden at the location of the SE well. The basement contact and the channel have been included in the model using information from the northeast and southeast well logs. A source sampling interval of 2.5 m was employed from 5m to 60m depth as will be employed when imaging the data, and a receiver sampling of 5m was used as in the experiment. This yields a total of 924 data points. The synthetic results were calculated using the finite difference code described in Chapter II. (Note: we have not included the air-earth interface in this example. The exclusion of this boundary is discussed in more detail in the next section.)

Because all data contain some type of error, a necessity for producing accurate images through data inversion is to estimate the quality of the noise. To do this we have analyzed the two repeated sets of data taken after the injection. Wilt et al. (1995a) present the average errors for these repeat data to be 2.2% in amplitude and 0.8 degrees in phase, and 3.3% in amplitude and 1.1 degrees in phase for the INJ-NW and INJ-SW repeats, respectively. A more rigorous way to analyze the noise is to look at the mean and standard deviation of the errors as a function of common source-receiver offset. As shown in Figure 30, this type of analysis shows the repeatability noise to decrease at the same rate as the signal amplitude for shorter offsets, and then become approximately constant for longer offsets. This implies that at these larger offsets the noise as a percentage of the data is going to be much larger than at the shorter offsets. The noise model that we have employed here is based on this analysis and assumes random Gaussian noise with a standard deviation equal to 2% of the magnetic field for amplitudes greater than 1×10^{-6} A/m, and a standard deviation equal to 2×10^{-8} A/m when the field drops below this value. The data were then weighted within the inversion scheme with this noise distribution, which effectively downweights the longer offset data.

The image resulting from the simulation is given in Figure 31, and in Figure 32 we have plotted the average residual error as a function of the iteration number. Notice that the residual decreases smoothly and flattens out as it approaches the estimated noise level. The forward and inversion domains consist of 46x46x54 and 42x42x50 cells, respectively, and thus a total of 88 200 cells were used to estimate the electrical conductivity. However for compactness only 34 000 cells within the region of interest are shown in Figure 31. To run this using 512 processors of the Intel Paragon took approximately 6 hours, or 1/2 hour per iteration where the inversion was launched assuming a whole space of 0.033S/m.

Figure 31 shows that we have recovered the general geology of conductive sediments overlying resistive basement as well as the location of the plume remarkably well. However, notice that the sharp edges of the plume have not been recovered; this is to be expected from a scheme that imposes smoothness constraints on the solution. In addition notice that neither the

channel structure, the aquifer nor the contact within the basement are imaged very well. We feel that the former is caused by too large of a receiver sampling interval in the vertical direction. We have deduced this because the plume and the channel are both 4m thick. However the source sampling interval within the plume is 2.5m, while the receiver sampling interval within the channel is 5m. The plume is recovered by the finer sampling interval while the channel is not. Thus the receiver sampling interval needs to be decreased in order to recover finer structures.

The inability to resolve the aquifer may also be attributed to insufficient receiver sampling. However the aquifer is a resistive target of fairly low contrast with respect to the background when compared to the conductive targets. Thus because cross well EM is an inductive method, the aquifer may not be generating enough of a scattered response to be resolved.

With regards to the basement contact, we believe that the resolution can be improved if a higher frequency is employed. This is based on the fact that the basement structure is fairly resistive such that there is only 1/2 skin depth of attenuation between the wells compared to 1.5 to 2 skin depths attenuation in the conductive sediments. As demonstrated by Alumbaugh and Morrison (1995b), greater attenuation implies greater resolution. Because an increase in frequency implies an increase in attenuation, a higher frequency should provide better resolution.

However, part of the problem in accurately recovering the basement structure, especially the formation of the relatively conductive artifact near the source well in Figure 31, can be rectified by employing more complete data coverage around the imaging region. This is illustrated in Figure 33 where we have simulated and imaged a data set that includes four additional receiver wells between the four original wells. This example took 7 hours to run using the same number of processors as above. Notice that the artifact near the central well has disappeared, and that the contact though distorted is better defined. In addition there is a minor improvement in locating the position of the channel as well as the conductivity distribution within the plume. However notice that the general shape of the plume has not been altered. Thus, although we may have problems with the Richmond data in imaging structures finer than the source and receiver sampling intervals and fully recovering the basement structure, we should be able to accurately locate the position of the plume which was the primary objective of this experiment.

ELECTRICAL RECONSTRUCTIONS OF THE FIELD DATA

Initially the data were weighted using the same noise estimates as used for the synthetic examples in the previous section. Unfortunately, the normalized residual error refused to converge when this noise model was employed, which indicates that we were giving to much weight to poor quality data. Thus we chose to weight the data by two percent of the maximum amplitude for each source relative to all the receivers in a given observation well. This type of weighting puts more emphasis on short offset positions compared to the long ones; because the data quality tends to be

better when the source and receiver are close together, this gives greater weighting to good data. This weighting scheme was found to allow the inversion to converge.

A second major difference between the Richmond data and the simulations above is that the measurements at Richmond were made in the presence of the air-earth interface. Because the inclusion of a half space interface is computationally more demanding, employing this as a background model could cause the inversion scheme to run much slower. To determine if we needed to employ a half-space rather than whole-space within the inversion, a synthetic data set was calculated for the model in Figure 29 which contained this boundary, and then this data inverted assuming a whole space. It was demonstrated that the half-space effects rapidly decreased with increasing depth and thus it was determined that we could launch the inversion assuming a 0.033S/m conductive whole-space.

The image resulting from the pre-injection data is given in Figure 34, the post-injection data in Figure 35, and the average residuals for the inversions in Figure 36. These each took approximately 4 hours to run using 512 processors of the Paragon. Both images clearly show the conductive overburden as well as the resistive basement. In addition, the plume is evident in the post injection image (Figure 35) at 30 m depth to the north of the injector well. To emphasize the plume we can subtract the pre-injection image from the post-injection as illustrated in Figure 37. Although the plume is evident from 26 to 32 m depth which corresponds to the injection interval, the thickness is not evident here because we have only included selected depth slices.

Conductive zones also appear within the overburden in Figures 34 and 35 between the receiver wells where there is no data coverage. Because these did not appear in the synthetic example (Figure 31) one would tend to imply that these zones represent structure. One alternate suggestion is that the different weighting scheme used to invert the data has somehow caused less sensitivity to these regions resulting in artifacts. To rule this out we inverted the synthetic data employing the same data weighting scheme as employed in this section and found no generation of these phenomena. A second more plausible explaination is that these are artifacts caused by inaccurate knowledge of the source and receiver locations. To demonstrate how this idea was developed, we next examine the residual errors between the measured and predicted data for each individual source-receiver combination.

ERROR ANALYSIS

Alumbaugh and Morrison (1995b) show that by plotting the residual error as a function of source and receiver position, one can determine if there exists a non-random bias either in the data or in assumptions made within the imaging scheme. In that particular example they show how large biases in the residuals and artifacts in the images can result if a 2-D model geometry is employed within the inversion for imaging data generated in a 3-D media. Here we can be fairly

7 1

confident that because we are using a full 3-D model geometry this type of bias will not exist. Thus any non-random component of the residual that does appear is more likely to be due to errors in the data collection process.

Before presenting the residual errors between the Richmond field data and the predicted data of the final images, it is instructive to investigate the residuals from the resolution analysis section. Here the plotted values are simply the difference between the measured and predicted data normalized by the weighting. Figure 38 shows the residuals for the image in Figure 31 when the receiver is in the NW well. Notice the random pattern which indicates there is no bias in the data. This is to be expected because the noise that was added is random in nature. However, as illustrated in Figure 39 this is not the case for the post-injection data collected in the Richmond experiment. (The pre-injection data residuals demonstrate almost and identical response and are not shown here.) The residual gradually increases with increasing source and receiver depth such that the largest errors are observed when both sensors are located in the basement. Further, the southeast well has the largest residuals, which can be as large as 10. Because the data weighting was 2% of the maximum amplitude, this normalized value is equivalent to 20% amplitude error. Note that the residuals in the imaginary component of the data are always less than corresponding real component. This is expected because 1) the high resistivity of the basement coupled with the fairly low frequency ensures that the measurements were made in the quasi-static field so that the real part of the field is much larger than the imaginary and 2) the weighting was based on data amplitude rather than the real and imaginary components separately.

The clustering of large residual values in the basement indicates that the data are some how biased as the source and receiver get deeper into the earth. One possible source of this error could be a deviation in the wells from the assumed vertical orientation. Numerical experiments with a two layered-earth model (conductive overburden and resistive basement) showed that a similar bias can be produced if the predicted data are calculated assuming the wells are straight but the measurements are made in boreholes that are deviated. A deviation as little as 1/2 to 2 m at the bottom of the wells was found to produce errors on the same order of magnitude as those observed.

To further test this idea, the data employed in the resolution analysis portion of this chapter were again run through the inversion scheme assuming two different scenarios: (1) the bottom of the observation wells were assumed to be deviated in towards the injector well and (2) the bottom of the wells were assumed to be deviated outward away from the injector. This was accomplished by taking the synthetic data that were calculated assuming vertical wells and altering the horizontal locations of the receivers with depth in the inversion. Specifically the inward or outward receiver deviations at each depth were calculated by assuming zero horizontal offset at the surface and linearly varying it downward such that at 60m depth the horizontal offset was 2 m for the southeast

well, 0.2 m for the southwest well and 1 m for the northeast and northwest wells. These values were selected based on the size of the data residuals observed in the field experiment. Here the data weighting scheme employed was identical to that used in the inversion of the field data.

The data residuals for the two scenarios are illustrated in Figures 40 and 41, while the resulting images are given in Figures 42 and 43. The residual plots show a bias along the diagonal near the bottom of the wells which in general is very similar to those of the data. In addition notice that if the receivers are assumed to be closer to the source than is actually occurring, the residuals are positive along the diagonal. If the reverse is true the residuals are negative. Thus we can crudely determine the errors in positioning of the receivers with respect to the transmitter. However, because the injector well could also be deviated we can not determine the exact location of the sensors with this analysis. Finally, the electrical reconstructions for the two scenarios in Figures 42 and 43 clearly show the presence of artifacts resulting from the sensor location errors that are similar to those observed in Figures 34 and 35. However, notice that the plume is still fairly accurately located.

CONCLUSIONS

The 3-D inversion scheme has been successfully employed to image data collected in a crosswell EM experiment. The scheme has not only provided images of the site, and especially of the injected salt water plume, but also has been demonstrated to be of great use in defining the resolution of the experiment as well as what type of errors are present in the data. From this analysis we have determined that 1) the plume is fairly well resolved with this configuration and 2) the wells at the Richmond Field station are most likely not vertical, but rather are deviated with increasing depth. In addition the speed and versatility of the massively parallel computer platform has been demonstrated; jobs that only take 3 to 4 hours on the Intel Paragon could take as long as a week on a high end workstation.

In the near future we will be analyzing other data sets that employ different source and receiver configurations. Although we are currently limited by the number of sources and receivers that can be employed, we are currently looking at methods to remedy this situation as outlined in chapter III. Finally, we would like to suggest that any geophysical imaging experiment that includes measurements within boreholes should have the wells accurately surveyed in to determine if any deviations exist. As illustrated here, this can have a large impact on the accuracy of the resulting images.
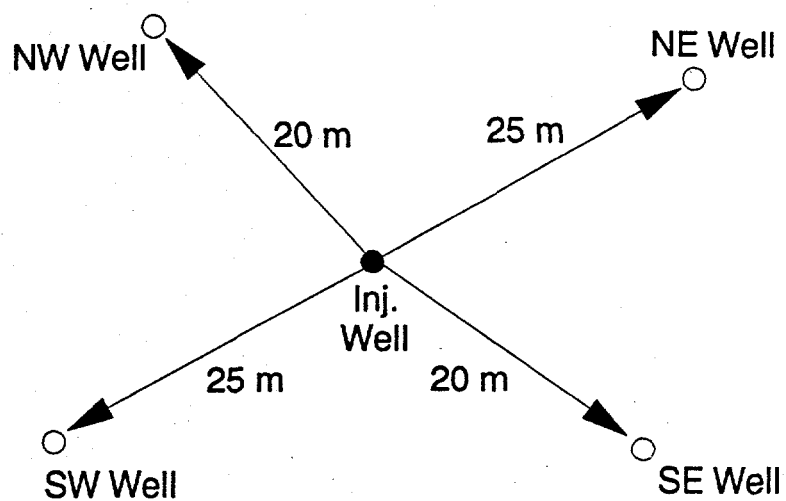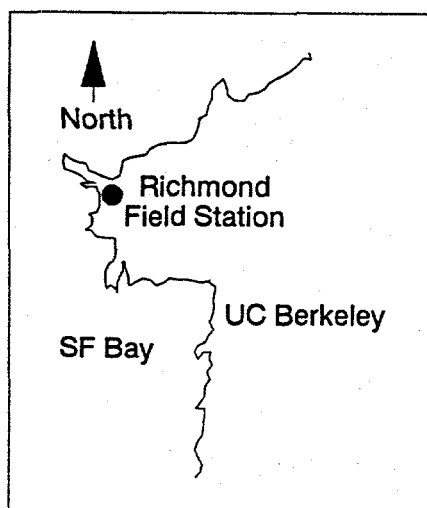
Figure 28. Layout of the Richmond-field site experiment.
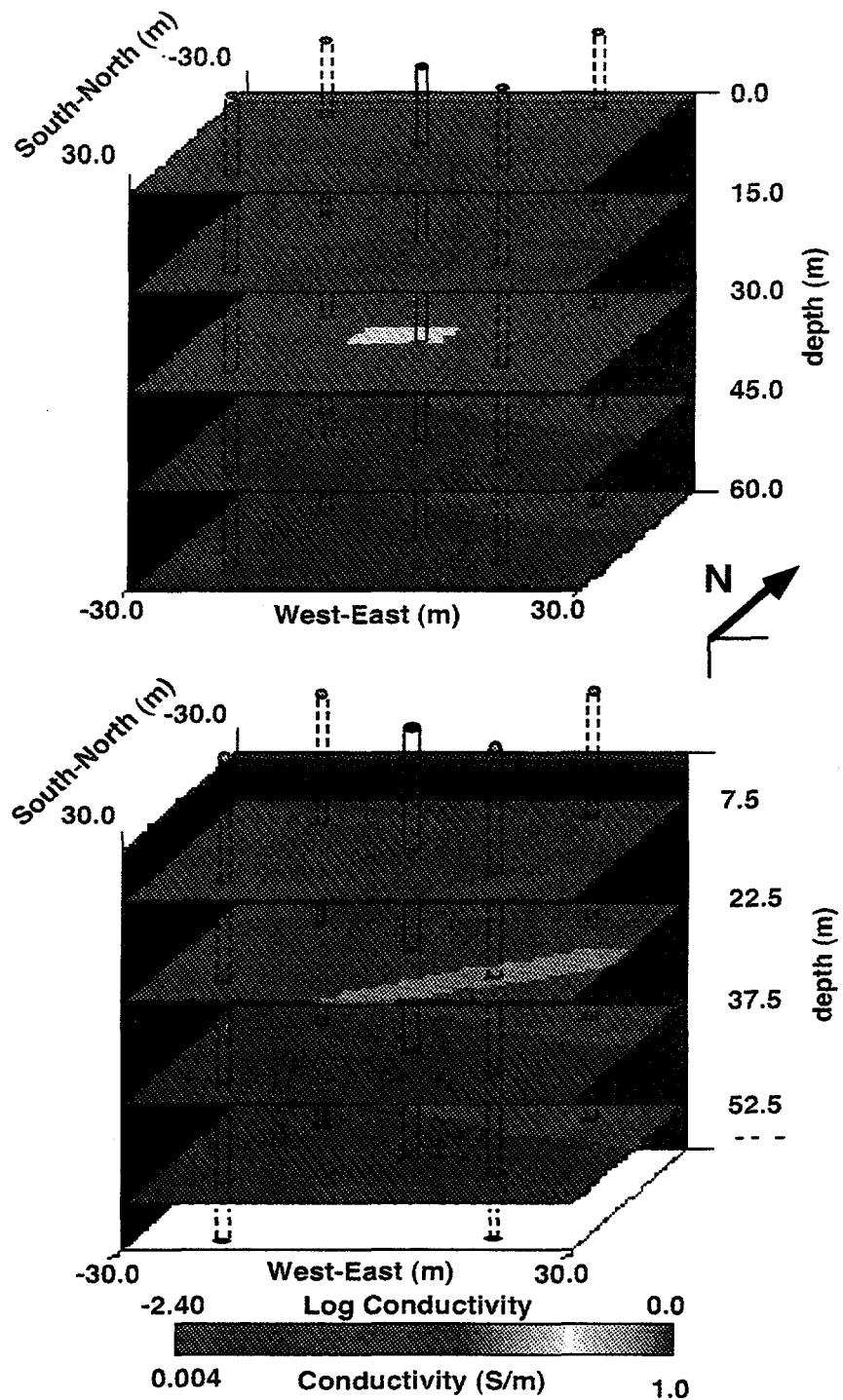
Figure 29. The five well Richmond test model used in the design experiment. Four observation wells and the injector well are indicated along with depth slices of the conductivity.
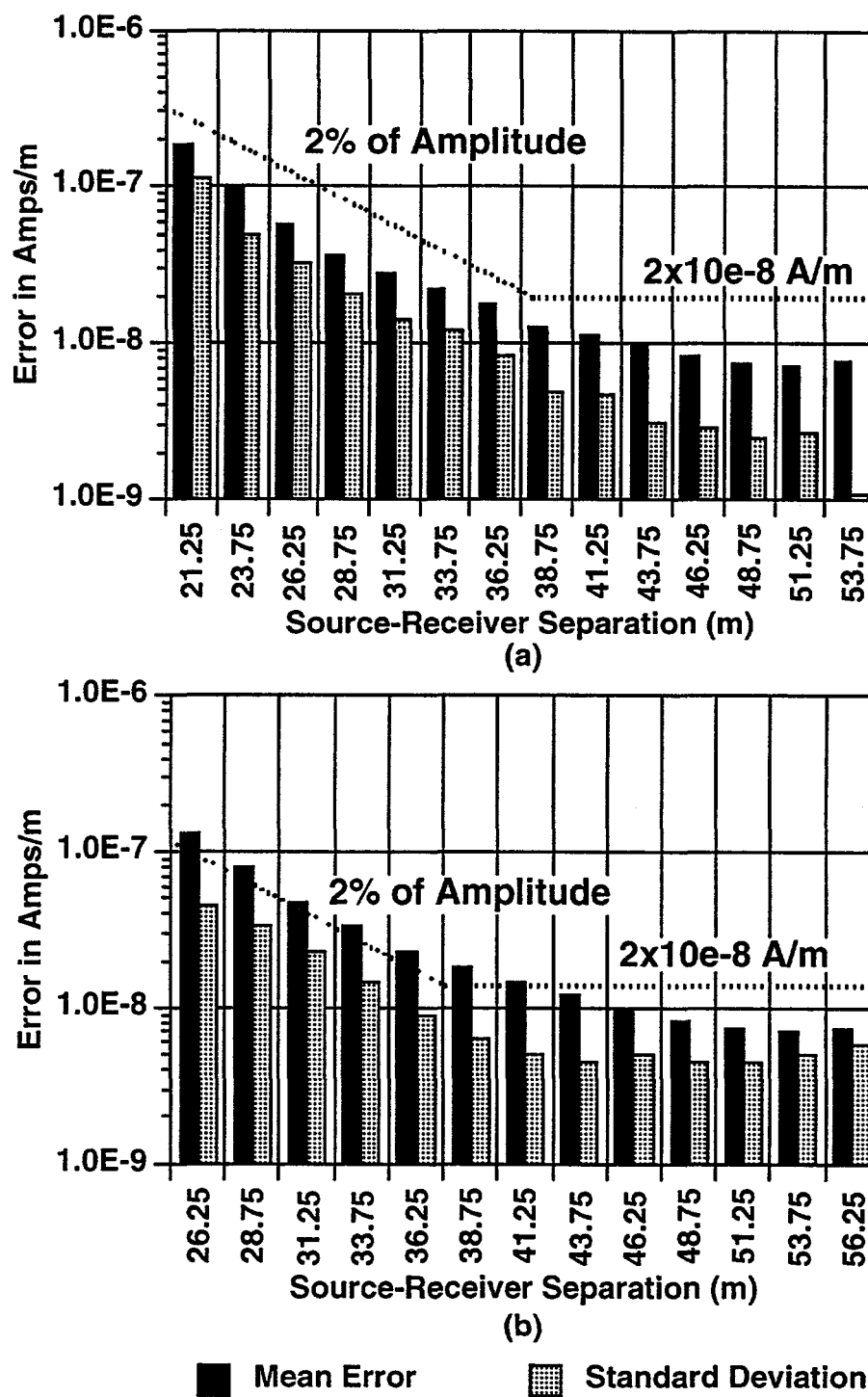
Figure 30. Mean error and standard deviations as a function of source-receiver separation for two data sets collected between a) the central and NW wells at an interval of one week, and b) the central and SW wells at a two week interval.
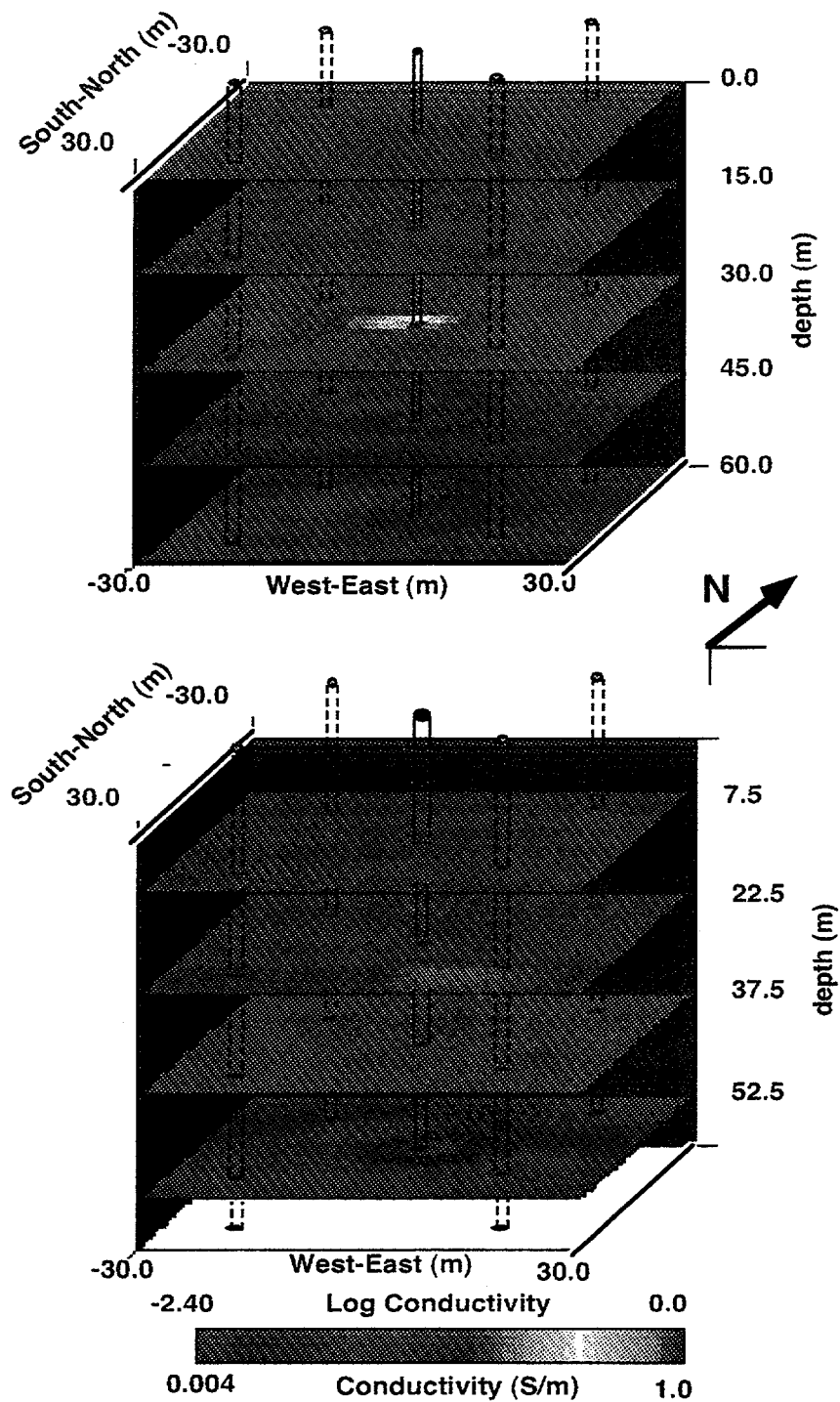
Figure 31. Conductivity reconstruction of the synthetic data generated from the model shown in Figure 29.
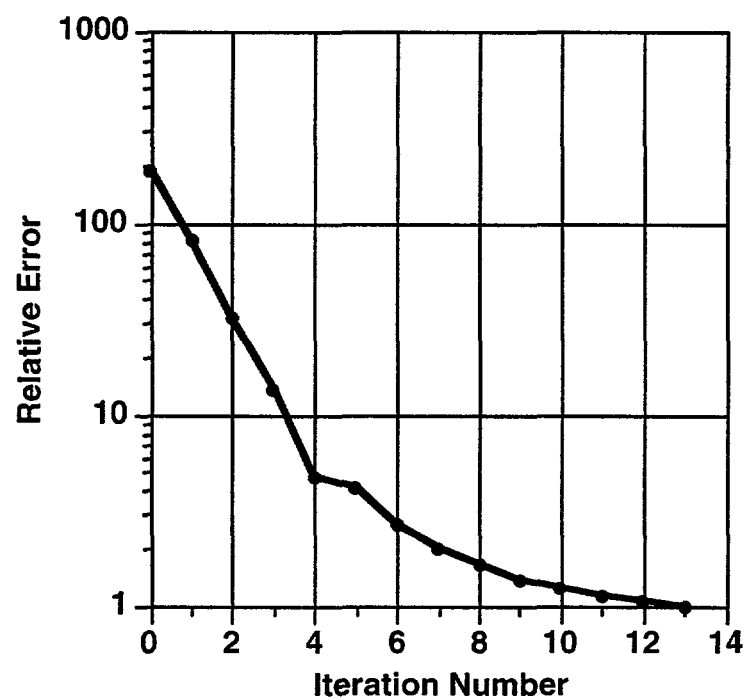
Figure 32. Normalized squared error plotted against iteration number for five well test model. Thirteen iterations were needed to obtain convergence.
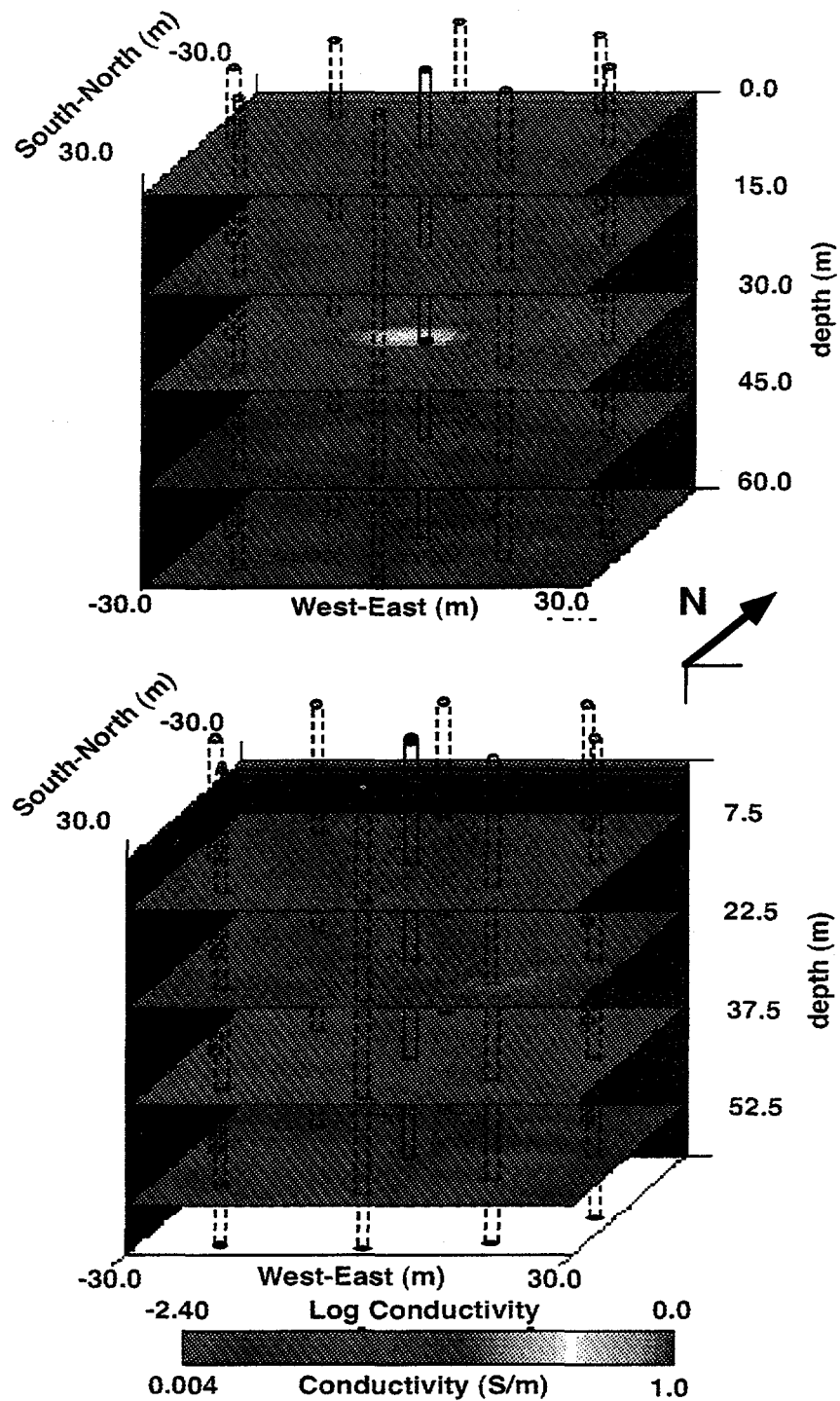
Figure 33. Conductivity reconstruction of the synthetic data generated from the model shown in Figure 29 employing eight wells.
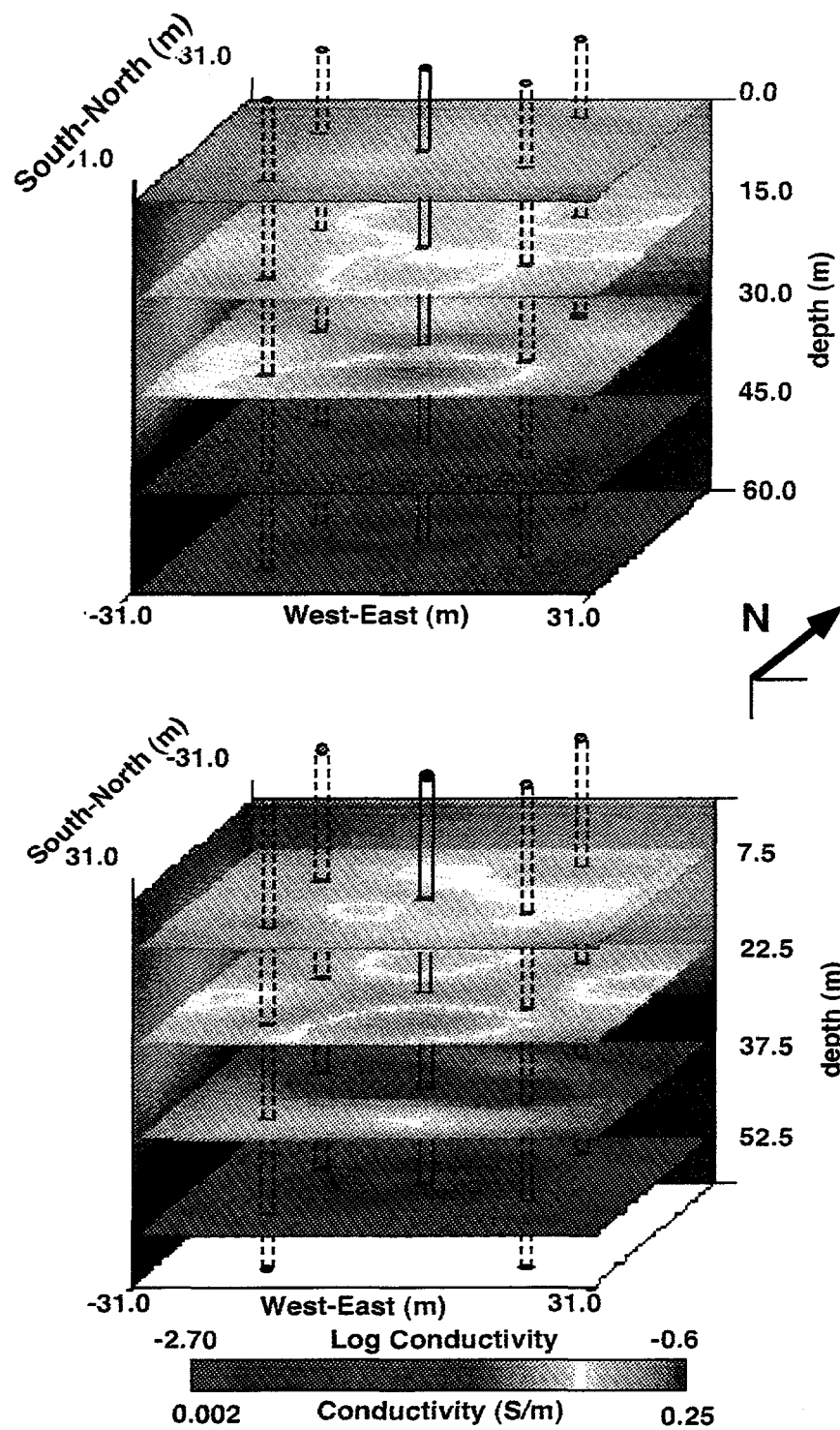
Figure 34. Conductivity reconstruction of the pre-injection data set for different depth slices.
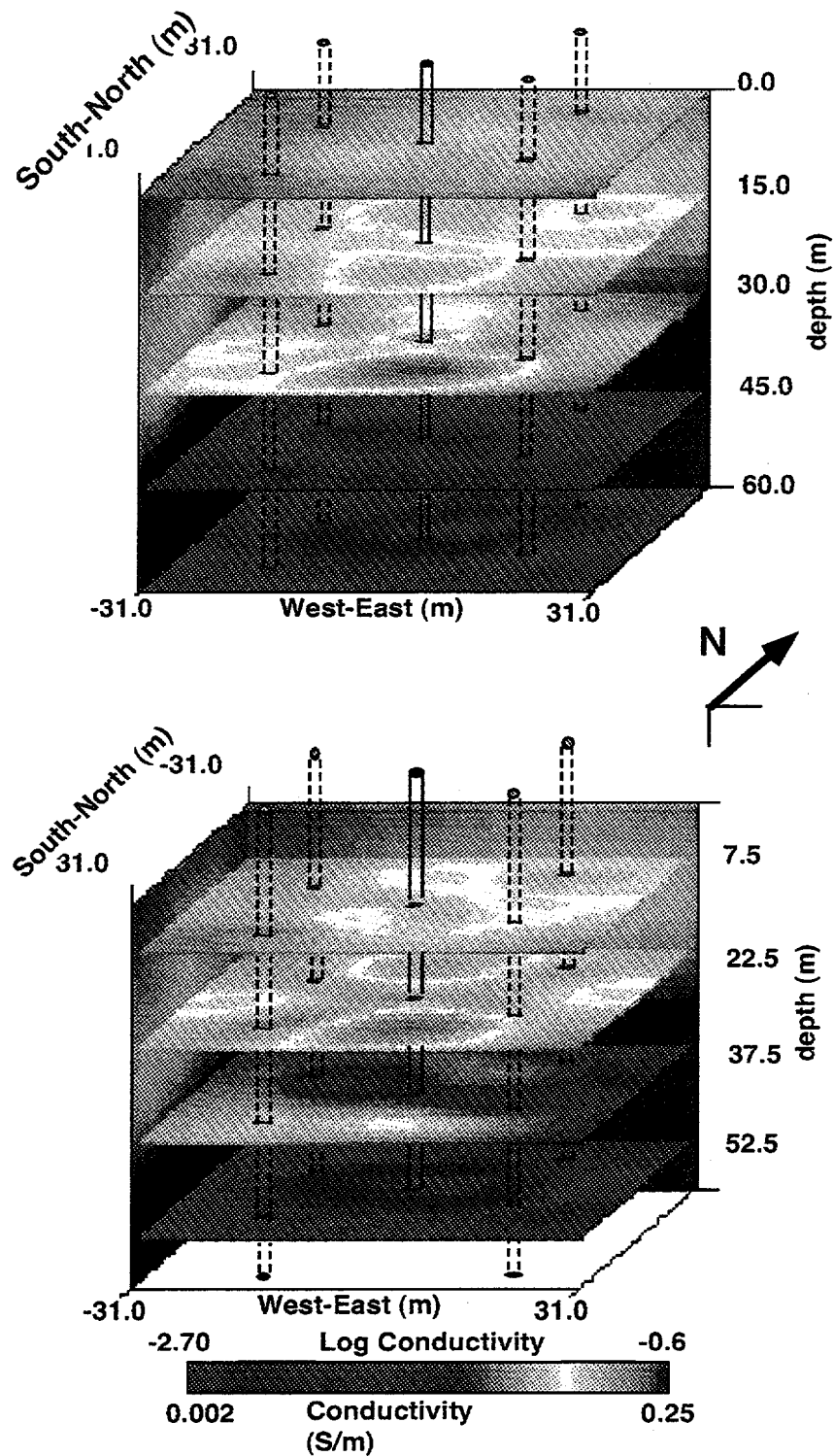
Figure 35. Conductivity reconstruction of the post-injection data set for different depth slices. The plume is indicated in the 30 m depth slice near the injector well.
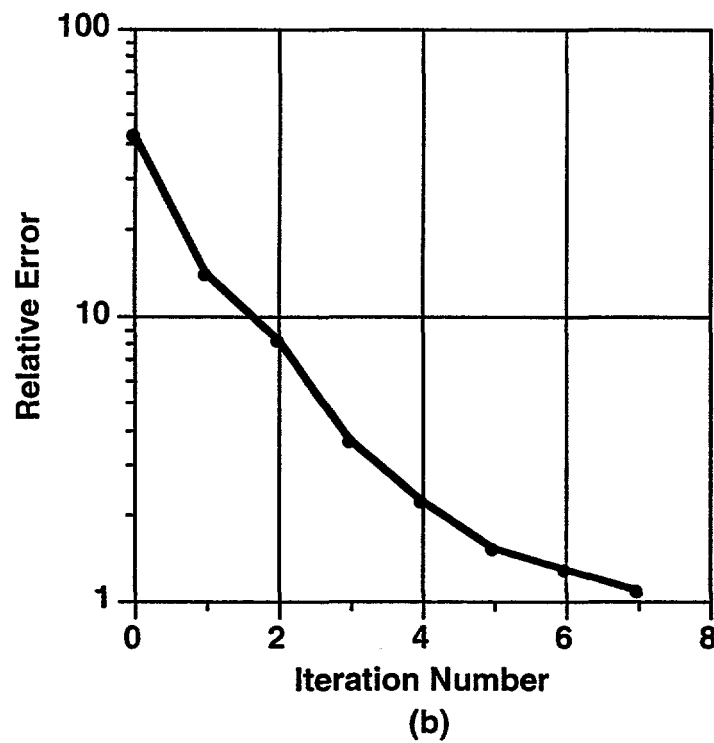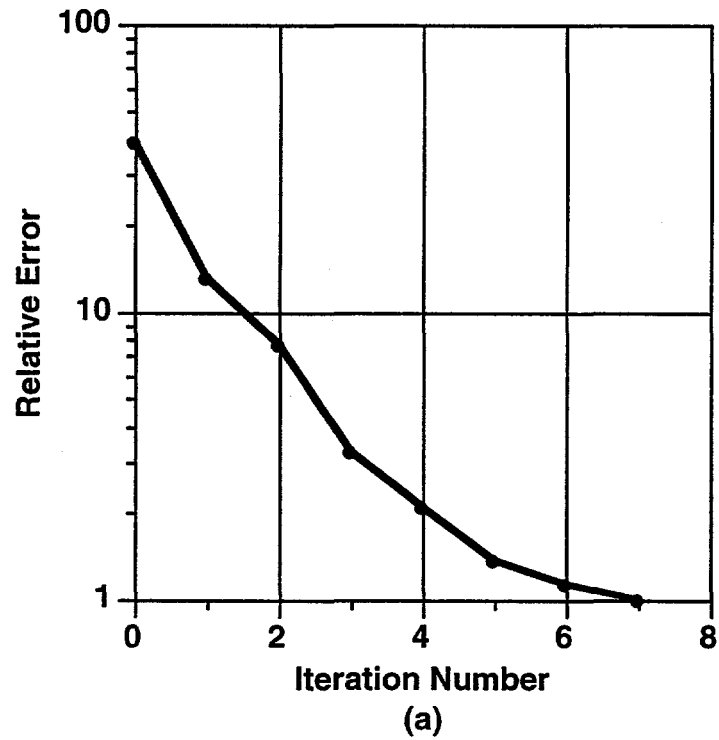
Figure 36. Normalized squared error plotted against iteration number for the (a) pre and (b) post injection data sets acquired at Richmond. The data were weighted by 2% of the maximum amplitude for each source relative to all receivers in a given observation well.
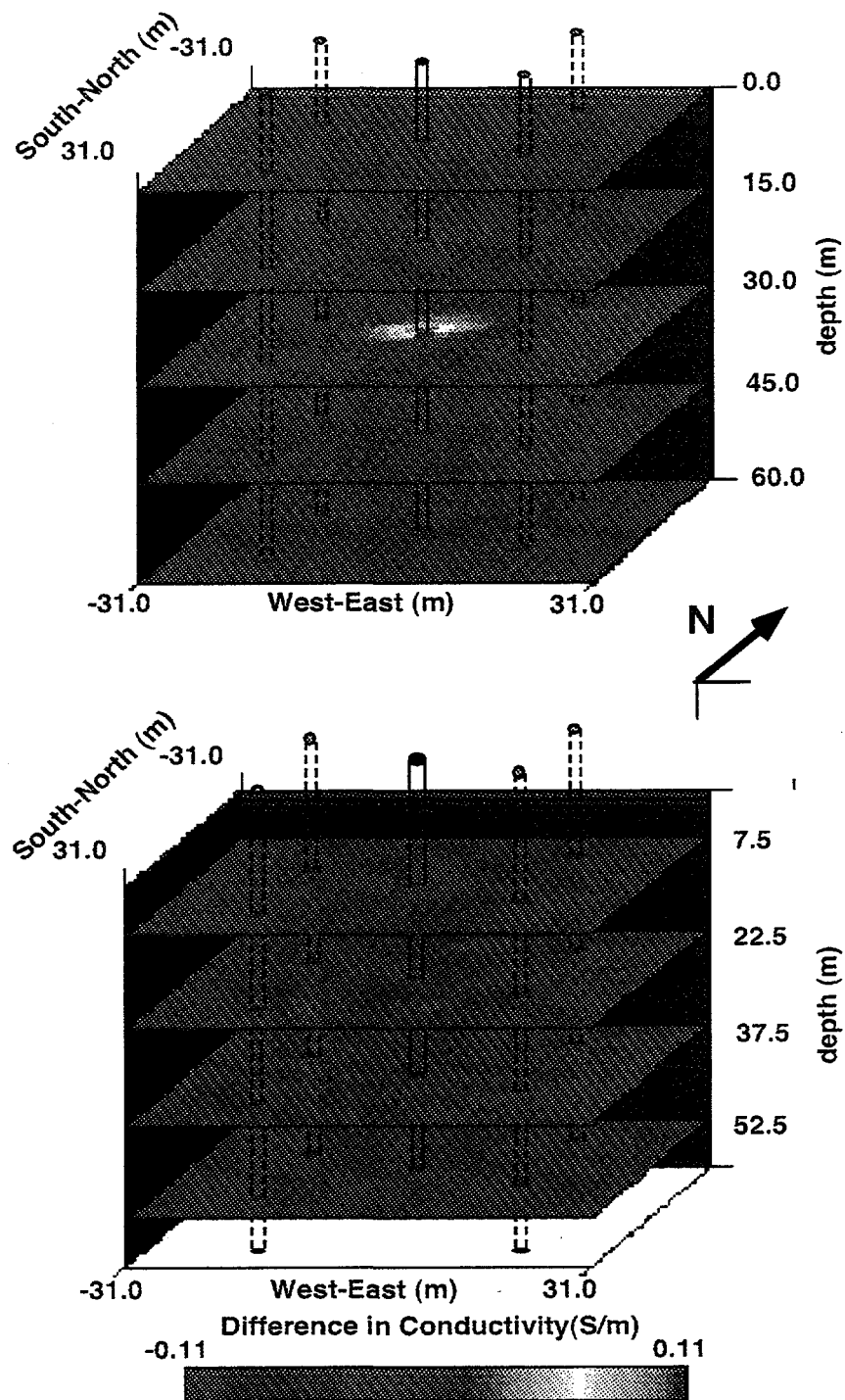
Figure 37. Difference image between the (a) pre and (b) post-injection reconstructions of the field data.
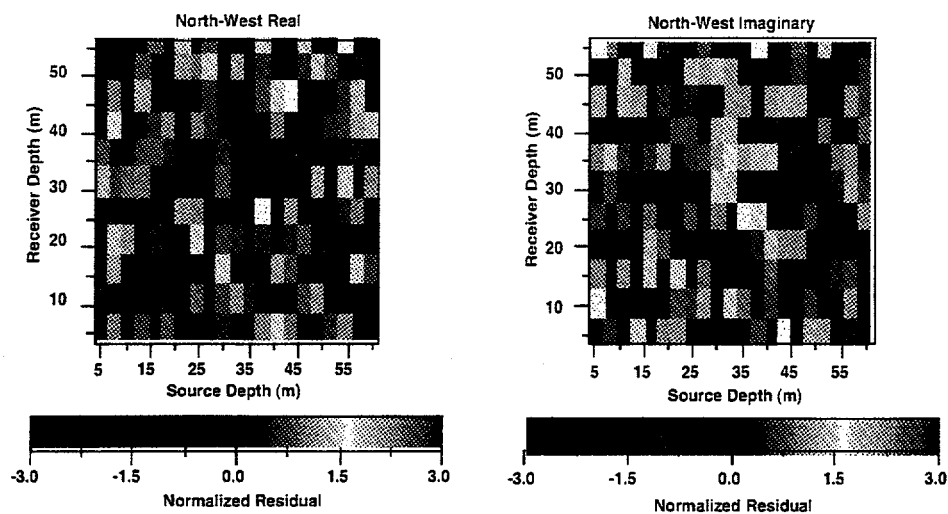
Figure 38. Residuals errors of synthetic data calculated for the image in Figure 31. Real and imaginary residuals are shown only for the northwest well.
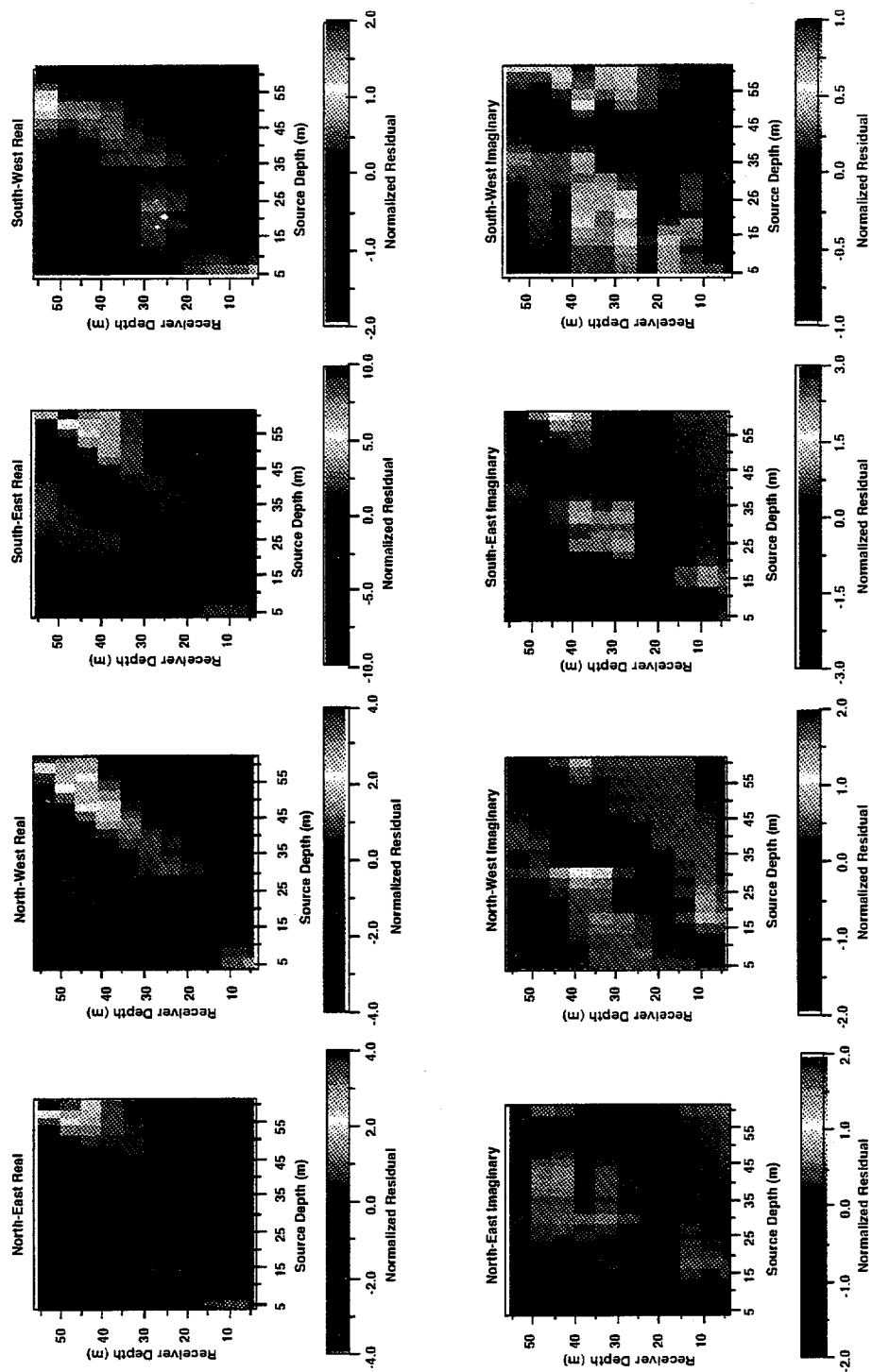
Figure 39. Residual errors for the post injection data. Results are shown for all four observation wells.
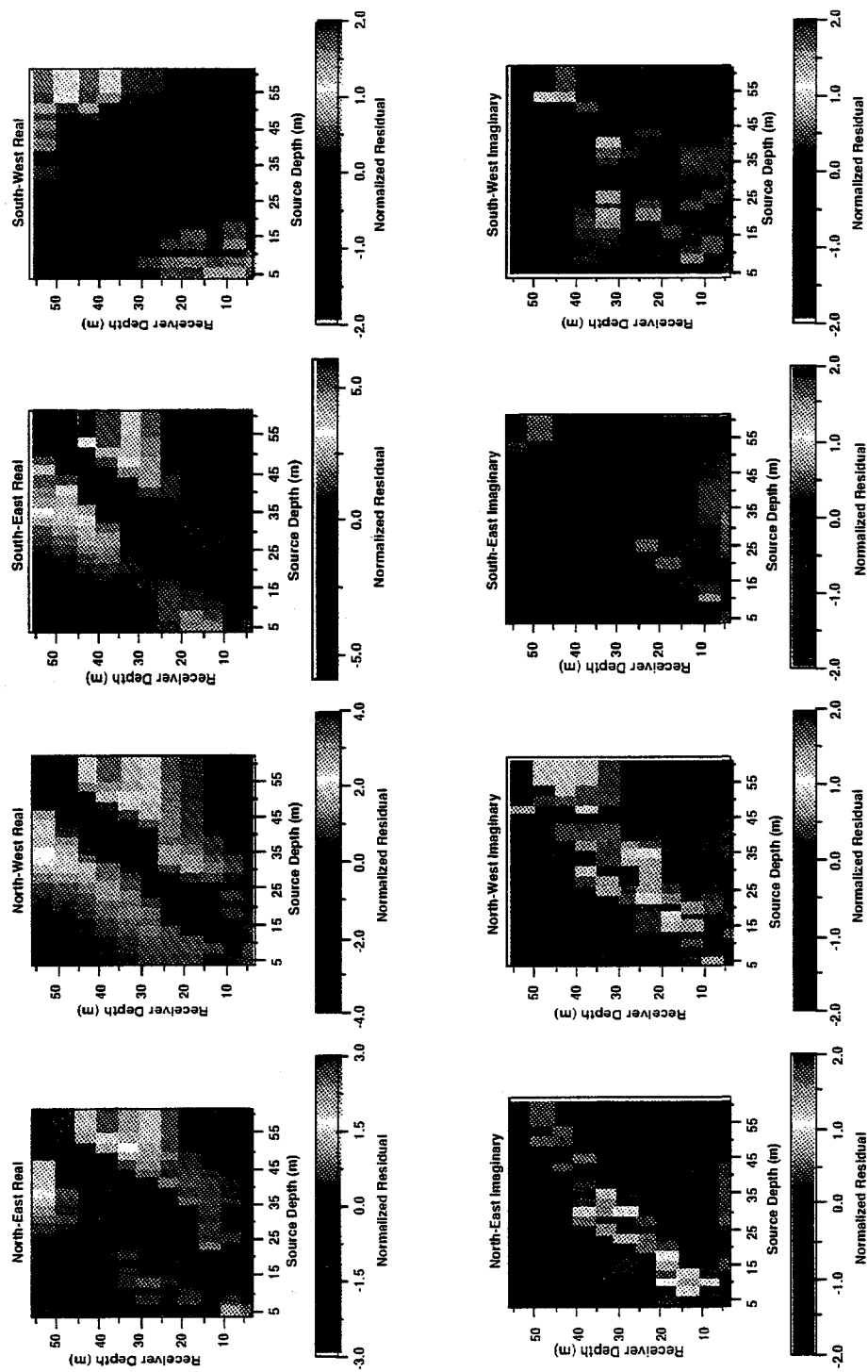
Figure 40. Residuals for a four well design experiment, where the observation wells are assumed to be deviated towards the injector.
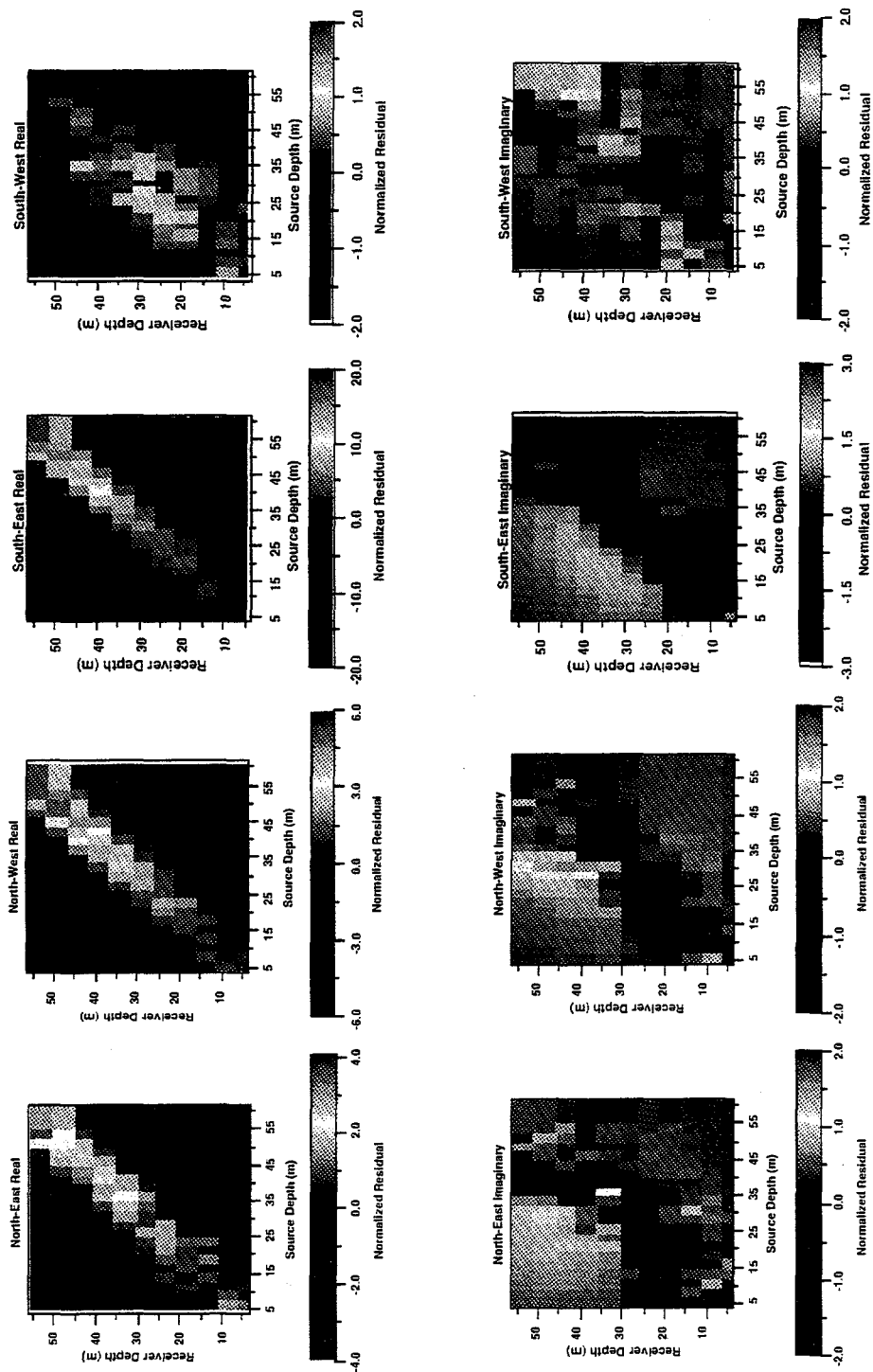
Figure 41. Residuals for a four well design experiment, where the observation wells are assumed to be deviated away from the injector.
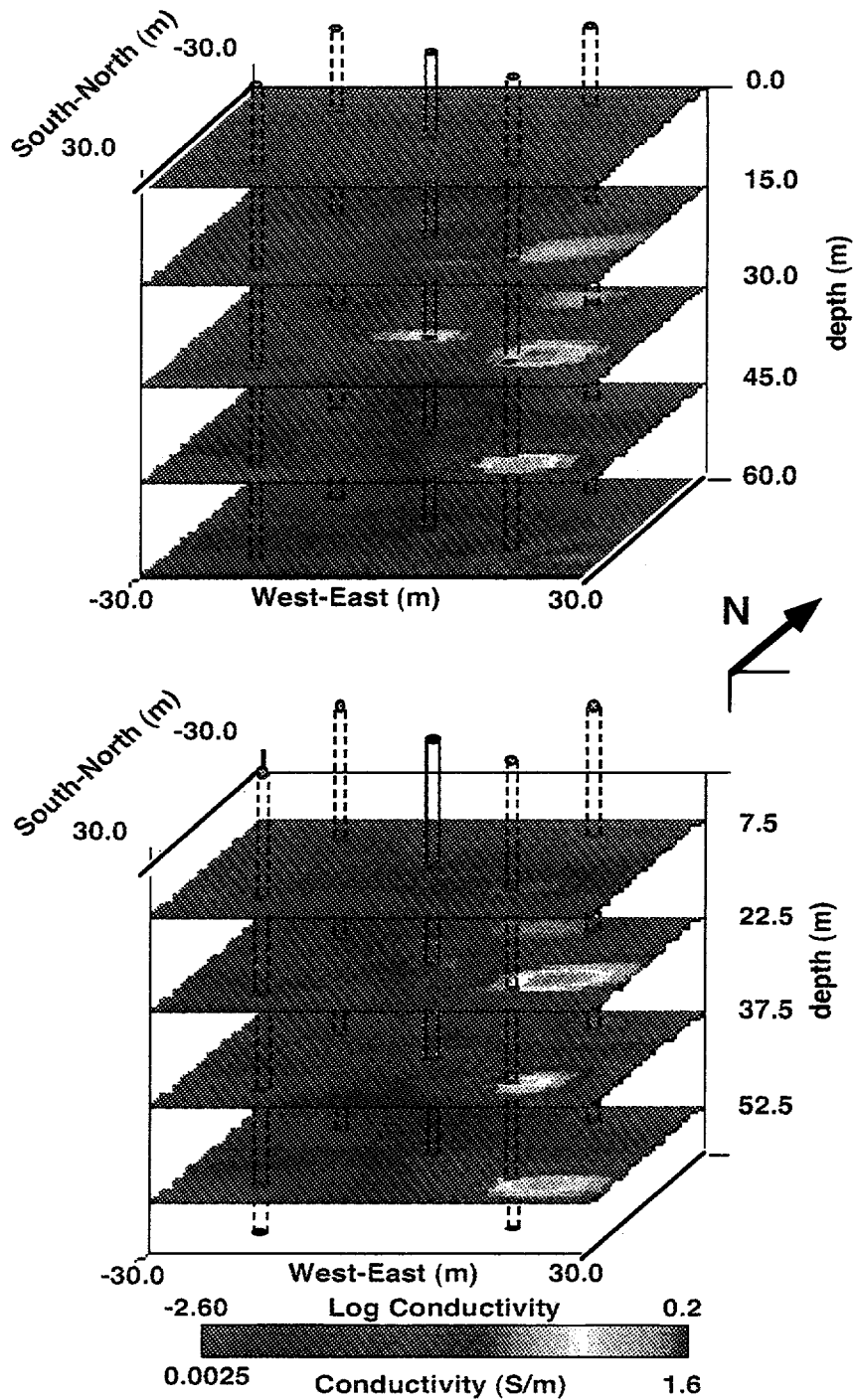
Figure 42. Conductivity reconstruction for the four well test model assuming the wells are deviated toward the injector.
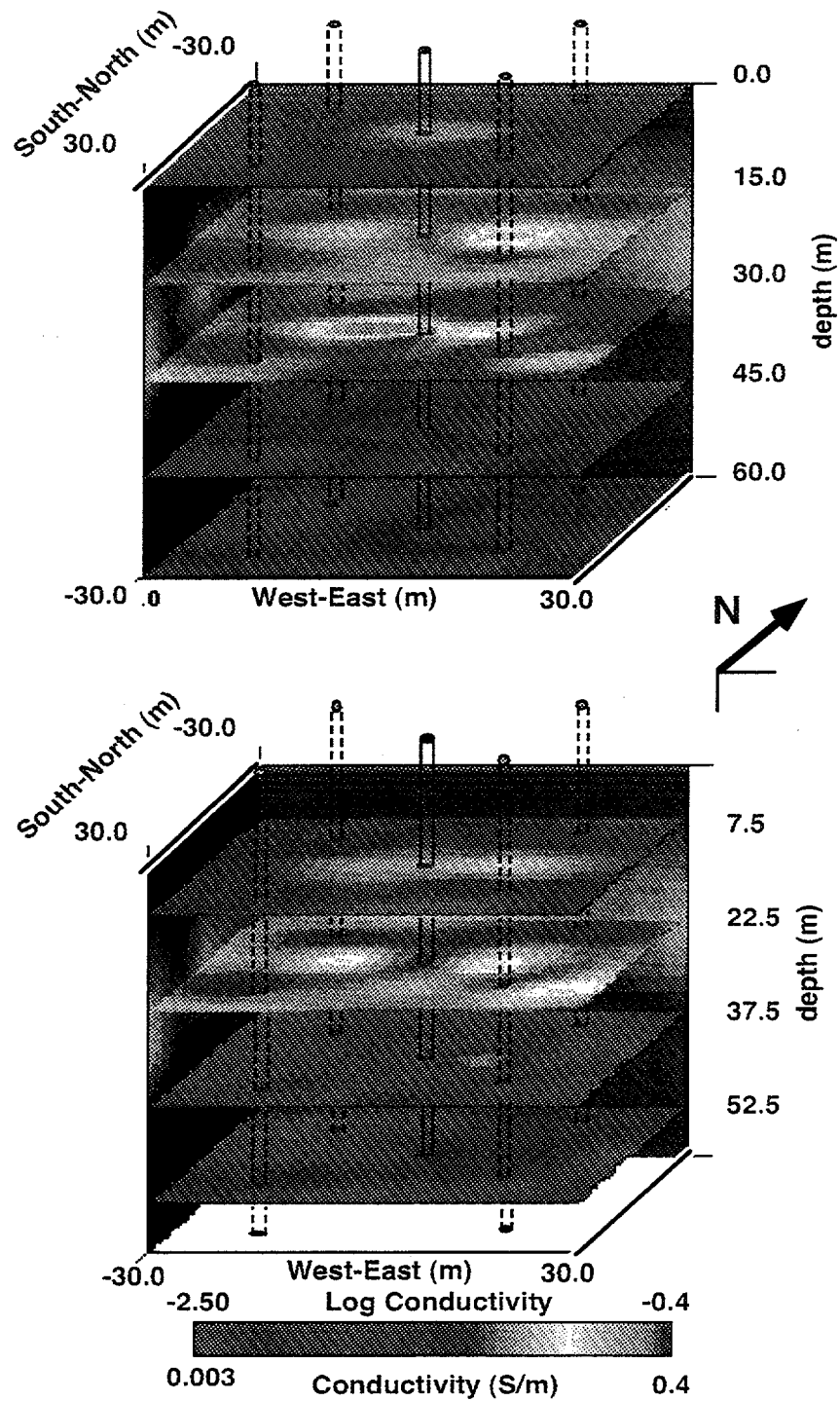
Figure 43. Conductivity reconstruction for the four well test model assuming the wells are deviated away from the injector.

CHAPTER V

CONCLUDING REMARKS

This report has demonstrated techniques that can be used to construct solutions to the 3-D electromagnetic inverse problem using full wave equation modeling. To this point great progress has been made in developing an inverse solution using the method of conjugate gradients which employs a 3-D finite difference solver to construct model sensitivities and predicted data. The forward modeling code has been developed to incorporate absorbing boundary conditions for high frequency solutions (radar), as well as complex electrical properties, including electrical conductivity, dielectric permittivity and magnetic permeability. In addition both forward and inverse codes have been ported to a massively parallel computer architecture which allows for more realistic solutions that can be achieved with serial machines.

While the inversion code has been demonstrated on field data collected at the Richmond-field site, techniques for appraising the quality of the reconstructions still need to be developed. Here it is suggested that rather than employing direct matrix inversion to construct the model covariance matrix which would be impossible because of the size of the problem, one can linearize about the 3-D model achieved in the inverse and use Monte-Carlo simulations to construct it.

Using these appraisal and construction tools, it is now necessary to demonstrate 3-D inversion for a variety of EM data sets that span the frequency range from induction sounding to radar: below 100 kHz to 100 MHz. Appraised 3-D images of the earth's electrical properties can provide researchers opportunities to infer the flow paths, flow rates and perhaps the chemistry of fluids in geologic mediums. It also offers a means to study the frequency dependence behavior of the properties in situ. This is of significant relevance to the Department of Energy, paramount to characterizing and monitoring of environmental waste sites and oil and gas exploration.

A key obstacle that must be overcome if 3-D inversion is to be practical is the availability of reliable parallel computing platforms. Unreliable and over-used platforms, such as the Intel Paragon, while good for designing and testing research software, are not feasible for production work. For example, typical waiting times in the queue for jobs to execute on the Paragon can be a week or more with a high probability of a system crash at any time.

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 Morrison H. F., 1995a, Monitoring subsurface changes over time with cross-well electromagnetic tomography: Geop. Prosp., 43, 873-902.

Alumbaugh, D.L. and Morrison, H.F., 1995b, Theoretical and practical considerations for crosswell electromagnetic tomography using a cylindrical geometry: Geophysics, 60, 846-870.

Alumbaugh, D. L. and Newman G.A., 1995, Simulation of airborne electromagnetic measurements in three dimensional environments; proceedings of SAGEEP '95 (Symposium on the Application of Geophysics to Environmental and Engineering Problems), Orlando, Florida, 319-328.

Barrett, R., Berry, M., Chan, T.F., Demmel, J., Donato, J., Dongarra, J., Eijkhout, V., Pozo, R., Romine, C., and van der Vorst, H., 1994, Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods; Society for Industrial and Applied Mathematics.

Berenger, J., 1993, A perfectly matched layer for the absorption of electromagnetic waves: J. Comp. Phy., 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 3D 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.

Dubois, D., Greenbaum, A., and Lewis, J., 1979, Approximating the inverse of a matrix for use in iterative algorithms on vector processors; Computing, 22, 257-268.

Farquharson, C. G. and Oldenburg, D. W., 1995, Approximate sensitivities for the multi-dimensional electromagnetic inverse problem: Expanded abstracts of the Int. Symp. on Three-Dimensional Electromagnetics, Oct. 4-6, Schlumberger-Doll Research, Ridgefield, Connecticut, USA.

Freund, R., 1990, On conjugate gradient type methods and polynomial preconditioners for a class of complex non-Hermitian matrices; Numer. Math., 57, 285-312.

Freund, R., 1992, Conjugate gradient type methods for linear systems with complex symmetric coefficient matrices: SIAM J. Sci. Statist. Comput., 13, 425-448.

Habashy, T.M., Groom, R. W., and Spies, B.R., 1993, Beyond the Born and Rytov approximations: a nonlinear approach to electromagnetic scattering: J. Geophys. Res., 98, 1759-1775.

Habashy, T. M., Torres-Verdin C. and Oristaglio M. L., 1995, An overview of recent advances in inversion of large-scale electromagnetic data: Expanded abstracts of the Int. Symp. on Three-Dimensional Electromagnetics, Oct. 4-6, Schlumberger-Doll Research, Ridgefield, Connecticut, USA.

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.

Lanczos, C., 1950, An iterative method for the solution of the eigenvalue problem of linear differential and integral operators; J. Res. Nat. Bur. Stand., **45**, 255-282.

Liu, C., Shen, L.C., and Zhang, Y.C., Monitoring saltwater injection using conductivity images obtained by electromagnetic cross-hole measurements, Radio Science, **30**, 1405-1415.

McGillivray, P. R. and Oldenburg, D. W., 1990, Methods for calculating Fréchet 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., Hohmann, G. W. and Anderson, W. L., 1986, Transient electromagnetic response of a three-dimensional body in a layered earth: Geophysics, **51**, 1608-1627.

Newman, G. A., 1995, Crosswell electromagnetic inversion using integral and differential equations: Geophysics, **60**, 899-911.

Newman, G. A., and Alumbaugh, D. L., 1995, Frequency-domain modeling of airborne electromagnetic responses using staggered finite differences:Geophysical Prospecting, **43**, 873-902.

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; proceedings of SAGEEP '95 (Symposium on the Application of Geophysics to Environmental and Engineering Problems), Orlando, Florida, 725-732.

Saad, Y., 1985, Practical use of polynomial preconditionings for the conjugate gradient method; SIAM J. Sci. Stat. Comput., **6**, 865-881.

Sarkar, T.K., 1987, On the application of the generalized biconjugate gradient method; J. Electro. Waves and App., **1**, 223-242.

Shadid, J.N., and Tuminaro, R.S., 1994, A comparison of preconditioned nonsymetric Krylov methods on a large-scale MIMD machine, SIAM J. Sci. Comput., **15**, 440-459.

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.

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-Verdín, C. and Habashy, T.M., 1994, Rapid 2.5-D forward modeling and inversion via a new nonlinear scattering approximation, Radio Science, **29**, 1051-1079.

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.

Tripp, A. C. and Hohmann, G. W., 1984, Block diagonalization of the electromagnetic impedance matrix of a symmetric buried body using group theory: Inst. of Electr. and Electron. Eng., Trans. Geoscience and Remote Sensing, **GE-22**, 62-69.

Wilt, M. J, Alumbaugh, D. L., Morrison, H. F., Becker A., Lee K. H., and Deszcz-Pan M., 1995a, Crosshole electromagnetic tomography: design considerations and field results: Geophysics, **60**, 871-885.

Wilt, M., Morrison, H.F., Becker, A., Tseng, H.W., Lee, K.H., Torres-Verdin, C., and Alumbaugh, D., 1995b, Crosshole electromagnetic tomography: a new technology for oil field characterization; The Leading Edge, **14**, 173-177.

Wang T. and Hohmann, G. W., 1993, A finite difference time-domain solution for three-dimensional electromagnetic modeling: Geophysics, **58**, 797-809.

92

Xiong, Z., 1992, Electromagnetic modeling of 3-D structures by the method of system iteration using integral equations: Geophysics, **57**, 1556-1561.

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.

Zhang, J. Mackie, R. L. and Madden, T. R., 1995, Three-dimensional resistivity forward modeling and inversion using conjugate gradients: Geophysics, **60**, 1313-1325.

Zhdanov, M. S. and Fang, S., 1995a, 3D electromagnetic inversion based on the quasi-linear approximation: Expanded abstracts of the Int. Symp. on Three-Dimensional Electromagnetics, Oct. 4-6, Schlumberger-Doll Research, Ridgefield, Connecticut, USA.

Zhdanov, M.S., and Fang, S., 1995b, Quasi linear approximation in 3-D electromagnetic modeling; Expanded abstracts of the Int. Symp. on Three-Dimensional Electromagnetics, Oct. 4-6, Schlumberger-Doll Research, Ridgefield, Connecticut, USA.

APPENDIX A

The perfectly matched layer concept proposed in 2-D by Berenger (1993) and 3-D by Katz et al. (1994) was originally devloped for time domain simulation of Maxwell's equations. Here we demonstrate that this method is valid for the 3-D frequency domain Helmholtz equation for the scattered electric fields using the method of Chew and Weedon (1994). First we simplify equation (7) by assuming that we are at a boundary far away from any zones of anomalous electrical properties such that it can be written

$$\nabla_h \times \nabla_e \times \mathbf{E}^s + i\omega\mu_p\left(\sigma_p + i\omega\varepsilon_p\right)\mathbf{E}^s = 0. \tag{A-1}$$

Because we are far away from any anomalous zones, a possible plane wave solution to equation (A-1) along this particular boundary is given by

$$\mathbf{E}^s = \mathbf{E}_0^s e^{i\mathbf{k}\cdot\mathbf{r}} \tag{A-2}$$

where $\mathbf{k} = k_x\hat{\mathbf{i}} + k_y\hat{\mathbf{j}} + k_z\hat{\mathbf{k}}$ and $\mathbf{r} = x\hat{\mathbf{i}} + y\hat{\mathbf{j}} + z\hat{\mathbf{k}}$. Because $\nabla \times \mathbf{E}_0^s e^{i\mathbf{k}\cdot\mathbf{r}} = i\mathbf{k} \times \mathbf{E}_0^s e^{i\mathbf{k}\cdot\mathbf{r}}$, it is easy to show that when equation (A-2) is substituted into (A-1), the resulting expression has the form

$$-\mathbf{k}_h \times \mathbf{k}_e \times \mathbf{E}^s + i\omega\mu_p\left(\sigma_p + i\omega\varepsilon_p\right)\mathbf{E}^s = 0 \tag{A-3}$$

where

$$\mathbf{k}_e = \frac{k_x}{e_x}\hat{\mathbf{i}} + \frac{k_y}{e_y}\hat{\mathbf{j}} + \frac{k_z}{e_z}\hat{\mathbf{k}} \tag{A-4}$$

and

$$\mathbf{k}_h = \frac{k_x}{h_x}\hat{\mathbf{i}} + \frac{k_y}{h_y}\hat{\mathbf{j}} + \frac{k_z}{h_z}\hat{\mathbf{k}}. \tag{A-5}$$

Using a vector identity, the left hand term in equation (A-3) can be expanded to yield

$$\left(\mathbf{k}_h \cdot \mathbf{k}_e\right)\mathbf{E}^s - \mathbf{k}_e\left(\mathbf{k}_h \cdot \mathbf{E}^s\right) + i\omega\mu_p\left(\sigma_p + i\omega\varepsilon_p\right)\mathbf{E}^s = 0. \tag{A-6}$$

Because we are in a homogenous region absent of any free 'secondary' charge,

$$\nabla_h \cdot \mathbf{E}^s = \mathbf{k}_h \cdot \mathbf{E}^s = 0 \tag{A-7}$$

94

and thus we are left with

$$(k_h \cdot k_e)E^s + i\omega\mu_p(\sigma_p + i\omega\varepsilon_p)E^s = 0 \qquad \text{(A-8)}$$

or

$$k_h \cdot k_e = \frac{1}{e_x h_x}k_x^2 + \frac{1}{e_y h_y}k_y^2 + \frac{1}{e_z h_z}k_z^2 = \kappa^2 \qquad \text{(A-9)}$$

where $\kappa^2 = -i\omega\mu_p(\sigma_p + i\omega\varepsilon_p)$.

Let us now assume that the plane wave is obliquely incident on an interface at z=c where c is constant. Chew and Weedon (1994) show that the solution to equation (A-9) is that of a 3-D ellipsoid which is satisfied by

$$k_x = \kappa\sqrt{e_x h_x}\sin\theta\cos\phi \qquad \text{(A-10)}$$

$$k_y = \kappa\sqrt{e_y h_y}\sin\theta\sin\phi \qquad \text{(A-11)}$$

and

$$k_z = \kappa\sqrt{e_z h_z}\cos\theta. \qquad \text{(A-12)}$$

In addition they find the reflection coefficients for the TE and TM modes at the boundary to be

$$R^{TE} = \frac{k_{1z}e_{2z}\mu_2 - k_{2z}e_{1z}\mu_1}{k_{1z}e_{2z}\mu_2 + k_{2z}e_{1z}\mu_1} \qquad \text{(A-13)}$$

and

$$R^{TM} = \frac{k_{1z}h_{2z}\hat{y}_2 - k_{2z}h_{1z}\hat{y}_1}{k_{1z}h_{2z}\hat{y}_2 + k_{2z}h_{1z}\hat{y}_1} \qquad \text{(A-14)}$$

where the 1's represent the properties of medium the incident wave is traveling through, the 2's designates the medium it will be transmitted to, and $\hat{y}_i = \sigma_i + i\omega\varepsilon_i$.

Phase matching will occur if $k_{1x} = k_{2x}$ and $k_{1y} = k_{2y}$. To accomplish this we first set the material properties of the two media to be identical ($\kappa_1 = \kappa_2$) and then choose $h_x = e_x$ and $h_y = e_y$. If we now let $e_{1x} = e_{2x} = e_{1y} = e_{2y} = 1$, and furthermore set $\theta_1 = \theta_2$ and $\phi_1 = \phi_2$, then the two reflection coefficients in (A-13) and (A-14) are zero and no reflections are generated at the interface. However by making e_{2z} complex, we provide additional loss in k_{2z} which causes the wave to more rapidly attenuate in medium 2 than it would otherwise.

It must be mentioned that three assumptions have been made in this analysis which can not be incoporated into the 3-D FD modeling scheme. The first assumption is that e_x, e_y, h_x and h_y do not vary along the 'z' interface. In the corners of the mesh these values are also varying to incoporate absorption along the x and y interfaces, and thus perfect matching can not occur in these locations and reflections will be generated. However we have not experienced any serious problems with regards to this phenomenon. The second assumption that we have made here is that the interface is located far away from any regions of anomalous electrical properties. Nevertheless as the results in Figure 10d and 10e indicate, the PML is valid even when the stretching occurs within these regions, for instance at the mesh boundary located at the bottom of the model shown in Figure 10a. Finally, it is assumed that $h_j = e_j$ for j=x,y,and z. As shown in Appendix C, h_j is actually a weighted average of the e_j values assigned to two adjacent cells where the weighting depends on the cell dimensions. However by using both a constant value of e_j and a constant cell size throughout the PML region, any problems with this assumption can be avoided.

APPENDIX B

To develop an expression for the modified Helmholtz equation for the scattered electric fields we start with the modified Maxwell's equations given in expressions (1) and (2) where the modified derivative operators are given by expressions (3) and (4). We now define the primary, or background fields that exist in the stretched system to be

$$\nabla_e \times \mathbf{E^P} = -i\omega\mu_p\mathbf{H^P} - i\omega\mathbf{M^P} \qquad \text{(B-1)}$$

and

$$\nabla_h \times \mathbf{H^P} = \left(\sigma_p + i\omega\varepsilon_p\right)\mathbf{E^P} + \mathbf{J^P}. \qquad \text{(B-2)}$$

Subtracting (B-1) from (1), (B-2) from (2), and adding and subtracting an arbitrary source term to each equation yields

$$\nabla_e \times \mathbf{E^S} = -i\omega\mu\mathbf{H^t} + i\omega\mu_p\mathbf{H^P} + \left(i\omega\mu\mathbf{H^P} - i\omega\mu\mathbf{H^P}\right) \qquad \text{(B-3)}$$

and

$$\nabla_h \times \mathbf{H^S} = (\sigma + i\omega\varepsilon)\mathbf{E^t} - \left(\sigma_p + i\omega\varepsilon_p\right)\mathbf{E^P} + \left[(\sigma + i\omega\varepsilon)\mathbf{E^P} - (\sigma + i\omega\varepsilon)\mathbf{E^P}\right]. \qquad \text{(B-4)}$$

Properly subtracting the source terms yields the modified versions of Maxwell's equations for the scattered fields:

$$\nabla_e \times \mathbf{E^S} = -i\omega\mu\mathbf{H^S} - i\omega\left(\mu - \mu_p\right)\mathbf{H^P} \qquad \text{(B-5)}$$

and

$$\nabla_h \times \mathbf{H^S} = (\sigma + i\omega\varepsilon)\mathbf{E^S} + \left(\sigma - \sigma_p + i\omega(\varepsilon - \varepsilon_p)\right)\mathbf{E^P}. \qquad \text{(B-6)}$$

To derive the modified Helmholtz equation from these two equations, for numerical stability we first multiply (B-5) by μ_p/μ to yield

$$\frac{\mu_p}{\mu}\nabla_e \times \mathbf{E^S} = -i\omega\mu\mathbf{H^S} - i\omega\mu_p\frac{\left(\mu - \mu_p\right)}{\mu}\mathbf{H^P}. \qquad \text{(B-7)}$$

Taking the curl of (B-7) results in the following expression;

97

$$\nabla_h \times \frac{\mu_p}{\mu} \nabla_e \times \mathbf{E^s} = -i\omega\mu\nabla_h \times \mathbf{H^s} - i\omega\mu_p\nabla_h \times \left[\frac{(\mu - \mu_p)}{\mu}\mathbf{H^p} \right]. \qquad \text{(B-8)}$$

Finally, the right hand side of equation (B-6) is substituted for the $\nabla_h \times \mathbf{H^s}$ term in the above equation to arrive at the modified Helmholtz equation given in expression (7).

APPENDIX C

To develop the finite difference equations we start with the modified Helmholtz equation as given if equation (7). First let us expand out the two first order curl operators, i.e.,

$$\nabla_e \times \mathbf{E}^s = \left(\frac{1}{e_y} \frac{\partial Ez^s}{\partial y} - \frac{1}{e_z} \frac{\partial Ey^s}{\partial z} \right) \hat{\mathbf{i}} + \left(\frac{1}{e_z} \frac{\partial Ex^s}{\partial z} - \frac{1}{e_x} \frac{\partial Ez^s}{\partial x} \right) \hat{\mathbf{j}} + \tag{C-1}$$

$$\left(\frac{1}{e_x} \frac{\partial Ey^s}{\partial x} - \frac{1}{e_y} \frac{\partial Ex^s}{\partial y} \right) \hat{\mathbf{k}}$$

and

$$\nabla_h \times \left[\frac{(\mu - \mu_p)}{\mu} \mathbf{H}^p \right] = \left(\frac{1}{h_y} \frac{\partial}{\partial y} \frac{(\mu_z - \mu_p)Hz^p}{\mu_z} - \frac{1}{h_z} \frac{\partial}{\partial z} \frac{(\mu_y - \mu_p)Hy^p}{\mu_y} \right) \hat{\mathbf{i}} +$$

$$\left(\frac{1}{h_z} \frac{\partial}{\partial z} \frac{(\mu_x - \mu_p)Hx^p}{\mu_x} - \frac{1}{h_x} \frac{\partial}{\partial x} \frac{(\mu_z - \mu_p)Hz^p}{\mu_z} \right) \hat{\mathbf{j}} + \tag{C-2}$$

$$\left(\frac{1}{h_x} \frac{\partial}{\partial x} \frac{(\mu_y - \mu_p)Hy^p}{\mu_y} - \frac{1}{h_y} \frac{\partial}{\partial y} \frac{(\mu_x - \mu_p)Hx^p}{\mu_x} \right) \hat{\mathbf{k}}.$$

In this expression, μ_w for w=x,y,and z represents the magnetic permeability that is averaged across the face of two cells in the w'th direction. Next, expanding the second order curl on the left side of equation (7) we find that

$$\nabla_h \times \left[\frac{\mu_p}{\mu} \nabla_e \times \mathbf{E}^s \right] = \left(\frac{1}{h_y} \frac{\partial}{\partial y} \left(\frac{\mu_p}{\mu_z e_x} \frac{\partial Ey^s}{\partial x} \right) - \frac{1}{h_y} \frac{\partial}{\partial y} \left(\frac{\mu_p}{\mu_z e_y} \frac{\partial Ex^s}{\partial y} \right) - \right.$$

$$\left. \frac{1}{h_z} \frac{\partial}{\partial z} \left(\frac{\mu_p}{\mu_y e_z} \frac{\partial Ex^s}{\partial z} \right) + \frac{1}{h_z} \frac{\partial}{\partial z} \left(\frac{\mu_p}{\mu_y e_x} \frac{\partial Ez^s}{\partial x} \right) \right] \hat{\mathbf{i}} +$$

$$\left[\frac{1}{h_z} \frac{\partial}{\partial z} \left(\frac{\mu_p}{\mu_x e_y} \frac{\partial Ez^s}{\partial y} \right) + \frac{1}{h_z} \frac{\partial}{\partial z} \left(\frac{\mu_p}{\mu_x e_z} \frac{\partial Ey^s}{\partial z} \right) - \right. \tag{C-3}$$

$$\left. \frac{1}{h_x} \frac{\partial}{\partial x} \left(\frac{\mu_p}{\mu_z e_x} \frac{\partial Ey^s}{\partial x} \right) + \frac{1}{h_x} \frac{\partial}{\partial x} \left(\frac{\mu_p}{\mu_z e_y} \frac{\partial Ex^s}{\partial y} \right) \right] \hat{\mathbf{j}} +$$

$$\left[\frac{1}{h_x} \frac{\partial}{\partial x} \left(\frac{\mu_p}{\mu_y e_z} \frac{\partial Ex^s}{\partial z} \right) + \frac{1}{h_x} \frac{\partial}{\partial x} \left(\frac{\mu_p}{\mu_y e_x} \frac{\partial Ez^s}{\partial x} \right) - \right.$$

$$\left. \frac{1}{h_y} \frac{\partial}{\partial y} \left(\frac{\mu_p}{\mu_x e_y} \frac{\partial Ez^s}{\partial y} \right) + \frac{1}{h_y} \frac{\partial}{\partial y} \left(\frac{\mu_p}{\mu_x e_z} \frac{\partial Ey^s}{\partial z} \right) \right] \hat{\mathbf{k}}.$$

Now let us examine the parts of the Helmholtz equation corresponding to $\hat{\mathbf{i}}$, $\hat{\mathbf{j}}$, and $\hat{\mathbf{k}}$ separately since these are the three equations that we are going to be solving at each node. For the $\hat{\mathbf{i}}$ component we have

$$\frac{1}{h_y}\frac{\partial}{\partial y}\left(\frac{\mu_p}{\mu_z e_x}\frac{\partial Ey^s}{\partial x}\right) - \frac{1}{h_y}\frac{\partial}{\partial y}\left(\frac{\mu_p}{\mu_z e_y}\frac{\partial Ex^s}{\partial y}\right) - \frac{\mu_p}{h_z}\frac{\partial}{\partial z}\left(\frac{1}{\mu_y e_z}\frac{\partial Ex^s}{\partial z}\right) + \qquad \text{(C-4)}$$

$$\frac{1}{h_z}\frac{\partial}{\partial z}\left(\frac{\mu_p}{\mu_y e_x}\frac{\partial Ez^s}{\partial x}\right) + i\omega\mu_p(\sigma + i\omega\varepsilon)Ex^s = -i\omega\mu_p\big(\sigma - \sigma_p + i\omega(\varepsilon - \varepsilon_p)\big)Ex^P -$$

$$\frac{i\omega\mu_p}{h_y}\frac{\partial}{\partial y}\frac{(\mu_z - \mu_p)Hz^P}{\mu_z} - \frac{i\omega\mu_p}{h_z}\frac{\partial}{\partial z}\frac{(\mu_y - \mu_p)Hy^P}{\mu_y} .$$

Approximating equation (C-4) with finite differences yields

$$\left\{\frac{\mu_p}{\mu_{z_{i+\frac{1}{2},j+\frac{1}{2},k}}}\left[\frac{1}{(e_{xi}\Delta x_i)}\left(Ey^s_{i+1,j+\frac{1}{2},k} - Ey^s_{i,j+\frac{1}{2},k}\right) - \frac{1}{(e_{yj}\Delta y_j)}\left(Ex^s_{i+\frac{1}{2},j+1,k} - Ex^s_{i+\frac{1}{2},j,k}\right)\right] - \right.$$

$$\left.\frac{\mu_p}{\mu_{z_{i+\frac{1}{2},j-\frac{1}{2},k}}}\left[\frac{1}{(e_{xi}\Delta x_i)}\left(Ey^s_{i+1,j-\frac{1}{2},k} - Ey^s_{i,j-\frac{1}{2},k}\right) - \frac{1}{(e_{yj-1}\Delta y_{j-1})}\left(Ex^s_{i+\frac{1}{2},j,k} - Ex^s_{i+\frac{1}{2},j-1,k}\right)\right]\right\}\frac{1}{(h_{yj}\Delta\overline{y}_j)} +$$

$$\left\{\frac{\mu_p}{\mu_{y_{i+\frac{1}{2},j,k+\frac{1}{2}}}}\left[\frac{1}{(e_{xi}\Delta x_i)}\left(Ez^s_{i+1,j,k+\frac{1}{2}} - Ez^s_{i,j,k+\frac{1}{2}}\right) - \frac{1}{(e_{zk}\Delta z_k)}\left(Ex^s_{i+\frac{1}{2},j,k+1} - Ex^s_{i+\frac{1}{2},j,k}\right)\right] - \right.$$

$$\left.\frac{\mu_p}{\mu_{y_{i+\frac{1}{2},j,k-\frac{1}{2}}}}\left[\frac{1}{(e_{xi}\Delta x_i)}\left(Ez^s_{i+1,j,k-\frac{1}{2}} - Ez^s_{i,j,k-\frac{1}{2}}\right) - \frac{1}{(e_{zk-1}\Delta z_{k-1})}\left(Ex^s_{i+\frac{1}{2},j,k} - Ex^s_{i+\frac{1}{2},j,k-1}\right)\right]\right\}\frac{1}{(h_{zk}\Delta\overline{z}_k)} +$$

$$i\omega\mu_p\hat{y}_p Ex^s_{i+\frac{1}{2},j,k} = -i\omega\mu_p\left(\hat{y}_{i+\frac{1}{2},j,k} - \hat{y}_p\right)Ex^P_{i+\frac{1}{2},j,k} - \qquad \text{(C-5)}$$

$$i\omega\mu_p\left\{\left[\frac{\left(\mu_{y_{i+\frac{1}{2},j,k+\frac{1}{2}}} - \mu_p\right)}{\mu_{y_{i+\frac{1}{2},j,k+\frac{1}{2}}}}Hy^P_{i+\frac{1}{2},j,k+\frac{1}{2}} - \frac{\left(\mu_{y_{i+\frac{1}{2},j,k-\frac{1}{2}}} - \mu_p\right)}{\mu_{y_{i+\frac{1}{2},j,k-\frac{1}{2}}}}Hy^P_{i+\frac{1}{2},j,k-\frac{1}{2}}\right]\frac{1}{(h_{zk}\Delta\overline{z}_k)} - \right.$$

$$\left.\left[\frac{\left(\mu_{z_{i+\frac{1}{2},j+\frac{1}{2},k}} - \mu_p\right)}{\mu_{z_{i+\frac{1}{2},j+\frac{1}{2},k}}}Hz^P_{i+\frac{1}{2},j+\frac{1}{2},k} - \frac{\left(\mu_{z_{i+\frac{1}{2},j-\frac{1}{2},k}} - \mu_p\right)}{\mu_{z_{i+\frac{1}{2},j-\frac{1}{2},k}}}Hz^P_{i+\frac{1}{2},j-\frac{1}{2},k}\right]\frac{1}{(h_{yj}\Delta\overline{y}_j)}\right\}$$

where now $\hat{y} = \sigma + i\omega\varepsilon$. In this expression Δw_l for w=x,y,z and l=i,j,k represents the width of the l'th cell in the w'th direction. Similarly $\Delta\overline{w}_l$ is the distance in the w'th direction between the centers of cells l and l-1. Notice in this expression how the finite differences and the stretching parameters conveniently group together. It is also apparent that because $\Delta\overline{w}_l$ is essentially the weighted average of the widths of cells l and l-1, h_{wl} is the weighted average of e_{wl} and e_{wl-1}.

We can similarly expand the equations for the $\hat{\mathbf{j}}$ and $\hat{\mathbf{k}}$ terms which yields

$$
\left\{ \frac{\mu_p}{\mu_{x_{i,j+\frac{1}{2},k+\frac{1}{2}}}} \left[\frac{1}{(e_{yj}\Delta y_j)} \left(Ez^s_{i,j+1,k+\frac{1}{2}} - Ez^s_{i,j,k+\frac{1}{2}} \right) - \frac{1}{(e_{zk}\Delta z_k)} \left(Ey^s_{i,j+\frac{1}{2},k+1} - Ey^s_{i,j+\frac{1}{2},k} \right) \right] - \right.
$$

$$
\left. \frac{\mu_p}{\mu_{x_{i,j+\frac{1}{2},k-\frac{1}{2}}}} \left[\frac{1}{(e_{yj}\Delta y_j)} \left(Ez^s_{i,j+1,k-\frac{1}{2}} - Ez^s_{i,j,k-\frac{1}{2}} \right) + \frac{1}{(e_{zk-1}\Delta z_{k-1})} \left(Ey^s_{i,j+\frac{1}{2},k} - Ey^s_{i,j+\frac{1}{2},k-1} \right) \right] \right\} \frac{1}{(h_{zk}\Delta\overline{z}_k)} +
$$

$$
\left\{ \frac{\mu_p}{\mu_{z_{i+\frac{1}{2},j+\frac{1}{2},k}}} \left[\frac{1}{(e_{yj}\Delta y_j)} \left(Ex^s_{i+\frac{1}{2},j+1,k} - Ex^s_{i+\frac{1}{2},j,k} \right) - \frac{1}{(e_{xi}\Delta x_i)} \left(Ey^s_{i+1,j+\frac{1}{2},k} - Ey^s_{i,j+\frac{1}{2},k} \right) \right] - \right.
$$

$$
\left. \frac{\mu_p}{\mu_{z_{i-\frac{1}{2},j+\frac{1}{2},k}}} \left[\frac{1}{(e_{yj}\Delta y_j)} \left(Ex^s_{i-\frac{1}{2},j+1,k} - Ex^s_{i-\frac{1}{2},j,k} \right) - \frac{1}{(e_{xi-1}\Delta x_{i-1})} \left(Ey^s_{i,j+\frac{1}{2},k} - Ey^s_{i-1,j+\frac{1}{2},k} \right) \right] \right\} \frac{1}{(h_{xi}\Delta\overline{x}_i)} +
$$

$$
i\omega\mu_p\hat{y}_p Ey^s_{i,j+\frac{1}{2},k} = -i\omega\mu_p \left(\hat{y}_{i,j+\frac{1}{2},k} - \hat{y}_p \right) Ey^p_{i,j+\frac{1}{2},k} - \tag{C-6}
$$

$$
i\omega\mu_p \left\{ \left[\left[\frac{\left(\mu_{x_{i,j+\frac{1}{2},k+\frac{1}{2}}} - \mu_p \right)}{\mu_{x_{i,j+\frac{1}{2},k+\frac{1}{2}}}} Hx^p_{i,j+\frac{1}{2},k+\frac{1}{2}} - \frac{\left(\mu_{x_{i,j+\frac{1}{2},k-\frac{1}{2}}} - \mu_p \right)}{\mu_{x_{i,j+\frac{1}{2},k-\frac{1}{2}}}} Hx^p_{i,j+\frac{1}{2},k-\frac{1}{2}} \right] \frac{1}{(h_{zk}\Delta\overline{z}_k)} - \right.
$$

$$
\left. \left[\frac{\left(\mu_{z_{i+\frac{1}{2},j+\frac{1}{2},k}} - \mu_p \right)}{\mu_{z_{i+\frac{1}{2},j+\frac{1}{2},k}}} Hz^p_{i+\frac{1}{2},j+\frac{1}{2},k} - \frac{\left(\mu_{z_{i-\frac{1}{2},j+\frac{1}{2},k}} - \mu_p \right)}{\mu_{z_{i-\frac{1}{2},j+\frac{1}{2},k}}} Hz^p_{i-\frac{1}{2},j+\frac{1}{2},k} \right] \frac{1}{(h_{xi}\Delta\overline{x}_i)} \right\}
$$

and

$$\left\{ \frac{\mu_p}{\mu_{y_{i+\frac{1}{2},j,k+\frac{1}{2}}}} \left[\frac{1}{(e_{zk}\Delta z_k)} \left(Ex^s_{i+\frac{1}{2},j,k+1} - Ex^s_{i+\frac{1}{2},j,k} \right) - \frac{1}{(e_{xi}\Delta x_i)} \left(Ez^s_{i+1,j,k+\frac{1}{2}} - Ez^s_{i,j,k+\frac{1}{2}} \right) \right] - \right.$$

$$\left. \frac{\mu_p}{\mu_{y_{i-\frac{1}{2},j,k+\frac{1}{2}}}} \left[\frac{1}{(e_{zk}\Delta z_k)} \left(Ex^s_{i-\frac{1}{2},j,k+1} - Ex^s_{i-\frac{1}{2},j,k} \right) - \frac{1}{(e_{xi-1}\Delta x_{i-1})} \left(Ez^s_{i,j,k+\frac{1}{2}} - Ez^s_{i-1,j,k+\frac{1}{2}} \right) \right] \right\} \frac{1}{(h_{xi}\Delta \bar{x}_i)} +$$

$$\left\{ \frac{\mu_p}{\mu_{x_{i,j+\frac{1}{2},k+\frac{1}{2}}}} \left[\frac{1}{(e_{zk}\Delta z_k)} \left(Ey^s_{i,j+\frac{1}{2},k+1} - Ey^s_{i,j+\frac{1}{2},k} \right) - \frac{1}{(e_{yj}\Delta y_j)} \left(Ez^s_{i,j+1,k+\frac{1}{2}} - Ez^s_{i,j,k+\frac{1}{2}} \right) \right] - \right.$$

$$\left. \frac{\mu_p}{\mu_{x_{i,j-\frac{1}{2},k+\frac{1}{2}}}} \left[\frac{1}{(e_{zk}\Delta z_k)} \left(Ey^s_{i,j-\frac{1}{2},k+1} - Ey^s_{i,j-\frac{1}{2},k} \right) - \frac{1}{(e_{yj-1}\Delta y_{j-1})} \left(Ez^s_{i,j,k+\frac{1}{2}} - Ez^s_{i,j-1,k+\frac{1}{2}} \right) \right] \right\} \frac{1}{(h_{yj}\Delta \bar{y}_j)} +$$

$$i\omega\mu_p \hat{y}_p Ez^s_{i,j,k+\frac{1}{2}} = -i\omega\mu_p \left(\hat{y}_{i,j,k+\frac{1}{2}} - \hat{y}_p \right) Ez^p_{i,j,k+\frac{1}{2}} - \tag{C-7}$$

$$i\omega\mu_p \left\{ \left[\frac{\left(\mu_{x_{i,j+\frac{1}{2},k+\frac{1}{2}}} - \mu_p \right)}{\mu_{x_{i,j+\frac{1}{2},k+\frac{1}{2}}}} Hx^p_{i,j+\frac{1}{2},k+\frac{1}{2}} - \frac{\left(\mu_{x_{i,j-\frac{1}{2},k+\frac{1}{2}}} - \mu_p \right)}{\mu_{x_{i,j-\frac{1}{2},k+\frac{1}{2}}}} Hx^p_{i,j-\frac{1}{2},k+\frac{1}{2}} \right] \frac{1}{(h_{yj}\Delta \bar{y}_j)} - \right.$$

$$\left. \left[\frac{\left(\mu_{y_{i+\frac{1}{2},j,k+\frac{1}{2}}} - \mu_p \right)}{\mu_{y_{i+\frac{1}{2},j,k+\frac{1}{2}}}} Hy^p_{i+\frac{1}{2},j,k+\frac{1}{2}} - \frac{\left(\mu_{y_{i-\frac{1}{2},j,k+\frac{1}{2}}} - \mu_p \right)}{\mu_{y_{i-\frac{1}{2},j,k+\frac{1}{2}}}} Hy^p_{i-\frac{1}{2},j,k+\frac{1}{2}} \right] \frac{1}{(h_{xi}\Delta \bar{x}_i)} \right\}$$

respectively. Unfortunately the above equations will not produce a symmetric matrix. Thus symmetric scaling must be applied with equation (C-5) being multiplied by $(e_{xi}\Delta x_i)(h_{yj}\Delta \bar{y}_j)(h_{zk}\Delta \bar{z}_k)$, equation (C-6) by $(h_{xi}\Delta \bar{x}_i)(e_{yj}\Delta y_j)(h_{zk}\Delta \bar{z}_k)$, and (C-7) by $(h_{xi}\Delta \bar{x}_i)(h_{yj}\Delta \bar{y}_j)(e_{zk}\Delta z_k)$.

APPENDIX D

The inverse problem can be formulated to allow for positive parameters with a lower bounding constraint by using a log parameterization. To accomplish this we first define a perturbation in the model as

$$\delta(m^{(i)}) = (m-m^{(i)}) \tag{D1}$$

and then use the differential form of the natural log function to write

$$\delta(m^{(i)}) = (m^{(i)}-\epsilon) \, \delta \ln(m-\epsilon), \tag{D2}$$

where

$$\delta \ln(m-\epsilon) = \ln((m-\epsilon)) - \ln(m^{(i)}-\epsilon)), \tag{D3}$$

with m and $m^{(i)} > \epsilon$, and $\epsilon > 0$. Following the form of equation (20) and using equations (D1), (D2) and (D3) we can define a new functional,

$$S' = [(D((d-d^{P(i)})-A'^{P(i)}\delta \ln(m-\epsilon)))^t \, (D((d-d^{P(i)})-A'^{P(i)}\delta \ln(m-\epsilon))) - \chi^2]$$
$$+ \lambda(W\ln(m-\epsilon))^t(W\ln(m-\epsilon)), \tag{D4}$$

where the modified Jacobian matrix $A'^{P(i)}$ is obtained by multiplying by columns of the original matrix with elements of the vector $(m^{(i)}-\epsilon)$. Minimizing the above expression with respect to $\ln((m-\epsilon))$, enforces the lower bound positivity constraint, where

$$\ln(m-\epsilon) = [(DA'^{P(i)})^t \, (DA'^{P(i)}) + \lambda(W)^t(W)]^{-1} \, (DA'^{P(i)})^t \, (D\delta d'^{(i)}) \tag{D5}$$

and

$$\delta d'^{(i)} = (d - d^{P(i)} + A'^{P(i)}\ln(m^{(i)}-\epsilon)). \tag{D6}$$

103

Once $\ln(\mathbf{m}-\epsilon)$ is determined, the parameters themselves follow from the expression

$$\mathbf{m} = e^{\ln(\mathbf{m}-\epsilon)} + \epsilon. \tag{D7}$$

With this new formulation, the inversion process is designed to deliver smooth estimates of $\ln(\mathbf{m}-\epsilon)$. Nevertheless, with a prudent selection of the regularization parameter, we can also expect smooth reconstructions for the model parameters, $\mathbf{m}$, themselves.

APPENDIX E

Computational efficient and compact forms of the two matrix-vector multiplies are necessary if the inversion is to be practical. Consider fully expressing the matrix vector multiply in equation (28) as,

$$y_j = \sum_{k=1}^{M} D_{jj} A_{jk} u_k, \tag{E1}$$

where the summation is over M electrical parameters. The entry D_{jj} is the jth entry of the data weighting matrix and A_{jk} is an element of the Jacobian matrix. These elements are assumed to be real valued, where real and imaginary components are treated as separate elements. Because the real and imaginary components of the Jacobian matrix are jointly expressed in equation (27) as

$$\partial d_j / \partial m_k = g_j^t K^{-1}(\partial s / \partial m_k - \partial K / \partial m_k E_s), \tag{E2}$$

we can redefine the data weighting matrix, **D**, to be complex to arrive at the following expression for the first matrix-vector multiply:

$$y_j = \text{Cmplx}(\text{Re}(g_j^t K^{-1} \sum_{k=1}^{M} u_k(\partial s / \partial m_k - \partial K / \partial m_k E_s))\text{Re}(D_{jj}),$$

$$\text{Im}(g_j^t K^{-1} \sum_{k=1}^{M} u_k(\partial s / \partial m_k - \partial K / \partial m_k E_s))\text{Im}(D_{jj})), \tag{E3}$$

where y_j is now assumed to be complex instead of real.

The second matrix-vector multiply in equation (29) can be expressed as

105

$$z_k = \sum_{j=1}^{2N} A_{jk} D_{jj} y_j, \qquad\qquad \text{(E4)}$$

where 2N is the number of data used in the inversion and where components subscripts $j=1,N$ correspond to real entries while components $j=N+1,2N$ correspond to imaginary ones; note quantities, D_{jj} and y_j are regarded as real valued by this splitting. By associating real and imaginary components as a joint term in the above summation, we can also express equation (E4) as

$$z_k = \sum_{j=1}^{N} (A_{jk} D_{jj} y_j + A_{j+N\,k} D_{j+N\,j+N} y_{j+N}). \qquad\qquad \text{(E5)}$$

Next combining elements as $cmplx(A_{jk}, A_{j+N\,k})$, $cmplx(D_{jj}, D_{j+N\,j+N})$ and $cmplx(y_j, y_{j+N})$ and because z_k must always be real, we find

$$z_k = \text{Re} \sum_{j=1}^{N} cmplx(D_{jj}\, y_j,\ D_{j+N\,j+N}\, y_{j+N})^* \, cmplx(A_{jk}, A_{j+N\,k}), \qquad\qquad \text{(E6)}$$

where '*' stands for complex conjugation. By treating A_{jk}, D_{jj} and y_j as complex and using equation (E2), we finally arrive at

$$z_k = \text{Re}(\sum_{j=1}^{N} Cmplx(\text{Re}(D_{jj})\, \text{Re}(y_j),\ \text{Im}(D_{jj})\, \text{Im}(y_j))^* \, (g_i^t\, K^{-1}\, (\partial s/\partial m_k - \partial K/\partial m_k E_*))). \qquad \text{(E7)}$$

APPENDIX F

To show that the vector $\partial s/\partial m_k$ and matrix $\partial K/\partial m_k$ each has 12 non-zero entries, we start with the vector Helmholtz equation for the scattered electric field, E_s, (equation (7)), but we will modify it such that magnetic permeability changes from free space, μ_0, are minimal. Thus,

$$\nabla \times \nabla \times E_s(r) + i\omega\mu_0(\sigma(r)+i\omega\epsilon(r))E_s(r) = -i\omega\mu_0 J_s(r), \qquad (F1)$$

with the source of the scattering given by

$$J_s(r) = \{(\sigma(r)-\sigma^b(r)) + i\omega(\epsilon(r)-\epsilon^b(r))\}E^b(r). \qquad (F2)$$

Here we have assumed an $e^{i\omega t}$ time dependence with $i=\sqrt{-1}$, where ω represents the angular frequency. In equations (F1) and (F2) the 3-D conductivity and permittivity variations are given by $\sigma(r)$ and $\epsilon(r)$, with $\sigma^b(r)$ and $\epsilon^b(r)$ representing the corresponding background properties, which for purposes here are either a whole space or a layered space. The electric field of the background media, $E^b(r)$, drives the source vector, and arises from an impressed dipole source.

The scattered fields are determined by imposing a staggered finite difference approximation on equation (F1), using a rectangular grid with a Dirichlet boundary condition. Each cell in this grid has a conductivity and dielectric permittivity assigned to it, where the scattered and source fields are sampled at the edges of the cell as illustrated in Figure 20. Because of this sampling scheme the averaged electrical properties have to be determined at the cell edges (refer to Chapter II for the details). These averages can be evaluated by tracing out a line integral of the magnetic field centered on the midpoint of the cell edge. The resulting average conductivity and permittivity are simply a weighted sum of the conductivities and permittivities of the four adjoining cells, where the weighting is based on the area of each cell that is bounded by the line integral. This is simple application of Ampere's Law. A study of Figure 20 shows that with the twelve field samples, equations (F1) and (F2) will require 12 averages of conductivity and permittivity, with each average involving the conductivity and permittivity of the indicated cell. Since with every field sample, we have one equation in the linear system, $KE_s = s$, where $s = J_s$, it follows that $\partial s/\partial m_k$ and the matrix $\partial K/\partial m_k$ each have 12 non-zero entries.

Distribution:

1   University of Arizona
       MGE Dept. Bldg. #12
       Attn: Douglas J. LaBreque
       Tucson, AZ 85721

1   Jim Wait
       2210 E Waverly
       Tucson, AZ 85719-3848

2   University of California
       Engineering Geoscience Division
       Attn: Frank Morrison
       Alex Becker
       567 Evans Hall, MS-1760
       Berkeley, CA 94720

1   UC Berkeley
       Dept. of Material Science and Mining Engineering
       Attn: Michael Hoversten
       577 Evans Hall
       Berkeley, CA 94720

2   University of California
       Earth Sciences Div. MS 50E 4237
       Lawrence Berkeley National Laboratory
       Attn: Ki Ha Lee
            Louise Pellerin
       1 Cyclotron Road
       Berkeley, CA 94720

1   Electromagnetic Instruments Inc.
       Attn: Edward Nichols
       P.O. Box 463
       El Cerrito, CA 94530-04631

2   Newmont Exploration Ltd.
       Attn: Perry Eaton
       Misac Nabigihian
       1700 Lincoln St.
       Denver, CO 80203

Distribution (continued):

1 Colorado School of Mines
   Department of Geophysics
   Attn: Gary Olhoeft
   Golden, CO 80401-1887

4 Schlumberger Doll Research
   Attn: Vladimir Druskin
   Tarek Habashy, Michael Oristaglio
    and Brian R. Spies
   Old Quarry Road
   Ridgefield, CT 06877-4108

2 Department of Energy
   Office of Research and Development, NN-20
   Attn: Michael O'Connell and Karl Veith
   1000 Independence Ave. SW
   Washington, D.C. 20585-0420

1 Idaho National Engineering Laboratory
   LITCO
   Attn: Cathy Pfeifer
   P.O. Box 1625
   Idaho Falls, ID 83415

1 US DOE Idaho Operations
   Attn: George J. Schneider
   785 DOE Place
   MS 1119
   Idaho Falls, ID 83402

1 University of Illinois
   Dept. of Electrical and Computer Engineering
   Electromagnetics Laboratory
   Attn: Weng Chew
   Urbana, IL 61801

1 US Environmental Protection Agency
   Environmental Monitoring Systems Lab
   Attn: Aldo T. Mazzella
   Box 93478
   Las Vegas, NV 89193-3478

Distribution (continued):

1   U.S. Department of Energy
Office of Energy Research
Attn: William C. Luth
ER-622
9901 Germantown Road
Germantown, MD 20874-1290

1   Earth Resources Laboratory
Department of Earth, Atmospheric and Planetary Sciences
Attn: Randall L. Mackie
42 Carelton Street
Cambridge, MA 02142

1   Geodynamics Research Institute
Texas A&M University
Attn: Mark E, Evertt
College Station, TX 77843-3114

1   Haliburton Energy Services Inc.
Attn: Mark Haugland
2135 Highway 6 South
Houston, TX 77077

1   Schlumberger LWD
Attn: John R. Lovell
200 Schlumberger Blvd. #110
Sugarland, TX 77478-3129

2   Western Atlas Inc.
Atlas Wireline Services
  Attn: Kurt Strack
      Lev Tabarovsky
P.O. Box 1407
Houston, TX 77251-1407

1   University of Utah
Dept. Geology and Geophysics
Attn: Michael Zhdanov
717 Browning Bldg.
Salt Lake City, UT 84112

Distribution (continued):

1   Stanley H. Ward
        2004 Highland Drive
        Anacortes, WA 98221

1   Noranda Technology Centre
        Attn: John McGaughey
        240 Hymus Blvd.
        Point-Claire, Quebec
        CANADA H9R 1G5

1   Department of Geophysics and Astronomy
        University of British Columbia
        Attn: Douglas W. Oldenburg
        129-2219 Main Mall
        Vancouver, B.C.,
        CANADA V6T 1Z4

1   Ricardo Fernandez
        Geodatos S.A.I.C
        Roman Diaz Nx 773
        Providencia
        Santiago
        Chile

1   Andreas Hoerdt
        University of Cologne
        Institute for Geophysics and Meteorology
        Godesberger Str. 10
        50968 Cologne
        Germany

1   Mark Goldman
        IPRG
        P.O. Box 2286
        Holon 58122
        Israel

Distribution (continued):

1   Evert Slob
       Delft University of Technology
       Fac. of Mining and Petroleum Engineering
       Section of Applied Geophysics
       Mijnbouwstraat 120
       2628 RX Delft
       The Netherlands

	MS 1111	Scott Hutchinson, 1421
	MS 1111	John Shadid,1421
	MS 1110	Lydie Prevost, 1422
	MS 1110	Ray Tuminaro, 1422
	MS 0828	Paul Hommert, 1503
	MS 0533	Billy Brock, 2343
	MS 0533	William Schaedla, 2343
	MS 0865	Kimball Merewether, 2753
	MS 0865	Larry Warne, 2753
	MS 0701	Richard Lynch, 6100
	MS 0705	Lew Bartel, 6114
	MS 0705	Thurlow Caffey, 6114
	MS 0483	Bob Glass, 6115
5	MS 0750	David L. Alumabugh, 6116
5	MS 0750	Gregory A. Newman, 6116
	MS 0843	Daniel Cress, 9131
	MS 1110	David Womble, 1422
	MS 1166	Doug Riley, 9352

1	MS 9018	Central Technical Files, 8523-2,
5	MS 0899	Technical Library, 4414,
1	MS 0619	Print Media, 12615
2	MS 0100	Document Processing, 7613-2
		For DOE/OSTI (2)