DEVELOPMENT OF MATLAB BASED 3D INVERSION ALGORITHM FOR MT AND DCR DATA

Ph. D. THESIS

by

ARUN SINGH

DEPARTMENT OF EARTH SCIENCES
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE-247 667 (INDIA)
AUGUST, 2018

DEVELOPMENT OF MATLAB BASED 3D INVERSION ALGORITHM FOR MT AND DCR DATA

A THESIS

Submitted in partial fulfilment of the requirements for the award of the degree

of

DOCTOR OF PHILOSOPHY

in

EARTH SCIENCES

by

ARUN SINGH

DEPARTMENT OF EARTH SCIENCES
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE-247 667 (INDIA)
AUGUST, 2018

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled "**DEVELOPMENT OF MATLAB BASED 3D INVERSION ALGORITHM FOR MT AND DCR DATA**" in partial fulfillment of the requirements for the award of the Degree of Doctor of Philosophy and submitted in the Department of Earth Sciences of the Indian Institute of Technology Roorkee is an authentic record of my own work carried out during a period from July, 2014 to March, 2018 under the supervision of Dr. Pravin K. Gupta, Professor and Dr. M. Israil, Professor, Department of Earth Sciences, Indian Institute of Technology Roorkee, Roorkee.

The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

(ARUN SINGH)

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

(Pravin K. Gupta) (M. Israil)
Supervisor Supervisor

The Ph.D. Viva-Voce Examination of **Arun Singh**, Research Scholar, has been held on August, 2018.

Chairman, SRC Signature of External Examiner

This is to certify that the student has made all the corrections in the thesis.

Signature of Supervisor (s) Head of the Department

Dated: August , 2018

ABSTRACT

In this study, I present the development of a MATLAB based computer code, *AP3DMT*, for modeling and inversion of 3D Magnetotelluric (MT) and Direct Current Resistivity (DCR) data. This code can be used to invert just the MT data or the DCR, or invert both data sets simultaneously.

A 3D MT forward modeling code based on finite difference (FD) method for solving the vector Helmholtz equation, expressed in electrical field, is developed. The staggered grid is used for accurate simulation of nodal electric fields. The electric fields can be computed either using total field or primary/secondary approach. For efficiency, in terms of memory and computation time, the matrix equation is solved using iterative solver with incomplete **LU** decomposition as pre-conditioner. The convergence of iterative solver is further improved by using the static divergence correction.

The developed code, *AP3DMT*, comprises two independent components: grid generator code and modeling/inversion code. The grid generator code performs model discretization and acts as an interface by generating various I/O files. The inversion code performs core computations in modular form – forward modeling, data functionals, sensitivity computations and regularization. These modules can be readily extended to other similar inverse problems like Controlled-Source EM (CSEM), DCR (implemented by the author). The modular structure of the code provides a framework useful for implementation of new applications and inversion algorithms. The use of MATLAB and its libraries makes it more compact and user friendly.

The inversion code includes Gauss-Newton optimization (model space as well as data space) and non-linear inversion using conjugate gradient. In both these schemes, Jacobian is not computed explicitly, rather product of Jacobian (or its transpose) with a vector is computed. Special emphasis is given on the block representation of Jacobian for a multi-frequency, multi-component data set and its product with a vector using its three components. It is shown how, for Jacobian, the matrix formed by differentiation of system matrix is made independent of frequency to optimize the operations. A coarse grained parallelization is implemented over number of frequencies for both forward modeling and sensitivity computations.

The code developed in this study, has been tested on several published models. To demonstrate the versatility of the code, the accuracy of the simulated responses is verified by comparing the responses obtained using a different code and inversion was performed for two complex synthetic models. Further, the code was tested on field data. The data set was acquired over the past decade using Broadband MT survey by our group along the Roorkee-Gangotri profile in the Indo-Gangetic plain, Sub Himalayan and Lesser Himalayan region. The dataset was inverted using *ModEM* and using *AP3DMT* by the author. For comparison a 3D diagram of the ratio of inverted cell conductivities in two cases is presented.

During the development of divergence correction routine, we observed that this module can be extended to DCR modeling. The developed 3D DCR forward modeling code is based on FD method. The nodal potential are simulated on a normal grids. To strike a balance between computational time and accurate solutions two nodes are used between adjacent electrodes. For removal of singularity due to a point source primary/secondary approach is used. The primary potentials are computed analytically for half space. For secondary potentials, the matrix equation is solved using iterative solver or direct solver depending on the size of the matrix and available resources. The accuracy of the simulated response is verified by comparing the responses with published results. The relevant portions of original *AP3DMT* code was modified to incorporate DCR inversion. Since, *AP3DMT* had a modular structure, new modification were made only in forward problem and Jacobian with other portions unchanged. The product of Jacobian with a vector is efficiently managed. The versatility and capabilities of the inversion code was tested against two different models.

To make further progress in data interpretation, 3D joint inversion of MT and DCR is developed. The modular structure of *AP3DMT* code proves to be very useful in performing this task. Since, both these methods has different depth of penetrations model discretization is very crucial. For the joint inversion with each set having different number of data points, data weights needs to be re-calculated, if not then the more numerous data set of one type can cause the influence of another data on the imaging outcome to become insignificant. Two different schemes based on number of data points and gradient are tested. The joint inversion resolves the model better as compared to inversion of individual data sets. The importance of joint inversion is further demonstrated through a synthetic model. Also, both schemes of data weights re-computations is compared on the same model.

LIST OF PUBLICATIONS

Journals

Arun Singh, Rahul Dehiya, Pravin K. Gupta and Mohammad Israil, 2017. A MATLAB based 3D modeling and inversion code for MT data, *Computers & Geosciences*, (104): 1–11.

Rahul Dehiya, **Arun Singh**, Pravin K. Gupta and Mohammad Israil, 2017. Optimization of computations for adjoint field and Jacobian needed in 3D CSEM inversion, *Journal of Applied Geophysics*, (136): 444–454.

Rahul Dehiya, **Arun Singh**, Pravin K. Gupta and Mohammad Israil, 2017. 3D CSEM data inversion algorithm based on simultaneously active multiple transmitters concept, *Geophysical Journal International*, (2017): 1004-1017.

Conference/Workshop

Mohammad Israil, **Arun Singh**, Anita Devi, Pravin K. Gupta, and Mohammad Israil, 2018. MATLAB based code for 3D joint inversion of Magnetotelluric and Direct Current Resistivity imaging data, *EMIW*, Helsingør, Denmark.

Anita Devi, Mohammad Israil, **Arun Singh**, Pravin K. Gupta, and Mohammad Israil, 2018. Geothermal inferences drawn from 3D inversion of Magnetotelluric Data recorded from Chamoli region, Uttarakhand, India, *EMIW*, Helsingør, Denmark.

Mohammad Israil, Anita Devi, **Arun Singh** and Pravin K. Gupta, 2017. 3D Geoelectrical model of Sub-Himalayan Region − A Magnetotelluric study, *SPG*, Jaipur, India.

Arun Singh, Nitin Kumar, Pravin K. Gupta, and Mohammad Israil, 2016. A MATLAB based 3D modeling and inversion code for MT data, *EMIW*, Chiang Mai, Thailand.

Pravin K. Gupta, Rahul Dehiya, **Arun Singh** and Mohammad Israil, 2016. Efficient computation of adjoint field for 3D CSEM inversion, *EMIW*, Chiang Mai, Thailand.

Arun Singh, Rahul Dehiya, Pravin K. Gupta and Mohammad Israil, 2014. Development of block Inversion algorithm and its comparison with cell inversion schemes, *EMIW*, Weimar, Germany.

Rahul Dehiya, **Arun Singh**, Pravin K. Gupta and Mohammad Israil, 2014. Interpretation of CSEM data using 2D block inversion algorithm, *EMIW*, Weimar, Germany.

Acknowledgements

First and foremost, I would like to thank my supervisors Prof. Pravin K. Gupta and Prof. Mohd. Israil for their guidance, support and encouragement throughout my work at Indian Institute of Technology Roorkee, Roorkee and at Geological survey of India, Kolkata and making the completion of this thesis possible. It has been a privilege and pleasure being his student.

I thank my research committee members, Prof. Sandeep Singh, Prof. J. P. Narayan, Dr. Kamal, and honorable chairman of department research committee Prof. Mohd. Israil for always being there to help and provide general guidance, not to mention helping meet all the document deadlines.

During this research, the department has had three heads; Prof. A. K. Saraf, Prof. D. C. Srivastava and Prof. Sunil Bajpai and I sincerely acknowledge the support they provided by extending every sort of help as and when sought for.

I extend a special note of thanks to all faculty members of the department for their co-operation. I also thank, all the non-teaching staff of the department, especially Mr. Nayar (retd.) and Mr. Rishiraj who have helped me in all possible ways for the official work.

I express my heartily gratitude to Smt. Mamta Gupta and Smt. Saeeda as time spent occasionally during tea, dinner was very pleasant.

I thank Director General, Geological Survey of India for allowing to continue my Ph.D. as part time. I thank Add. Director General (Geophysics) and Director, Geophysics Division, Eastern Region, Kolkata for all their support. I thank my colleagues at office and my party-in-charge and party members during three field work for all their support and encouragement.

The financial support provided by Ministry of Human Resources and Development (MHRD), New Delhi to complete the present work is highly acknowledged. I am also thankful to Ministry of Earth Sciences, India for providing computational facilities. I thank Institute Computer Center, IIT Roorkee for providing MATLAB-2013. I am thanful to Dr. Gary Egbert and Dr. Anna Kelbert for providing *ModEM* code for comparison of results.

A special thanks to my former and current lab-mates Dr. Vishal, Dr. Rahul, Anita, Nitin, Pramjeet and Zubair for all the enlightening discussions we had during these years. I also

thank all my other friends from IIT Roorkee, who helped lighten the burden, especially, Nupur, Zia, Meenakshi, Neha, Amit, Aanand, Sohanlal, and Dr. Vinay. A special thanks to my colleagues at Geological Survey of India, Kolkata, Pramod, Raja, Anurag, Sneh, Vikash, Sandeep, Amitabh and Gokul for encouraging me in my pursuit of research while being in job.

A special thanks to my younger brothers Sanjeev, Gaurav and my sister Sunidhi for their unconditional love and support during crucial times.

Finally, I express my heartfelt gratitude to my father, Shri N. P. Singh and mother, Smt. Pravesh Singh for their unconditional love, encouragement and blessings. They have been a guiding force all my life and have given me necessary energy and happiness to work hard throughout this research work. I also express my feelings of gratitude to all those who helped me in this course but have not been listed here.

Roorkee (Arun Singh)

August 30, 2018

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

List of notation used.

Symbols	*Notations*
$\mathbf{E}$	electric field
$\mathbf{D}$	electric displacement current
$\mathbf{B}$	magnetic flux density
$\mathbf{H}$	magnetic field
μ	magnetic permeability
ε	electric permittivity
σ	electric conductivity
ρ	electric resistivity
ω	angular frequency
i	$\sqrt{-1}$
f	frequency
N_e	total number of internal nodes
N_d	total number of data points
$\mathbf{A_m}$	coefficient matrix
$\mathbf{e}$	nodal electric field vector
$\mathbf{s}$	source vector
ϕ	potential field
$\mathbf{Z}$	impedance tensor
$T_x,\ T_y$	vertical magnetic field transfer functions
$\tilde{\mathbf{m}}$, $\mathbf{m}$	model parameter vector
$\mathbf{m_0}$	apriori model
$\tilde{\mathbf{d}}^{obs}$, $\mathbf{d}^{obs}$	observed data
$\tilde{F}(\tilde{\mathbf{m}})$, $F(\mathbf{m})$	forward mapping
$\mathbf{C_d}$	data covariance matrix
$\mathbf{C_m}$	model covariance matrix
$\mathbf{r}$	residual
$\mathbf{J}$	sensitivity matrix
$\mathbf{L}_f$	sensitivity functional for single frequency
$\mathbf{A}^{-1}_{\mathbf{m},f}$	forward solver for single frequency
$\mathbf{P}_f$	solver sensitivity for single frequency

Chapter 1

INTRODUCTION

1.1 Preamble

The geoelectromagnetic method is an important branch of geophysics, in addition to seismic, gravity, magnetic etc. This method is used to recover a conductivity distribution of the earth's interior thus adding a third dimension to geological maps. The electrical conductivity distribution is translated in terms of geological formulation. This requires understanding of the link between the physical properties, conductivity and lithology, alteration and other petrophysical properties[99].

The conductivity of a rocks affect the geoelectromagnetic response to artificially or naturally stimulated static or time varying electromagnetic fields. The artificially stimulated source field methods are Controlled Source Electromagnetic (CSEM), Direct Current Resistivity (DCR) and Induced Polarization (IP), etc. whereas the naturally simulated methods are Magnetotelluric (MT), Telluric, Geomagnetic Depth Sounding and Self Potential (SP).

The source of MT method are the natural electromagnetic (EM) waves in the frequency range $10^{-5} - 10^{5}$ Hz. These fields are mainly associated with thunderstorm (> 1 Hz) and interaction of solar wind with the earth's magnetosphere (< 1 Hz)[68]. The orthogonal component of time varying electric and magnetic field are measured in the form of time series. The electric field components are linearly related to magnetic field through transfer function [16, 179].

The DCR method is one of the oldest and simplest method to investigate shallow subsurface structures. An electric field is set up by putting steady current into the ground

through a pair of electrodes. Another pair of electrodes is used to measure the field in the form of the electrical potential differences. When an electric current field is introduced into the earth, it sets up a distribution of accumulated electric charges both on and beneath the earth's surface. These charges exist in the region where there is a gradient of conductivity and a non-zero field crossing the electric boundaries. It is these accumulated charges that give rise to the electric potential that is measured. Many different electrode configuration can be used for field measurement.

1.2 Brief review of MT and DCR Literature

The available literature on MT and DCR is very vast and it is impossible to review all the works hence, I have reviewed only selected works. For theory, data acquisition, processing and interpretations one can refer various textbooks [9, 10, 21, 69, 107, 189, 207, 208] written on this subject. First, I will give a very brief literature review of techniques essential for interpretation of recorded data (modeling and data inversion) and later selected case studies. For obtaining resistivity distribution of the earth's interior, the MT transfer function in the form of impedances tensor or apparent resistivity and phase and vertical magnetic field transfer function (VTFs) and DCR apparent resistivity response are inverted individually or jointly using well defined mathematical techniques. This requires efficient and accurate modeling and inversion computer codes. The development of efficient, robust, versatile and user friendly computer code for modeling and inversion of 3D MT and DCR data is a subject matter of present thesis. The main focus is on the MT methods however, DCR is also included within MT, as one of the sub-program of the developed 3D MT inversion code was extended for DCR data.

1.2.1 3D MT and DCR Forward Modeling

The computation of responses of a model (forward modeling) is essential for the interpretation of the data. The analytic solutions are only available for a selected class of 3D models. However, for a general conductivity distribution, as encountered in nature, one has to used numerical techniques. The commonly employed numerical techniques for 3D MT and DCR forward modeling are:

1. *Integral Equation Method (IEM)*

In IEM, the Maxwell's equations are first reduced to second-kind Fredholm's integral equation with respect to electric field. This is known as the scattering equation. The Green's function technique is generally used to derive the scattering equation. In IEM only the target body is discretised, hence, the resulting system matrix is small but dense. Therefore, IEM are better suited for computing responses of confined targets such as conductive bodies in a half space or horizontally layered earth. Several developers have used the IEM for MT modeling [e.g. 49, 60, 190, 197, 209] and for DCR modeling [e.g. 35, 82, 116, 198] However, for a general conductivity structure, the Differential Equation Methods (DEM) are preferred.

2. *Finite Element Method (FEM)*

   In FEM, which is still not widely used, the fields are expressed as a superposition of some basic (generally, edge and nodal) functions. The main advantage of FEM is the efficient and accurate handling of curved boundaries (e.g., topography, shapes of ore-bodies etc.). However, this advantage is counterbalanced by the tedious and time consuming process of construction of the finite elements. Several workers have implemented the FEM technique for MT modeling [e.g. 42, 104, 108, 132, 134] and for DCR modeling [e.g. 12, 23, 52, 128, 139, 203].

3. *Finite Difference Method (FDM)*

   FDM is one of the most commonly used approach. For accurate solution of Maxwell's equation, the staggered grid is used [201]. If the PDE is formulated in terms of electric field components, these are defined on cell edges and the magnetic field components are defined on center of the cell faces and vice-versa. The matrix formed using staggered grid is sparse (13 non-zero elements per row), symmetric and non-hermitian having complex diagonal elements. Various forward modeling algorithms using FDM for solving MT problem have been developed [e.g. 43, 63, 89–91, 110, 140, 158, 164, 165, 200]. For DCR problem, normal grid is used. Dey and Morrison [33] developed a 3D FD algorithm to evaluate the potential for a point current source. Since then various algorithms have been developed [e.g. 87, 88, 166, 168, 205].

Other approaches such as mesh-free modeling [195], have also been implemented for MT fields computations.

Solution of System Matrix

Regardless of which numerical scheme is employed for solving the MT and DCR forward problem, the formulation is reduced to a system of linear equations that can be solved by an appropriate matrix solver. For FEM and FDM, the size of the system matrix is large hence, the iterative solvers are preferred over direct solvers as these solvers need less memory and computation time. Direct solvers may be efficient if solutions for multiple right-hand side vectors are needed for same system matrix for example in CSEM [169] or in DCR problem. The large memory requirement for direct solvers is still a challenge, particularly in case of large survey size. Therefore, the semi-iterative solvers based on Krylov subspace method are widely used in forward modelling. These solvers only require matrix-vector multiplication. Since, the system matrix is highly sparse, 13 non-zeros elements per row for MT and 7 for DCR, matrix-vector multiplication is easy to compute. The MT system matrix is generally complex and non-hermitian. For solving such a linear system, some of the popular iterative solvers are the generalised minimal residual (GMRES) [137], quasi-minimal residual (QMR) [44] and biconjugate gradient stabilised (BiCGSTAB) [186].

These solvers differ in both memory requirement, number of computations in each iteration and robustness. GMRES is a well-known Arnoldi-based method proposed by Saad and Schultz [137]. The GMRES leads to a non-increasing sequence of residual norms and, therefore, it always guarantees smooth and monotonically decreasing convergence, which may not necessarily be fast enough. The GMRES requires one matrix-vector multiplication per iteration. The main disadvantage of GMRES is its large storage requirement because the solver stores all previously-generated Arnoldi vectors. To over come this issue some modification has been proposed as restarted-GMRES and hybrid GMRES. QMR [44] and BiCGSTAB [186] are two Lanczos-based methods. These methods require relatively less memory which does not vary during iterations. The QMR and BiCGSTAB methods may produce oscillatory behavior as far as residual norm is concerned. For more details about these methods readers can refer the textbook by Saad [136]. Another important issue regarding the iterative solver is the condition number of system matrix which could be of the order of 10^9 to 10^{12} [4]. In case of FDM and FEM, the most popular preconditioners are Jacobi, SSOR, incomplete LU decomposition and multigrid. The right choice of preconditioner is very crucial for efficient computation. In fact, the choice of preconditioner is more critical than the iterative solver. The literature survey suggests that the incomplete

LU preconditioner is the most preferred one in MT modeling and is widely used by developers [4].

1.2.2 3D MT and DCR Inversion

All the inversion algorithms, including MT and DCR inversion, aim at finding a meaningful model, while fitting the data to an acceptable level in a stable manner. These inversion algorithm has to deal with some inherent complexities related to non-linearity and ill-posed nature of the problem. Further, due to the noisy and limited data, the inverse solution is non-unique hence, a priori information is provided to constrain the model. The constraints are applied in the form of a regularization functional. Regularization process also stabilizes the inverse problem which is unstable due to the ill-posed nature of the problem. The regularization of the objective functional [97] of the inverse problem is an important task. The common practice is to seek smooth model parameters. In many algorithms [e.g. 25, 110, 135], it is achieved by constructing the model covariance matrix using the finite difference approximation to the Laplacian ($\bigtriangledown^2$) operator or formulating regularization directly in terms of a smoothing operator [36, 155, 70]. The prior information such as faults or oceans can also be included in the regularization functional [155].

MT data acquisitions in field have usually been conducted along a profile or several profiles parallel to each other [e.g., 112, 182]. Dimensionality analyses of the data are applied in order to validate the 2D assumptions. These include skew analysis [170, 187], phase-sensitive skew analysis [6], the Groom and Bailey decomposition [50], tensor decomposition [22], rotational invariant analysis [193], strike decomposition [98], the phase tensor [17] and the Mohr circle [85]. For a review of 2D techniques including modeling, dimensional analysis and interpretation one can refer Ogawa [115]. Afterwards, 2D inversion is used to yield 2D cross-sectional models for profile interpretation. However, the 2D inversions are influenced by the 3D structure. Such effect has been demonstrated through synthetic data [80, 160] as well as real data [113, 149]. These studies indicate that if the data contains significant 3D effects, 2D inversion can mislead an interpretation. The ambiguity of the data can be overcome by using 3D inversion as it is not necessary to make an assumption about the strike direction although recommended. Instead of using 2D inversion, 3D inversion can also be performed for the 2D profile acquired previously as demonstrated by Siripunvaraporn et al. [160] for synthetic data and by Xiao et al. [196] for real data. Such

inversion can recover anomalies correctly as demonstrated by Newman et al. [113]. With 3D inversion, structures may be put outside the profile where they belong although their size, shape, depth and location cannot be determined exactly. This is in contrast to 2D inversion where structures are inserted beneath the profiles which could lead to misinterpretation [160]. Station locations need not be projected on the profile line for 3D inversion and hence a 3D model geometry is likely to be more reliable. For more advantages of 3D over 2D inversion readers can refer to Siripunvaraporn [154].

The disadvantage of 3D inversion is the large computational cost in terms of both computation time and memory. In 3D inversion the number of model parameters increases exponentially and are of the order of hundred-thousand to million, hence, the inherent complexities associated with large number of model parameters become more challenging. In spite of these challenges, several 3D inversion algorithms have been developed and used to accurately account for 3D nature of the model.

Many MT inversion algorithms have been developed in the past few decades. These algorithms, iterative in nature, includes the Occam's inversion [e.g. 25, 28, 159, 157, 161], the Gauss–Newton (*GN*) method [e.g. 54, 141, 142], the Gauss–Newton with the conjugate gradient (*GN-CG*) method [89, 109, 156], the quasi-Newton (*QN*) method [53, 5], and the non-linear conjugate gradient (*NLCG*) method [e.g. 24, 39, 70, 110, 135, 92]. These algorithms, except *NLCG*, uses some sort of variant of the Newton optimization method. The Newton type methods require computation of the Jacobian (sensitivity) matrix and Hessian (or its approximation) at each inversion iteration. However, for a 3D case, explicit computation of Jacobian or Hessian is computationally very expensive. The alternative approach is to bypass the formation of Jacobian by using memory efficient iterative Krylov-space solver such as conjugate gradients [e.g. 55, 56, 89, 109, 135, 156]. In *CG* method the product of Jacobian (or its transpose) and a vector is computed. Mackie and Madden [89] used this approach to solve for 3D MT inverse problem. In *NLCG* method, the gradient of the penalty functional is computed rather than the Jacobian matrix. *NLCG* requires only two forward and one gradient computation per source per inversion iteration. However, the less computation time per iteration is offset by slow convergence as compared to *GN* methods. Through 2D MT data inversion experiment, Rodi and Mackie [135] showed, that the *NLCG* takes more time than *GN* method and that preconditioned *NLCG* is equivalent to *GN* in terms of computation time. Pre-conditioned *NLCG* has also been implemented to

improve the convergence [110, 111, 135] but it increases the cost per iteration. Kelbert et al. [70] has solved the inverse problem using *NLCG* in model transform domain and observed that the convergence is similar to preconditioned *NLCG*. For detailed review on different 3D MT inversion algorithms readers can refer to Avdeev [4] and Siripunvaraporn [154]. Even with all these advancements, the solution of 3D MT problem will remain a very challenging and efforts are being focused on minimising these challenges.

Various techniques have been presented for 3D DCR data inversion over the last three decades. Petrick Jr et al. [125] determines the positions of conductive anomalies using the concept of alpha centers. Rijo [133] performed inversion by using data bank of forward solutions for a certain 3D models while inversion based on the Born approximation was illustrated by Li and Oldenburg [84]. Shima [148] used FEM to improve the result obtained using alpha centre method. Park and Van [118] published the 3D inversion scheme using FD and least square inversion. Since then various authors have presented inversion algorithms [e.g. 40, 52, 87, 139, 204]. In these algorithms, the forward solutions are generally obtained by FDM or FEM and inversion are based on Gauss–Newton (or its variant) technique. For the 3D case the problem is large hence, explicit computation and storage of the Jacobian (sensitivity) matrix is avoided [e.g. 204].

The inversion methods discussed above are fraught with the problem of non-uniqueness [100]. Data acquired in surveys is usually restricted to the surface of the Earth or the shallow subsurface, often with relatively large spacing between measurement sites, and affected by noise. In such a situation different models can explain the observed data, creating ambiguity in their interpretation [174]. Applying regularization stabilizes the inversion and creates a model with certain characteristics but does not alleviate the underlying problem [119]. Joint inversion approaches promise to reduce the set of acceptable models by combining several geophysical methods in a single inversion scheme and requiring the resulting model to explain all data simultaneously [188]. First, different methods have different resolving kernels and the null space for one type of data can be resolved by the other [65]; second, the sources of noise and its impact on the data often differ so that adding another method can improve the results more than adding more data of the same type. Consequently joint inversion approaches have gained some attention recently.

After a brief discussion modeling and inversion of MT and DCR data, I shall discuss their applications. MT and DCR methods are used in solving variety of real world problems

related with deep, intermediate and shallow depth in the earth's interior. Some of these are mentioned in the following.

1.2.3 Literature in Case Studies

As the depth of investigation of MT class of methods depends on the frequency band as well as the resistivity distribution in the earth's interior. By choosing appropriate frequency band, MT method can be used for variety of applications.

Through the preliminary inversion of USArray transportable array MT data, Patro and Egbert [123] observed extensive areas of high conductivity in the lower crust beneath all of southeastern Oregon, and beneath the Cascade Mountains, USA along with significant variations in the upper mantle conductivity. Ernst et al. [41] and Habibian et al. [57] carried out survey to study the deep structure of the Trans-European Suture Zone (under international EMTESZ-Pomerania Project) and observed highly conductive Cenozoic-Mesozoic sedimentary cover reaching depths up to 3 km. Smirnov and Pedersen [163] conducted MT experiments across the Sorgenfrei-Tornquist-Zone(STZ) (northwestern part of Baltic shield), a major branch of the TESZ and observed zone of enhanced conductivity in the lower crust and uppermost mantle (position coincides with STZ). Korja et al. [76] conducted broad-band MT along a 180 km long profile in Jämtland, Sweden. Their 2D inversion shows an electrically highly conducting layer beneath the Caledonides images alum shales over Precambrian basement. The INDEPTH (InterNational DEep Profiling of Tibet and the Himalaya) project provides insight of structure and evolution of Tibetan Plateau. Wei et al. [194] observed wide spread presence of high conductivity fluid at a depth 15-20 km in southern Tibet and at a depth of 30-40 km in the northern Tibet. Bai et al. [7] observed two major zones or channels of high electrical conductivity at a depth of 20-40 km and these channels extend horizontally more than 800 km from the Tibetan plateau into southwest China. Unsworth et al. [183] also observed crustal melting in Himalayas from northern Tibet side.

In India, The Main Frontal Thrust (MFT), the Main Boundary Thrust (MBT) and the Main Central Thrust (MCT) has been delineated in Gharwal Himalayas by Israil et al. [62], Miglani et al. [103] and Rawat et al. [131] and in Sikkim Himalayas by Patro and Harinarayana [124]. In these studies the nature of the low resistivity associated with the Main Himalayan Thrust in the higher Himalayas (north of the MCT) might indicate presence

of metamorphic fluids released due to under thrusting of the Indian plate. In Eastern Indian Craton, Bhattacharya and Shalivahan [11], Shalivahan and Bhattacharya [144] conducted MT study and were able to resolve the lower crust and upper mantle boundary. Such an identification of crust-mantle boundary was possible due to the absence of a conducting lower crust. In literature there are many other references available where MT method has been used for studying the crustal structure of earth Brasse et al. [14], Brasse and Soyer [15], Korja et al. [76], Patro et al. [122], Shalivahan and Bhattacharya [145] and Smirnov and Pedersen [163].

In Eastern India, geothermal study was carried out in the Bakreswar Hot Spring by Sinharay and Bhattacharya [152] and Sinharay et al. [153]. They reported that the north-south fault close to Bakreswar is a shallow feature and thus cannot act as a heat source. MT results indicate that the location of the geothermal reservoir is deep. Yamaya et al. [199] performed geothermal investigations in the crater area of Tarumai volcano, northeastern Japan. Their survey brought out two conductive structures beneath the lava dome of Tarumai volcano. The deeper and extremely conductive body corresponded to a convecting zone containing rising hydrothermal fluid while the shallow conductor was interpreted as an aquifer. This study provides an insight that the aquifer plays a significant role controlling the temperature and chemical components of erupting fluids. Komori et al. [75] has also documented the effect of the hydrothermal alteration on the surface conductivity of rock matrix. Various other researchers e.g., Cumming and Mackie [27], Jones et al. [64], Heise et al. [58, 59], Oskooi et al. [117], Takasugi et al. [173] have also used MT for geothermal studies.

Tezkan et al. [176] used the radio MT and transient EM to study the waste deposit site in Cologne (Germany) and observed that due to the highly conducting waste deposit only the top of the waste site and the lateral boundary of it could be resolved. Tezkan et al. [177] showed that the radiomagnetotelluric technique is a powerful tool for waste site exploration.

The DCR method have variety of application, mainly related with shallow subsurface investigations. DCR method has been extensively used for hydro-geophysical applications such as groundwater exploration, environmental. geotechnical, archaeological studies, etc.

Coscia et al. [26] used 3D crosshole ERT for investigated of the hydrogeological properties and responses of a productive aquifer in northeastern Switzerland. Pidlisecky et al. [127] determined the subsurface distribution of saltwater- and freshwater-saturated

sediments and the geologic controls on fluid distributions along the coast of Monterey Bay (6.8-km)

Ramirez et al. [130] used this technique for detection of leaks in underground storage tanks. The investigation of an abandoned mining dump which contains slag material originating from steel production were done by Günther et al. [52]. DCR studies has been successful used for the prediction of the location of future rupture surfaces of a slowly moving loess landslide along the River Danube in South Hungary [171, 172].

Resistivity prospecting was employed to detect the remnants of wall foundations in the place where the main urban complex of the ancient city of Europos, northern Greece once stood [181]. Argote-Espino et al. [3] used 3D ERT and identified four anomalous areas of interest at archaeological structures site of Teteles de Ocotitla, Tlaxcala, Mexico. The archaeological excavations confirmed the results obtained using the geophysical method.

To further improve interpretation, the MT data has been analyzed in combination of other geophysical methods like seismic. Through a synthetic data synthetic data Manglik and Verma [94] showed that the joint inversion yields better layer parameters, than the individual MT or seismic inversions. Sharma et al. [147] delineated sediments below flood basalts by using seismic and MT and later to delineate a four-layered crustal structure in the Southern Granulite Terrain, India [95]. Moorkamp et al. [106] presented joint inversion of MT, gravity and seismic refraction data and through a synthetic example demonstrated its usefulness.

Vozoff and Jupp [188] combined MT and Schlumberger soundings and inverting the data sets simultaneously for 1D structures. Other examples of lD joint inversion were presented by Gomez-Trevino and Edwards [48] for CSEM and Schlumberger sounding and by Raiche et al. [129] for coincident loop transient EM with Schlumberger sounding. 2D joint inversion were performed by various workers [e.g. 2, 20, 138]. Seher and Tezkan [143] used RMT and DCR measurements for the characterization of conducting soils. Yogeshwar et al. [202] studied the impact of sewage irrigation and groundwater contamination near Roorkee in north India using the RMT and DCR method. Tezkan et al. [176] has used RMT and transient EM for investigation of industrial and domestic waste sites in Germany.

1.3 Definition of Problem

The objective of this study is to develop an efficient, accurate and user friendly 3D forward modeling and inversion algorithm for MT data. The development of an accurate and efficient 3D MT inversion code is an active area of research and the forward modeling is an integral part of any inversion code. Literature survey reveals that there are possibilities of further development on following aspects (i) parallel implementation, (ii) efficient computation of sensitivity matrix, (iii) reduction in number of forward solver calls, (iv) generalized anisotropic model, and (v) joint inversion.

The present thesis focuses on development of MATLAB based code for 3D MT inversion and extension of the same code to DCR data and finally to joint inversion of MT and DCR data. A brief summary of the work done is presented below.

1.4 Thesis Layout

The thesis is organized into seven chapters briefly summarized below.

The first chapter is an introduction to the thesis. It carries a detailed literature review and it defines the problem.

In chapter 2, the mathematical formulation of forward and inverse problem, including a brief description of EM theory, is presented. It also formulates a boundary value problem comprising the governing partial differential equation and requisite boundary conditions. The finite difference method to solve the forward problem is presented along with a discussion on matrix solver. The expression for data sensitivities is presented and the Jacobian (or its transpose) matrix-vector multiplication is discussed in detail where the Jacobian is represented using block matrices.

Third chapter explains the salient features of the developed 3D inversion MATLAB code, *AP3DMT*, for MT data. The main emphasis is on modular implementation of the basic components of inversion - forward modeling, model regularization, data functionals, and sensitivity computation - which are reusable and readily extensible. The code has two components. This first component is a standalone grid generator. The main feature of grid generator is its robustness in handling complex geological features needed to simulate responses for complex 3D structures. The second component is the inversion algorithm.

It includes *GN* with conjugate gradients (data and model space) and *NLCG* optimization technique. The detailed description and implementation of forward modeling, inversion, sensitivity computations and other components are discussed here. The efficiency of inversion algorithm in the code is enhanced by representing the differentiated system matrix in a frequency independent mode. The product of Jacobain (or its transpose) with a vector is implemented through components of Jacobian. The coarse level parallel computing over frequency is implemented. The code's versatility for grid generation is demonstrated.

In fourth chapter inversion of two synthetic examples and field MT data is presented. Over the past decade Broadband MT survey was conducted by our group along the Roorkee-Gangotri profile in the Indo-Gangetic plain, Sub Himalayan and Lesser Himalayan region. The MT system used in this survey was Metronix, MT system. This dataset was inverted using *ModEM* [39] by Devi et al. [32] and using *AP3DMT* by the author. For comparison a 3D diagram of the ratio of inverted cell conductivities in two cases is presented.

Fifth chapter deals with the extension of divergence correction sub-program of *AP3DMT* to full fledged 3D DCR modeling and inversion algorithm. Since *AP3DMT* code was in modular form, only the necessary changes were made in forward modeling and Jacobian routines. The forward modeling utilizes finite difference scheme for solving the governing partial differential equation. The source singularity is overcome using the primary/secondary approach [88]. The primary potentials are computed analytically for half space. To solve the system matrix for secondary potential, either a preconditioned iterative solver or a direct solver (depending on the number of sources and size of system matrix) is used. The accuracy of the simulated response is tested using the responses from published papers. The inversion scheme is similar to that used in *AP3DMT*. The inversion is demonstrated with the help of two synthetic examples, one representing a complex model.

Sixth chapter deals with the integration of MT and DCR codes into a joint inversion code. Since, both these methods sense the sub-surface resistivity of earth differently they are employed together to constrain the interpretation. For joint inversion, crucial aspects like model discretization, data scaling, error floor etc. are discussed. The joint inversion results of one synthetic test model are discussed.

Seventh chapter summarizes the work and lists some of the possible future directions of research.

Chapter 2

3D MT FORWARD AND INVERSE MODELING

2.1 Preamble

The Magnetotelluric (MT) techniques belong to the class of geophysical methods that helps in deciphering subsurface conductivity distribution. This conductivity distribution is obtained by performing inversion of the recorded data. Over the decades considerable efforts have been invested in the development of accurate and computationally efficient inversion algorithms. The developed algorithms includes Gauss–Newton (variant of Newton method) schemes [e.g. 89, 141, 161], and direct gradient-based minimization schemes such as non-linear conjugate gradients (*NLCG*) [e.g. 70, 110] or quasi-Newton schemes [e.g. 5, 111].

The Newton method although converges quadratically has not been implemented widely because the computation of second order derivative of the predicted response (Hessian) needed in this method is considered prohibitively expensive [54]. The Gauss–Newton (*GN*) provides a good compromise between computational cost and performance as in this method only the first derivative of the predicted response with respect to model parameter (Jacobian) is required. The convergence rate of the *GN* is far better than the gradient-based methods. In a gradient-based method such as (*NLCG*) the gradient is computed and used to find a new conjugate direction for model update and the step length is estimated using line search methods. As compared to *GN*, *NLCG* required less forward calls per iteration but this efficiency is off-set by larger number of inversion iterations required to converge to a desired level of accuracy. Due to large number of parameters in 3D, explicit formation

of the Jacobian is avoided. For efficiency, in terms of memory and time, it is avoided through Krylov subspace based iterative solver like conjugate gradient (CG) [e.g. 90, 109] or computing product of Jacobian transpose with a vector as required in *NLCG*. However, for computationally efficient inversion algorithm, an accurate forward modeling algorithm is essential as it is used to compute the model responses and is heavily called while computing product of Jacobian with a vector.

Various numerical schemes have been implemented for modeling of MT data. These approaches are based on finite difference [1, 43, 39, 90, 164, 165, 192], finite elements [104, 132] and integral equation [180, 190] schemes. Each of these techniques has certain advantages e.g., integral equation methods are efficient in case of compact model, finite element is better suited for complex model geometry whereas finite difference (FD) is simpler in concept. Irrespective of which technique in employed, these schemes lead to a linear system of equations. The system of equations, generally ill-conditioned, is solved using a preconditioned iterative solver because these require less memory and computation time. Even with the use of an efficient preconditioner like incomplete **LU**, the convergence is very slow especially at low frequencies. This is due to the fact that the divergence of numerically calculated electric field is non-zero. To overcome this issue, Smith [165] proposed a static divergence correction which dramatically improves the convergence of the iterative solver thus reducing the computation time.

In this chapter, I will first briefly discuss the MT theory and the governing partial differential equation. Various aspect of forward modeling like model discretization, boundary conditions, matrix solver, pre-conditioner and static divergence correction are discussed in detail. Next, the formulation of the inverse problem and different inversion schemes are discussed. A detailed description of computations based on the sensitivity matrix (Jacobian) such as its multiplication with a vector is given in this chapter.

2.2 Electromagnetic Theory

The Maxwell's equations describes the propagation of the electromagnetic (EM) fields. The equations describe the relation between time varying electric and magnetic fields. The first

two equation in their differential form are,

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \tag{2.2.1}$$

$$\nabla \times \mathbf{H} = \frac{\partial \mathbf{D}}{\partial t} + \mathbf{J}, \tag{2.2.2}$$

where, **E** is the electric field (V/m), **B** is the magnetic flux density (Wb/m^2), **H** is the magnetic field (A/m), **D** is the electric displacement current (C/m^2), and **J** is the current density (A/m^2). Two more equations can be derived from eq 2.2.1 and eq 2.2.2 by taking the divergence,

$$\nabla \cdot \mathbf{B} = 0, \tag{2.2.3}$$

$$\nabla \cdot \mathbf{D} = q. \tag{2.2.4}$$

The eq 2.2.3 simply denies the existence of magnetic monopoles (Gauss's law for magnetism) and eq 2.2.4 shows that the electric field is the result of the distribution of electric charge (Gauss's law for electricity). The above are the fundamental equations in electromagnetism in five vector fields (**E**, **D**, **B**, **H** and **J**) [191].

The constitutive relations can be used to eliminates three of these and Maxwell's equations can be expressed using two vector fields only. For linear, isotropic media of electric conductivity, σ, magnetic permeability, μ and electric permittivity, ε, three further relationships have been shown to hold,

$$\mathbf{B} = \mu \mathbf{H}, \tag{2.2.5}$$

$$\mathbf{D} = \varepsilon \mathbf{E}, \tag{2.2.6}$$

$$\mathbf{J} = \sigma \mathbf{E}. \tag{2.2.7}$$

Since the variations of the dielectric permittivity ε and the magnetic permeability μ for most rocks are very small in comparison to the variations of the electric conductivity σ, we can use the free-space values for both ε and μ and set $\mu = \mu_o = 1.25566 \times 10^{-6}$ H/m and $\varepsilon = \varepsilon_o = 8.85 \times 10^{-12}$ F/m. Using eq 2.2.5 $-$ 2.2.7 the Maxwell's equations can be rewritten as,

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \tag{2.2.8}$$

$$\nabla \times \mathbf{B} = \mu_0 \varepsilon \frac{\partial \mathbf{E}}{\partial t} + \mu_0 \sigma \mathbf{E}. \tag{2.2.9}$$

2.3 Governing Equation

Substituting eq 2.2.9 in eq 2.2.8 and assuming $e^{i\omega t}$ time dependence and ignoring displacement current we get,

$$\nabla \times \nabla \times \mathbf{E} + i\omega\mu\sigma\mathbf{E} = 0, \tag{2.3.1}$$

where $i = \sqrt{-1}$. Alternatively, the problem can also be formulated for magnetic fields as,

$$\nabla \times \rho \nabla \times \mathbf{H} + i\omega\mu\mathbf{H} = 0, \tag{2.3.2}$$

where ρ is the electrical resistivity. Siripunvaraporn et al. [158] showed that at fine grid, the EM problem can be solved either using vector partial differential equation (PDE) for **E** (eq 2.3.1) or **H** (eq 2.3.2) with same level of numerical accuracy. However, for a coarser grid, solutions obtained from the formulation in **E** are more accurate. Therefore, in this study eq 2.3.1 is solved for modeling of MT data.

2.4 Forward Modeling Scheme

This section describes how the numerical solution of the governing equation eq 2.3.1 are computed.

2.4.1 Finite Difference Formulation

The electric field **E** are simulated by solving the vector Helmholtz equation (eq 2.3.1) using FD method. The first step is discretization of model using grid lines parallel to x, y, and z co-ordinate axes. For more accurate solution eq 2.3.1 is approximated on a staggered-grid [201], as shown in Fig. 2.1. Since the PDE is formulated in terms of electric field components (E_x, E_y and E_z), these are defined on cell edges and the magnetic field components (H_x, H_y and H_z) are defined on center of the cell faces. The discrete finite difference equations using staggered grid are given in Appendix A. After discretization, each cell is assigned a constant conductivity value. To accommodate this change in conductivity at the interface

of a cell, volume weighted average of the conductivity is computed and assigned to each interface. Now, the forward problem is solved using appropriate boundary condition. At the boundary faces the field is assumed to be due to 2D resistivity distributions. To simulate this condition, several sparsely spaced grid lines are added on all side of the modeling domain. The collection of all the discrete FD equations for all nodes, for a frequency, f, results in a matrix equation represented as,

$$\mathbf{A_m e} = \mathbf{s}, \tag{2.4.1}$$

where $\mathbf{A_m}$ is a frequency dependent $N_e \times N_e$ sparse symmetric complex matrix with 13 non-zero elements per row; $\mathbf{s}$ is the N_e dimensional source vector and $\mathbf{e}$ is the N_e dimensional vector representing electric fields at the N_e internal nodes. The system matrix is transformed to a symmetric form by pre-multiplying it with a diagonal matrix whose elements depend on cell volume [43].

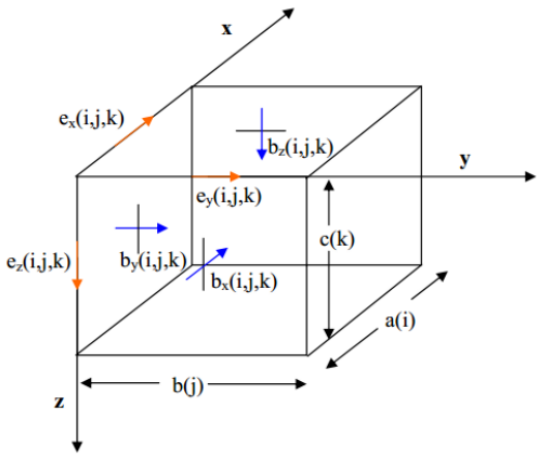

Figure 2.1: Staggered finite difference grid for the 3D MT forward problem. Since the PDE is formulated in terms of electric field components, these are defined on cell edges and the magnetic field components are defined on center of the cell faces.

The three components of the electric field are calculated for a current system which flows in N-S (a mode polarization) or in E-W (b mode polarization) direction. To obtain the same electric field components, however, for another orientation of the current system, we can rotate the model by 90° and form eq 2.4.1 again and solve for $\mathbf{e}$ [192]. The need for the three electric field components using from two orientations of the current system is based on the fact that the model responses in 3D MT are the full impedance tensor and the VTFs. Thus, to construct the full impedance tensor we need two orientations of the current system.

The right hand side vector, **s** in eq 2.4.1 contains the values of the boundary conditions (BCs). For 3D MT forward modeling the boundary values are those which correspond to the all six faces (north, south, east, west, top and bottom) of the model. To formulate BCs at four vertical faces, the conductivity reaches its 2D distribution at these faces and 1D distribution at the edges of these faces. Hence, we need to compute both 1D and 2D solutions and assign the computed values to the grid node at these four vertical faces. For solving the 2D problem at the end faces we have to distinguish between solving the TE- or TM- modes problems depending on the current system which flows in the 3D grid. For example, if the current system flows in E-W direction then we have to solve the 2D TE- mode problem at the easternmost and westernmost faces and TM- mode problem at the northernmost and southernmost faces.

Alternatively, one can use the concept of partitioning of field. In this approach the total electric field, $\mathbf{E}$, is considered as a superposition of primary electric field, $\mathbf{E}_p$, and secondary electric field, $\mathbf{E}_s$, as,

$$\mathbf{E} = \mathbf{E}_p + \mathbf{E}_s. \tag{2.4.2}$$

$\mathbf{E}_p$ is a field simulated for simple background model like half space or layered earth model having conductivity σ_p. If σ_a represents residual conductivity of the medium such that the total conductivity, σ, of medium can be express as,

$$\sigma = \sigma_p + \sigma_a. \tag{2.4.3}$$

Substituting eq 2.4.2 and 2.4.3 into eq 2.3.1 leads to decomposition of eq 2.3.1 into two equations as,

$$\bigtriangledown \times \bigtriangledown \times \mathbf{E}_p + i\omega\mu\sigma_p\mathbf{E}_p = 0, \tag{2.4.4}$$

$$\bigtriangledown \times \bigtriangledown \times \mathbf{E}_s + i\omega\mu\sigma\mathbf{E}_s = -i\omega\mu\sigma_a\mathbf{E}_p. \tag{2.4.5}$$

Now, the forward problem can be solved in two steps. In the first step, eq 2.4.4 is solved to get the value of primary fields, $\mathbf{E}_p$, for background model. Once the primary fields are available, eq 2.4.5 is solved to compute the secondary field, $\mathbf{E}_s$, due to scattered source. The total fields are computed from the primary and secondary fields using eq 2.4.2. In this study we have used the first approach for computing the fields.

2.4.2 Matrix Equation Solver

The system matrix $\mathbf{A}_m$ is a very sparse matrix and can be very large in size, for example, million by million (depending on total number of nodes). The structure of matrix $\mathbf{A}$ for a $4 \times 4 \times 3$ grid is shown in Fig. 2.2. For solving such sparse matrix iterative solvers are better suited as compared to a direct solver due to small memory requirement. Also, in terms of computation time iterative solvers are better. This sparse linear system is solved iteratively using bi-conjugate gradient stabilized (BiCGSTAB) scheme which belongs to a class of Krylov subspace techniques. The system matrix is, in general, ill-conditioned and the convergence rate is very slow. The incomplete $\mathbf{LU}$ decomposition of the diagonal sub-block matrix for preconditioning is quite effective [89]. Still at very low frequency, as in case of MT, converge is very slow. In this case, we need static divergence correction which is discussed in the next section.

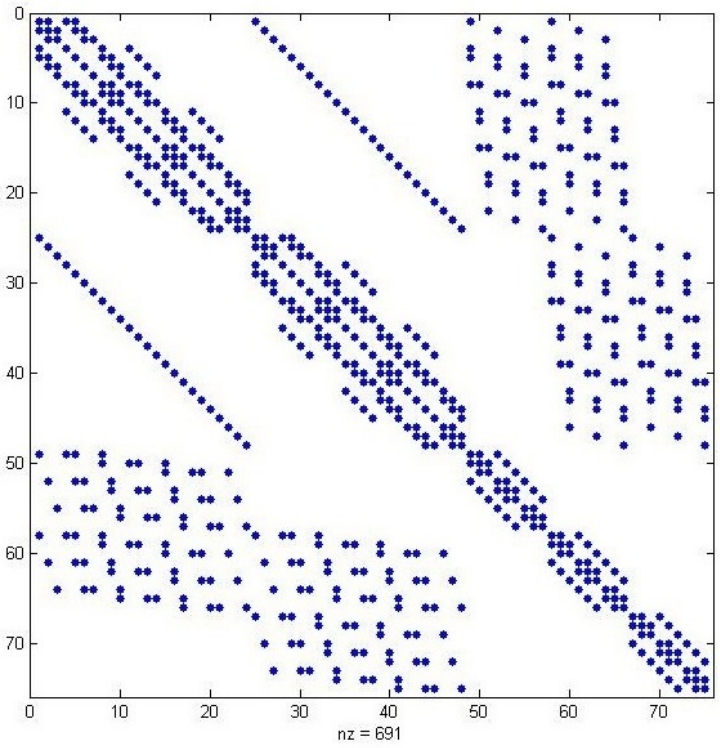

Figure 2.2: Structure of system matrix $\mathbf{A}$ for $4 \times 4 \times 3$ grid lines, where only 691 elements are non-zero.

2.4.3 Static Divergence Correction

Even with the benefits of a staggered grid, in the limit $\omega \longrightarrow 0$, the geometric term $\nabla\times\nabla\times\mathbf{E}$ in eq 2.3.1 dominates over the term $i\omega\mu\sigma\mathbf{E}$ in low conductivity region such as air $\sigma \longrightarrow 0$, making the solution inaccurate. To overcome this issue, a static correction, first proposed by Smith [165], is periodically applied. Mathematically static divergence correction can be derived by taking the divergence of eq. eq 2.3.1 as,

$$\bigtriangledown\cdot(\sigma\mathbf{E}) = 0. \tag{2.4.6}$$

The solution at low frequencies do not satisfy the continuity equation (eq 2.4.6) hence, a correction is made to $\mathbf{E}$. This is achieved by adding the gradient of an unknown scaler, ϕ, to $\mathbf{E}^n$ (the solution at the n^{th} iteration of the BiCGSTAB solver). Now, the task is to estimate ϕ such that the corrected field $\mathbf{E}^c(=\mathbf{E}^n+\bigtriangledown\phi)$ satisfy eq 2.4.6. Using the definition of $\mathbf{E}^c$ and eq 2.4.6, the scaler, ϕ, is calculated from,

$$\bigtriangledown\cdot\sigma(\bigtriangledown\phi) = -\bigtriangledown\cdot(\sigma\mathbf{E}^n). \tag{2.4.7}$$

eq 2.4.7 is Poisson-like equation and solved using FD method. The symmetric system matrix formed in this case is solved using Conjugate Gradient (CG) iterative solver (tolerance 10^{-4} in this study). The divergence correction is applied after a certain number (40 in this study) of BiCGSTAB iterations. The convergence is improved by applying divergence correction and it thereby significantly reduces the computational time needed for solution of eq 2.3.1.

To demonstrate the efficiency achieved through divergence correction, the forward problem is solved for period values of 100 s. The convergence plots of these tests are shown in Fig. 2.3 for both source polarizations. The convergence curve for the case without divergence correction reaches the desired tolerance (set at 10^{-7}) is reached in 942 and 1087 iterations for a and b polarization respectively. In case of divergence correction, the convergence threshold is reached in 140 and 171 iterations for a and b polarization respectively.

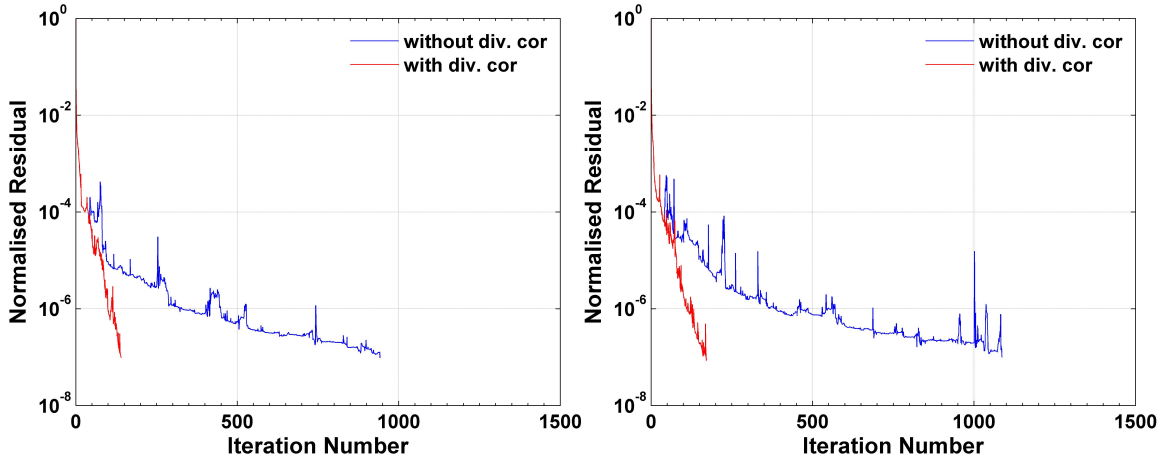

Figure 2.3: Iterative solver convergence plots to show the comparison of with and without divergence correction for (left) *a* polarization, and (right) for *b* polarization for 100 s.

2.4.4 MT Response Computation

Once the electric field are computed, they are interpolated to the observation sites. The magnetic field, $\mathbf{H}$, corresponding to an electric field solution $\mathbf{E}$ can be expressed as $\mathbf{H} = (-i\omega\mu)^{-1}\nabla\times\mathbf{E}$. The computation of $\mathbf{H}$ at the observation sites requires a simple transformation (including interpolation) matrix which is a discrete approximation of scaled curl operator where the scaling factor is reciprocal of $-i\omega\mu$. From these electric and magnetic field the impedance tensor, $\mathbf{Z}$, and vertical magnetic field transfer functions (VTFs) are computed at the observation sites. In the 3D MT case, the impedance is a 2×2 tensor, which requires two independent electric field solutions computed for two different source polarizations [37, 159, 192]. Thus impedance tensor and VTFs can be written as,

$$\begin{pmatrix} Z_{xx} & Z_{xy} \\ Z_{yx} & Z_{yy} \end{pmatrix} = \begin{pmatrix} E_x^a & E_x^b \\ E_y^a & E_y^b \end{pmatrix} \begin{pmatrix} H_x^a & H_x^b \\ H_y^a & H_y^b \end{pmatrix}^{-1}, \tag{2.4.8}$$

$$\begin{pmatrix} T_x \\ T_y \end{pmatrix} = \begin{pmatrix} H_z^a & H_z^b \end{pmatrix} \begin{pmatrix} H_x^a & H_x^b \\ H_y^a & H_y^b \end{pmatrix}^{-1}, \tag{2.4.9}$$

where, indices a and b represents two different source polarizations. The components of the impedance tensor and VTFs [110] can be rewritten as,

$$\begin{aligned} Z_{xx} &= \frac{E_x^a H_y^b - E_x^b H_y^a}{D}, \qquad Z_{xy} = \frac{E_x^b H_x^a - E_x^a H_x^b}{D}, \\ Z_{yx} &= \frac{E_y^a H_y^b - E_y^b H_y^a}{D}, \qquad Z_{yy} = \frac{E_y^b H_x^a - E_y^a H_x^b}{D}, \end{aligned} \tag{2.4.10}$$

$$T_x = \frac{H_z^a H_y^b - H_z^b H_y^a}{D}, \qquad T_y = \frac{H_z^b H_x^a - H_z^a H_x^b}{D}, \tag{2.4.11}$$

where, $D = E_x^a H_y^b - E_x^b H_y^a$. Once impedance tensor and VTFs are computed, other derived responses can be easily computed.

2.5 Inversion

All the inversion algorithms aim at finding a meaningful model $\tilde{\mathbf{m}}$, a M- dimensional model parameter vector, while fitting the data $\tilde{\mathbf{d}}^{obs}$ of dimension N_d to an acceptable level in a stable manner. We consider minimization of the penalty functional defined as,

$$\Phi(\tilde{\mathbf{m}}, \tilde{\mathbf{d}}^{obs}) = (\mathbf{d}^{obs} - \tilde{F}(\tilde{\mathbf{m}}))^T \mathbf{C}_\mathbf{d}^{-1} (\mathbf{d}^{obs} - \tilde{F}(\tilde{\mathbf{m}})) + \lambda (\tilde{\mathbf{m}} - \mathbf{m_0})^T \mathbf{C}_\mathbf{m}^{-1} (\tilde{\mathbf{m}} - \mathbf{m_0}), \tag{2.5.1}$$

where $\tilde{F}(\tilde{\mathbf{m}})$ is the forward mapping, $\mathbf{C_d}$ is the data covariance matrix, $\mathbf{m_0}$ is the apriori model, $\mathbf{C_m}$ is the model covariance matrix or regularization term and λ is the trade-off parameter. $\mathbf{C_d}$ is generally diagonal hence, it can be eliminated from definition of penalty functional by simply rescaling of data and forward mapping. Both $\mathbf{m_0}$ and $\mathbf{C_m}$ can also be eliminated from eq 2.5.1 by setting $\mathbf{m} = \mathbf{C_m}^{-1/2}(\tilde{\mathbf{m}} - \mathbf{m_0})$. This transformation reduces eq 2.5.1 to,

$$\phi(\mathbf{m}, \mathbf{d}^{obs}) = (\mathbf{d}^{obs} - F(\mathbf{m}))^T (\mathbf{d}^{obs} - F(\mathbf{m})) + \lambda \mathbf{m}^T \mathbf{m}, \tag{2.5.2}$$

where, $F(\mathbf{m}) = \tilde{F}(\mathbf{C_m}^{1/2}\mathbf{m} + \mathbf{m_0})$. After minimizing eq 2.5.2 in transform domain the model parameters are transformed back into the space of the original model parameter $\tilde{\mathbf{m}} = \mathbf{C_m}^{1/2}\mathbf{m} + \mathbf{m_0}$. For details one can refer to Kelbert et al. [70]. In this study, we have implemented both quasi-linear inversion and non-linear inversion using conjugate gradient for the minimization of eq 2.5.2.

2.5.1 Quasi-linear Inversion

In this approach, the objective functional is first approximated by a Taylor series expansion. The quadratic approximation of the objective functional is then minimized to produce a series of the updated models iteratively. In the Gauss Newton (*GN*) method only the first derivative in the Hessian matrix of Newton's method is retained but the second-order derivative is discarded. This leads to an iterative sequence of approximate solutions as,

$$(\mathbf{J}^T\mathbf{J}+\lambda\mathbf{I})\delta\mathbf{m}_n = \mathbf{J}^T\mathbf{r}-\lambda\mathbf{m}_n, \tag{2.5.3}$$

where $\mathbf{m}_n$ are the model parameters at the n^{th} iteration, $\mathbf{J}$ is Jacobian and $\mathbf{r}=\mathbf{d}^{obs}-F(\mathbf{m}_n)$ is the residual. eq 2.5.3 is solved for $\delta\mathbf{m}_n$ and the new updated model parameter vector is obtained as $\mathbf{m}_{n+1}=\mathbf{m}_n+\delta\mathbf{m}_n$. For stability this linearized scheme generally requires some form of step length damping [96, 135]. Alternatively, instead of solving for $\delta\mathbf{m}_n$ one can solve for $\mathbf{m}_{n+1}$ using Occam approach [25, 120]. In Occam's algorithm eq 2.5.3 is written as,

$$(\mathbf{J}^T\mathbf{J}+\lambda\mathbf{I})\mathbf{m}_{n+1} = \mathbf{J}^T\hat{\mathbf{d}}, \tag{2.5.4}$$

where $\hat{\mathbf{d}}=\mathbf{d}^{obs}-F(\mathbf{m})+\mathbf{J}\mathbf{m}_n$. In data space [155, 159] the solution of eq 2.5.4 is written as,

$$\mathbf{m}_{n+1}=\mathbf{J}^T\mathbf{b}_n; \qquad (\mathbf{J}\mathbf{J}^T+\lambda\mathbf{I})\mathbf{b}_n=\hat{\mathbf{d}}. \tag{2.5.5}$$

To avoid explicit computation and storage of $\mathbf{J}$, eq 2.5.4 and eq 2.5.5 are solved with a memory efficient Krylov subspace iterative solver such as conjugate gradients (*CG*). In this approach, the product of matrix and an arbitrary vector such as $(\mathbf{J}^T\mathbf{J}+\lambda\mathbf{I})\mathbf{m}$ is computed and this can be performed at the cost of just two forward problems. Following Newman and Alumbaugh [109], at the n^{th} inversion iteration, the regularization parameter λ for *GN* is determined as $\lambda = r_{sum}/2^{n-1}$ where, r_{sum} is the largest row sum of $real(\mathbf{J}^T\mathbf{J})$.

2.5.2 Non-linear inversion using conjugate gradient

In this approach, eq 2.5.2 is directly minimized using a gradient based optimization technique like non-linear conjugate gradient (*NLCG*) [71, 111, 135]. Here, the gradient of eq 2.5.2 with

respect to the variation in model parameter, $\mathbf{m}$, is computed as,

$$\left.\frac{\partial \Phi}{\partial \mathbf{m}}\right|_{\mathbf{m}_n} = -2\mathbf{J}^T\mathbf{r} + 2\lambda\mathbf{m}_n, \tag{2.5.6}$$

and it is used to a compute new 'conjugate' search direction. The 'line search' is used to minimize the penalty functional along this direction and it requires solving forward problem few times and the gradient is recomputed. Basic computational steps for *NLCG* include solving forward problem for model parameter, $\mathbf{m}_n$, and multiplication of $\mathbf{J}^T$ by the residual, $\mathbf{r}$. However, for the regularization parameter approach described above, this scheme does not work because varying the regularization parameter would compromise the orthogonality of search directions [38]. Following Kelbert et al. [70] *NLCG* iterations are performed for fixed value of λ and when misfit stalls i.e. difference between misfits of two previous iterations is less than a predefined threshold, λ is reduced by a predetermined factor (10 in present study).

The schemes for minimizing eq 2.5.2, as discussed above, are expressed in terms of data and model parameter vectors $\mathbf{d}^{obs}$ and $\mathbf{m}$, forward mapping $F(\mathbf{m})$, Jacobian $\mathbf{J}$ (or $\mathbf{J}^T$), data and model covariance matrices $\mathbf{C_d}$ and $\mathbf{C_m}$. The modular implementation of any inversion algorithm can be readily carried out using these components. In the next section we briefly discuss and provide formulation of Jacobian $\mathbf{J}$ in terms of the forward solver, model parameterization and the numerical simulation of the necessary observation operator as implemented in a MATLAB based code, named as *AP3DMT* in this thesis.

2.5.3 Data Sensitivities

The computation of predicted data $\mathbf{d}_f$ (forward mapping $F_f(\mathbf{m})$), for a single frequency f, involves two steps; (*i*) solving the vector Helmholtz equation for electric field $\mathbf{E}$ and, (*ii*) computing the predicted data like impedance tensor at observation sites using these fields. After using FD approximation, the PDE (eq 2.3.1) is transformed into the matrix equation eq 2.4.1. The predicted data set is obtained at the observation points via,

$$\mathbf{d}_f^j = F_f^j(\mathbf{m}) = \psi_j(\mathbf{e}(\mathbf{m}), \mathbf{m}), \qquad j = 1, \ldots, nobs. \tag{2.5.7}$$

where, ψ_j is generally a non-linear function of $\mathbf{e}$ (and possibly $\mathbf{m}$). Using chain rule, the expression for the Jacobian matrix of the forward mapping $F_f(\mathbf{m})$, is given as,

$$\mathbf{J}_f^{ik} = \partial_k F_f^i(\mathbf{m}) = \sum_j \partial_j \psi^i(\mathbf{e})\ \partial_k \mathbf{e}^j(\mathbf{m}) + \partial_k \psi^i(\mathbf{m}), \tag{2.5.8}$$

where, ∂_k signifies partial differentiation with respect to the k^{th} argument of a function [reading $\psi(\mathbf{m})$ as $\psi(m^1, m^2, ..., m^M)$]. Let $\mathbf{G}_f$, $\mathbf{L}_f$ and $\mathbf{Q}_f$, the partial derivative matrices, for the frequency f, be defined as,

$$\mathbf{L}_f^{ij} = \partial_j \psi^i(\mathbf{e})\big|_{\mathbf{e}_n, \mathbf{m}_n} \quad \mathbf{G}_f^{jk} = \partial_k \mathbf{e}^j(\mathbf{m})\big|_{\mathbf{m}_n}, \quad \mathbf{Q}_f^{ik} = \partial_k \psi^i(\mathbf{m})\big|_{\mathbf{e}_n, \mathbf{m}_n}. \tag{2.5.9}$$

here, $\mathbf{e}_n$ is the solution of eq 2.4.1 for model parameter $\mathbf{m}_n$. Hence, the Jacobian at $\mathbf{m}_n$ can be written in matrix notation as,

$$\mathbf{J}_f = \mathbf{L}_f \mathbf{G}_f + \mathbf{Q}_f. \tag{2.5.10}$$

Now, differentiating eq 2.4.1 with respect to model parameter we obtain,

$$\mathbf{A}_{\mathbf{m},f} \left[\frac{\partial \mathbf{e}}{\partial \mathbf{m}}\bigg|_{\mathbf{m}=\mathbf{m}_n} \right] = - \frac{\partial}{\partial \mathbf{m}} (\mathbf{A}_{\mathbf{m},f} \mathbf{e}_n)\bigg|_{\mathbf{m}_n}, \tag{2.5.11}$$

or,

$$\mathbf{A}_{\mathbf{m},f} \mathbf{G}_f = \mathbf{P}_f. \tag{2.5.12}$$

Here $\mathbf{e}_n$ is considered independent from the parameter vector $\mathbf{m}$. Combining eq 2.5.10 and eq 2.5.12 and noting the fact that for 3D MT data $\mathbf{Q}_f = 0$, we get Jacobian in a matrix form as,

$$\mathbf{J}_f = \mathbf{L}_f \mathbf{A}_{\mathbf{m},f}^{-1} \mathbf{P}_f, \tag{2.5.13}$$

where, (*i*) the solver sensitivity $\mathbf{P}_f$, defines the product of sensitivity of the coefficient matrix (with respect to the model parameter) and electric field; (*ii*) the forward solver, $\mathbf{A}_{\mathbf{m},f}^{-1}$ (maps $\mathbf{P}_f$ to $\frac{\partial \mathbf{e}}{\partial \mathbf{m}}$); (*iii*) sensitivity functional, $\mathbf{L}_f$ (maps perturbation in the EM solution to perturbation in the predicted data i.e. $\frac{\partial \mathbf{e}}{\partial \mathbf{m}}$ to $\frac{\partial \mathbf{d}_f^{pred}}{\partial \mathbf{m}}$).

To evaluate the impedance tensor and the VTFs, the EM field solutions for two source polarizations are required ($\mathbf{e} = (\mathbf{e}^a, \mathbf{e}^b)$). Thus, the rows of Jacobian are formed from the components of these two source polarizations.

The columns of solver sensitivity $\mathbf{P}_f^a$, for single frequency index, f, and for the a polarization, are defined as $\alpha \mathbf{A}'_{m_i,f} \mathbf{e}^a$ for *(i = 1, ... , M)*. Here, $\mathbf{A}'_{m_i,f}$ is the $N_e \times N_e$ diagonal matrix obtained after the differentiation of the coefficient matrix with respect to model

parameter m_i with each element scaled by $(-i\omega\mu)^{-1}$, α being the scalar $-i\omega\mu$, $\mathbf{e}^a$ is the electric field for a polarization. Performing such operation for all the model parameters results in a $N_e \times M$ matrix. However, columns of $\mathbf{P}_f^a$ can also be written as $\alpha\bar{\mathbf{E}}\mathbf{A}'_{m_i,f}$ where, $\bar{\mathbf{E}}$ is the $N_e \times N_e$ diagonal matrix and $\mathbf{A}'_{m_i,f}$ is now a column vector of length N_e. Hence, $\mathbf{P}_f^a$ can be written as $\alpha\bar{\mathbf{E}}\mathbf{A}'_{m,f}$ with $\mathbf{A}'_{m,f}$ being a $N_e \times M$ matrix. Thus $\mathbf{P}_f$ can be written for both the polarizations as,

$$\mathbf{P}_f = \alpha \begin{bmatrix} \bar{\mathbf{E}}^a & 0 \\ 0 & \bar{\mathbf{E}}^b \end{bmatrix} \begin{bmatrix} \bar{\mathbf{P}} \\ \bar{\mathbf{P}} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_f^a \\ \mathbf{P}_f^b \end{bmatrix}. \tag{2.5.14}$$

The components $\bar{\mathbf{E}}^a$ and $\bar{\mathbf{E}}^b$ are diagonal matrices with electric fields corresponding to the two source polarizations, a and b, as their elements and $\bar{\mathbf{P}}$ represents $\mathbf{A}'_{m,f}$. The elements of real and frequency independent solver sensitivity $\bar{\mathbf{P}}$ depend on the numerical grid and model parameter and these are inexpensive to compute.

The matrix $\mathbf{L}_f$ is defined as,

$$\mathbf{L}_f = \begin{bmatrix} \mathbf{L}_f^a & \mathbf{L}_f^b \end{bmatrix}. \tag{2.5.15}$$

A row of matrix $\mathbf{L}_f$ has two blocks, each of length N_e, which multiply to the perturbations in electric field for the two polarizations and sums together to yield the total perturbation in the data. The detailed expression of $\mathbf{L}_f$ for impedance tensor and VTFs and other derived responses are given in Appendix B. Thus, $\mathbf{J}_f$ can be expressed in the matrix form as,

$$\mathbf{J}_f = \begin{bmatrix} \mathbf{L}_f^a & \mathbf{L}_f^b \end{bmatrix} \begin{bmatrix} \mathbf{A}_{\mathbf{m},f}^{-1} & 0 \\ 0 & \mathbf{A}_{\mathbf{m},f}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{P}_f^a \\ \mathbf{P}_f^a \end{bmatrix} = \mathbf{L}_f^a \mathbf{A}_{\mathbf{m},f}^{-1} \mathbf{P}_f^a + \mathbf{L}_f^b \mathbf{A}_{\mathbf{m},f}^{-1} \mathbf{P}_f^b. \tag{2.5.16}$$

Generally, the data set is obtained for a range of frequencies. Thus, both the vectors, observed and predicted, can be decomposed into $(f = 1, \ldots, N_f)$ independent blocks, corresponding to each frequency as,

$$\mathbf{d} = \begin{bmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_{N_f} \end{bmatrix} = \begin{bmatrix} F_1(\mathbf{m}) \\ \vdots \\ F_{N_f}(\mathbf{m}) \end{bmatrix}; \qquad \mathbf{d}^{obs} = \begin{bmatrix} \mathbf{d}_1^{obs} \\ \vdots \\ \mathbf{d}_{N_f}^{obs} \end{bmatrix}. \tag{2.5.17}$$

Similarly, the full Jacobain $\mathbf{J}$ can be decomposed using its components as,

$$\mathbf{J} = \begin{bmatrix} \mathbf{L}_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{L}_{N_f} \end{bmatrix} \begin{bmatrix} \mathbf{A}^{-1}_{\mathbf{m},1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{A}^{-1}_{\mathbf{m},N_f} \end{bmatrix} \begin{bmatrix} \mathbf{P}_1 \\ \vdots \\ \mathbf{P}_{N_f} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_1\mathbf{A}^{-1}_{\mathbf{m},1}\mathbf{P}_1 \\ \vdots \\ \mathbf{L}_{N_f}\mathbf{A}^{-1}_{\mathbf{m},N_f}\mathbf{P}_{N_f} \end{bmatrix}, \tag{2.5.18}$$

and its transpose as,

$$\mathbf{J}^T = \begin{bmatrix} \mathbf{P}_1^T\mathbf{A}^{-1}_{\mathbf{m},1}\mathbf{L}_1^T & \dots & \mathbf{P}_{N_f}^T\mathbf{A}^{-1}_{\mathbf{m},N_f}\mathbf{L}_{N_f}^T \end{bmatrix}, \tag{2.5.19}$$

here, $\mathbf{A}^T_{\mathbf{m},f} = \mathbf{A}_{\mathbf{m},f}$ (symmetric matrix).

Further, the model parameter $\mathbf{m}$ is real but all the computations are efficiently performed using complex arithmetic. Data may be real (e.g., phase) or complex (e.g., impedance tensor, VTFs). For the penalty functional eq 2.5.2, we have assumed that the data set is real, that is, both real and imaginary parts of a complex observation are treated as independent elements of the data set $\mathbf{d}$, and the operator $\mathbf{J}$ has been recast as real mapping from model parameter to data vector (refer Egbert and Kelbert [39]).

2.6 Summary

A 3D forward modeling algorithm for MT data is developed. The electric fields are simulated on staggered grid using finite difference scheme. For efficiency, in terms of memory and computation time, the matrix equation is solved using iterative solver with incomplete $\mathbf{LU}$ decomposition of the diagonal sub-block matrix as preconditioner. The convergence of iterative solver is further improved by using the static divergence correction.

The basic theory of inversion is discussed. The detailed expressions for data sensitivities for multi-frequency, multi-component data are presented. Further, representation of Jacobian matrix in block form using its three components for easy implementation is discussed.

Chapter 3

OVERVIEW OF 3D MT CODE : AP3DMT

3.1 Preamble

Several 3D electromagnetic data inversion algorithms were developed and implemented in various computer codes during the first decade (notably, [5, 55, 110, 111, 157, 206], etc.). Most of these codes were focused on inversion with a particular computational approach. Recently, Egbert and Kelbert [39] and Kelbert et al. [71] presented a Fortran code, *ModEM*, with its emphasis on modular implementation of the basic components of inversion - forward modeling, model parametrization and regularization, data functionals, sensitivity computation and inversion algorithms which are reusable and readily extensible. This and most of the previous codes are written in FORTRAN programming language. Many of these algorithms permit interchangeability and re-usability of various sub-programs thereby providing a base code for further development of new inversion schemes. However, it is difficult for a new researcher to make desired changes for experimentation or further development. MATLAB provides a solution to this problem. It has a powerful computation environment with extensive numerical libraries; data visualization capabilities, and easier high-level programming language grammar, resulting from simple vector and matrix multiplication based on arrays that need no memory dimensioning. Also, it provides a mixed language environment, which is useful for bottleneck computations that do not run fast enough in MATLAB.

In 2014 when we initiated present thesis work [1] only 2D MT codes [81] were available in MATLAB. Keeping the advantages of MATLAB in mind we have developed a versatile

[1] Before working on 3D, the author has developed 2D block inversion algorithm [150] for MT data in MATLAB as a part of master's dissertation (July, 2013 to July, 2014).

3D code which can be used for inversion of various combinations of MT data jointly or individually. The code is named as *AP3DMT* in the thesis and is published [151]. *AP3DMT* provides a basic framework for 3D modeling and inversion of MT data which is flexible for further development and experimentation with different schemes of inversion, parametrization and regularization. This code will provide a basic framework to researchers who are conversant with MATLAB and willing to use or develop its capabilities for future applications.

In the present chapter, the main focus is on program structure with some basic mathematical formulae required for implementation. The main body of the code is divided into two parts. The first part includes grid generator and I/O sub-programs while the second part deals with modeling and inversion. An overview of the program including grid generator for common models with cuboids and/or polyhedron shaped target bodies and forward and inverse modeling is discussed. The main feature of grid generator is its robustness in handling complex geological features needed to simulate responses for complex 3D structures. Such an efficient grid generator is useful when performing block inversion [150]. Later detailed description and implementation of forward modeling, inversion and sensitivity computations, along with a coarse grained parallelization over frequency is discussed. *AP3DMT* code can be easily modified to accommodate modifications in the forward problem (e.g., different forward solver, different types of responses, model parameters, etc.) without any modifications in other sub-programs like, inversion sub-program. Finally, I will demonstrate the code versatility for grid generation.

3.2 Code Structure

The *AP3DMT* program is developed over last three and half year. The first version of this code was tested in January 2016. Since then, new features have been added (to be discussed in the chapter 5 and 6). In the current stage it has 2 scripts [2] and 114 functions [3]. Out of these 114 functions, 8 were taken/adapted from other sources. Apart from these scripts and functions, additional scripts are also written for conversion of model and data to different formats (e.g., edi files to our format) and for plotting of model and data. Structurally, the *AP3DMT* program is divided into two independent parts as shown in Fig. 3.1. The first part

[2] equivalent to program in FORTRAN
[3] equivalent to subroutines in FORTRAN

consists of 1 script and 28 functions and creates control file and other necessary files (e.g., model file etc.). This part also includes a graphic user interface (GUI) and grid generator. The second part with 1 script and 86 functions, uses previously generated files and performs the modeling and/or inversion computations. Details of these two independent parts follows.

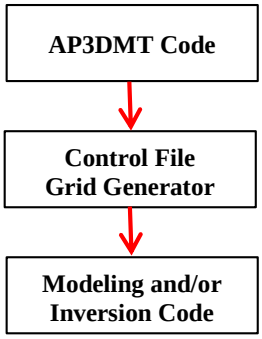

Figure 3.1: Schematic representation of *AP3DMT* showing two independent parts, file making and modeling/inversion part.

3.3 Preparation of Files

This code generates four files to be subsequently used in modeling/inversion code. Fig. 3.2 outlines the various steps used for file preparation and grid generation. The program starts by running the main script, *ProgramRun* which invokes the GUI (Fig. 3.3). In the GUI user has to fill the necessary details like, dimension, type of grid (manual or generated), data units, field variation, frequencies, solver details, inversion parameters, etc. The user must provide an input model file (**.xls* file) which contains observation sites co-ordinates, manual grids (if any) and model description (synthetic and/or initial guess) on different sheets. In this file, the model is described in terms of layers, target bodies and modeling domain. The cuboid shaped target bodies are defined using seven parameters (six for location and one for resistivity) whereas, the polyhedron shaped bodies are defined using three times the total number of vertices for location description and one parameter for resistivity. The program uses information from GUI and the **.xls* file and generates following four files,

(i) parameter file,
(ii) data file (data corresponding to all data types and for all stations and periods),
(iii) model file (synthetic and/or apriori model), and
(iv) covariance file (optional).

The parameter file contains the information about forward solver (e.g., tolerance, number of iterations, number of divergence correction), inversion parameter (e.g., inversion iterations, misfit tolerance, inversion domain) etc. The data file has data corresponding to all data types and for all stations and periods. Information about field variations, units of data are written in the header. In the model (synthetic and/or initial guess) files, grid information and resistivity of each cell is written either in linear or log scale. The model covariance file allows the user to turn off the smoothening across the domain boundaries, and also allows the model parameters within a domain to be frozen (e.g., for water bodies resistivity, air resistivity in case of topography, etc.).

In this code, the main component is a grid generation program which efficiently handles arbitrarily shaped target bodies along with topography (or bathymetry). The detailed description of grid generation for some selected target bodies (varying shapes) is given in the next section.

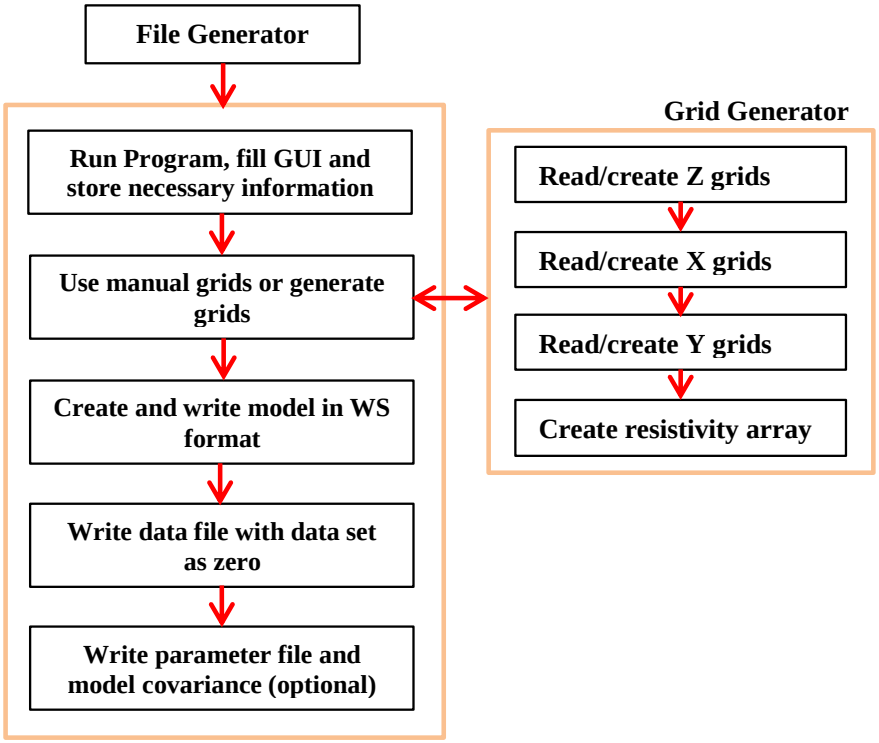

Figure 3.2: Schematic representation of file making code with grid generation.

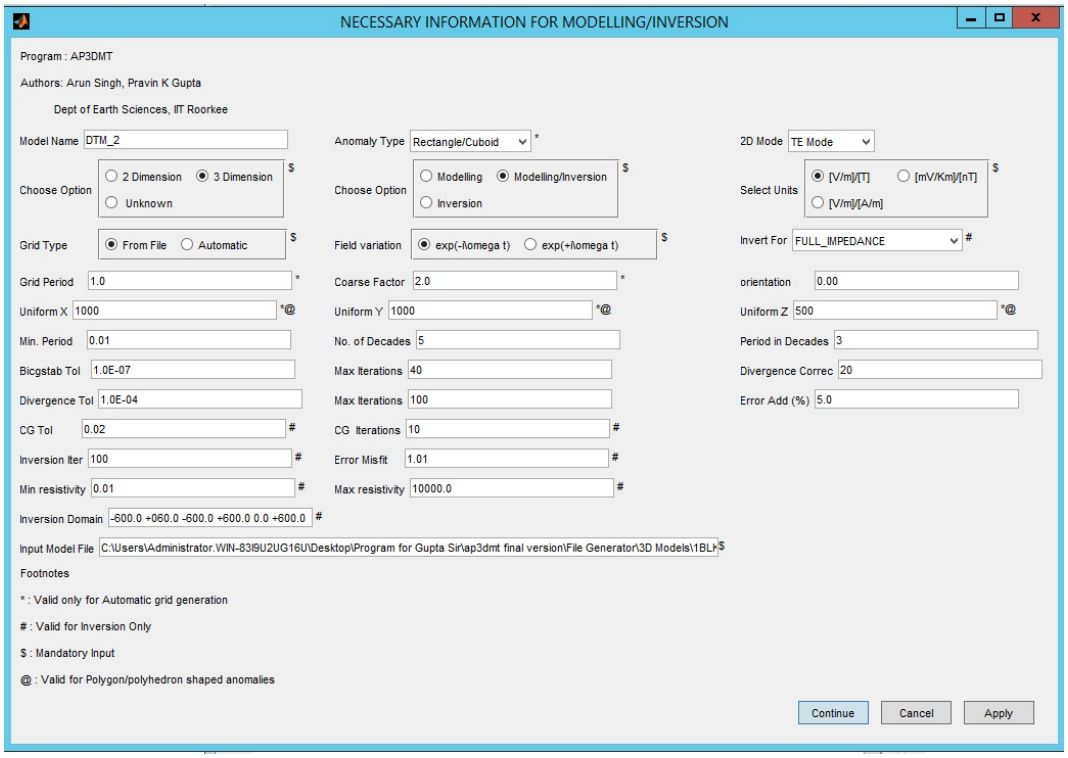

Figure 3.3: Figure showing Graphic User Interface (GUI) for providing user input.

3.4 Grid Generation Code

The grid generation code is used for model discretization. The code efficiently handles both cuboid and arbitrarily shaped target bodies (represented by polyhedrons). Table 3.1 gives the description of the major functions used in this code. The discretization for cuboid and arbitrarily shaped target bodies is discussed separately.

Table 3.1: Main functions used in *AP3DMT* for preparation of files and grid generation. (* downloaded from MATLAB file exchange)

Functions	**Description**
*DialogBox**	Calls GUI and reads filled information [72]
*inputsdlg**	Enhanced input dialog box supporting multiple data types [72]
NormalBodies3D	Grid making for model with cuboids shaped target bodies
ArbitraryBodies3D	Grid making for model with polyhedron shaped target bodies
ZGrids	Generate grids in *z*- direction
YXGrids	Generate grids in *x*- and *y*- direction
*gridfit**	Models a surface on a 2D grid from scattered or semi-scattered z(x,y) [30]
*inhull**	Tests if a set of points are inside a convex hull [31]
ResistivityArray	Construct 3D resistivity array
WriteModel	Write Model in a .dat file
WriteData	Write Data in a .dat file

3.4.1 Cuboid Bodies

For cuboid shaped target bodies, the grids are generated on the basis of skin depths across an interface. First, the domain of modeling is divided into regions on the basis of target bodies and a resistivity matrix is constructed in *ResistivityArray*. For example, let us consider two blocks of resistivities ρ_a and ρ_b (1 and 100 Ω-m respectively with each of dimensions $40 \times 20 \times 10$ km) placed adjacent to each other and in a half space of resistivity ρ_h (10 Ω-m) (Fig. 3.4a). At an interface (say vertical interface number *'2'*, Fig. 3.4b), the minimum skin depth, δ, among the two different regions, is computed (columns between interface *1-2* and interface *2-3*) and then two grid lines are generated on either side of the interface with spacing equal to $\delta/4$ and then $\delta/3$ and so on (task performed in *YXGrids*). This approach ensures fine grids near the conductivity contrast to obtain accurate results in these areas. The coarseness of the grid, increases as one moves away from the interface (Fig. 3.5a). The coarseness of the grid can be easily increased by preserving the overall pattern but changing period used for skin depth computation. Once the grids are generated, final resistivity matrix is created (in *ResistivityArray*) where each element represents a discretized cell of constant resistivity. Alternatively, manual grids can also be provided in the input file but one must ensure that a grid line passes through each interface for true representation of the desired anomaly. This feature makes the code more user friendly as the user can assign desired grids as per the requirement.

3.4.2 Polyhedron Bodies

To represent the complex geological features and structures, polyhedrons are best suited. For example, consider a body of resistivity ρ_a (5 Ω-m with dimensions $80 \times 28 \times 16$ km with top at 4 km depth), with its major axis making an angle of 45° with respect to vertical line, embedded in half space a of resistivity ρ_h (100 Ω-m) (Fig. 3.4c). This polyhedron body (oblique block) is initially approximated by a cuboid *abcd* (length and breadth along cartesian coordinate system) which encapsulates the whole target body (Fig. 3.4c) and a resistivity matrix is created. Uniform grids of specified size (2 km) are generated for this region as a stair case model which is better suited for such bodies. In other regions of the model, the grids are generated on the basis of skin depth criteria, as discussed above (Fig. 3.5b). After grids generation, construction of the resistivity matrix begins. All the cell centers of the cuboid (encapsulating all the polyhedron bodies) are checked using the MATLAB function, *inhull*, which tests whether a point is inside a convex hull or not. If the cell center lies inside a convex hull then the cell is assigned of the target body resistivity else that of the background.

3.4.3 Topography

To incorporate topography (and/or bathymetry), we first model, in the form of z(x,y), a surface from the scattered or semi-scattered data using the MATLAB function, *gridfit*. Although *gridfit* is not an interpolant, it builds a surface over a complete lattice (models grids in xy plane), extrapolating smoothly into the corners. Once the surface has been modeled, a uniform grid in the vertical direction is generated till the maximum elevation (depression) is reached. Later, based on these surface values the resistivity matrix is modified.

To demonstrate the robustness of grid generator, we demonstrate the construction of resistivity matrix, a spiral like conductor of 10 Ω-m embedded in a half-space of resistivity 100 Ω-m, is considered. The spiral is of constant diameter, 10 km, with its center at origin (coordinate system). The spiral is always confined between -30 km to $+30$ km in both x- and y- directions (Fig. 3.6). Uniform grids with 1.5 km spacing in $x-$ and $y-$ directions were made. Fig. 3.7 represents the discretized model with this spiral like conductor on xy planes at different depths planes.

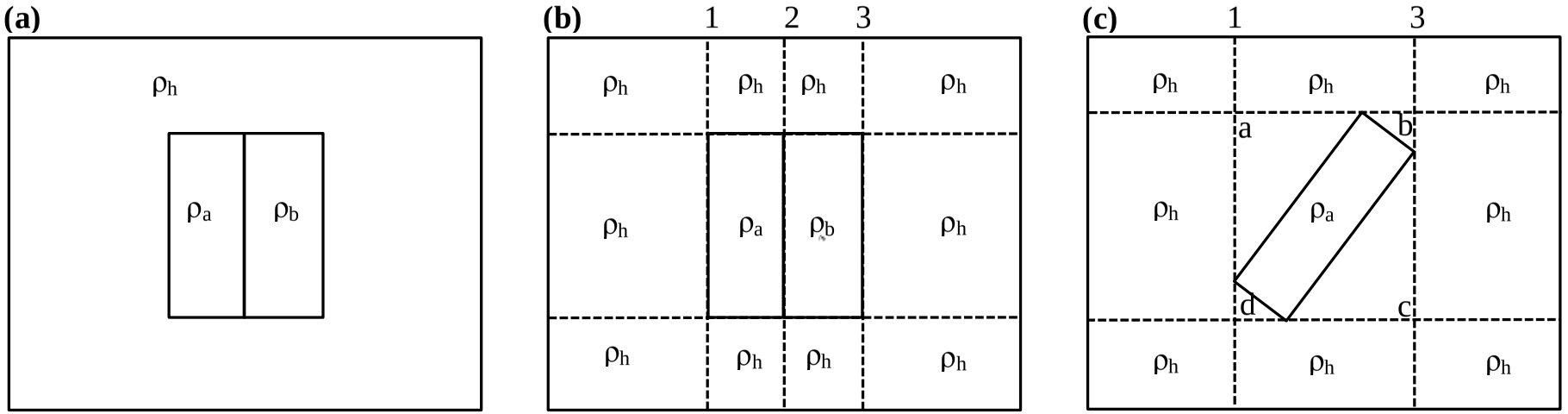

Figure 3.4: (a) Top view of a model with 2 blocks of resistivity ρ_a and ρ_b Ω-m placed adjacent to each other and embedded in ρ_h Ω-m half space. (b) Model is discretized into regions on the basis of target bodies. (c) Top view of a model with ρ_a Ω-m body, major axis rotated by 45° clockwise, embedded in ρ_h Ω-m half space.

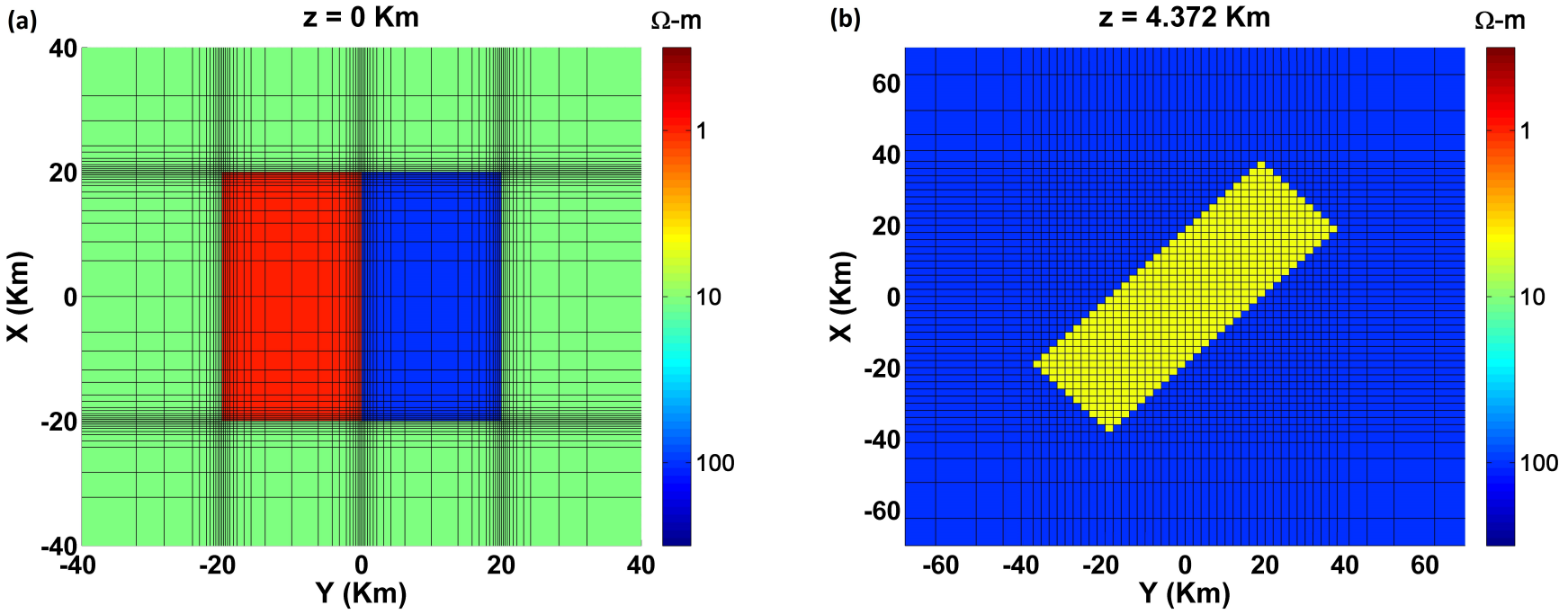

Figure 3.5: Schematic diagram showing top view of the (a) 3D model with conductive (1 Ω-m) and resistive (100 Ω-m) prisms buried in a layered Earth [210] with grids at z=0 km. Period of 1 s was used for computing the skin depth. (b) 5 Ω-m body, major axis rotated clockwise by 45°, embedded in 100 Ω-m half space with grids at z=4.37 km.

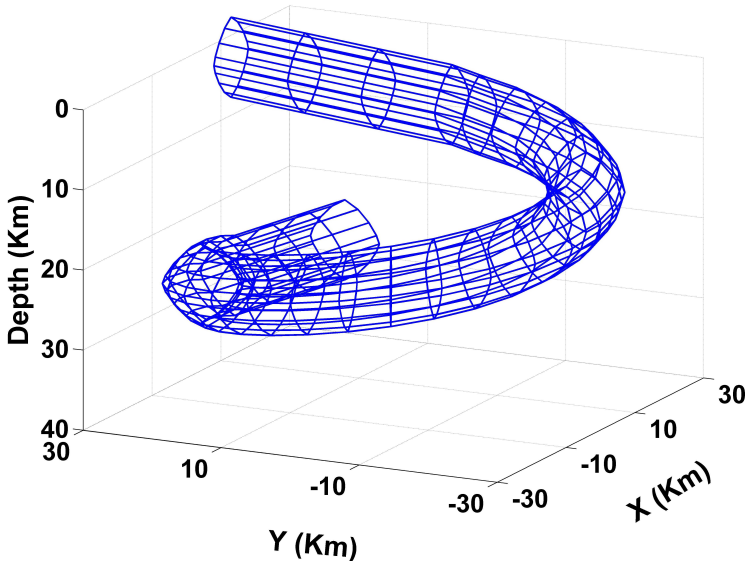

Figure 3.6: Mesh model of a spiral conductor (10 Ω-m) embedded in a half space (100 Ω-m). The spiral is always confined between -30 km to $+30$ km in both x- and y- directions.

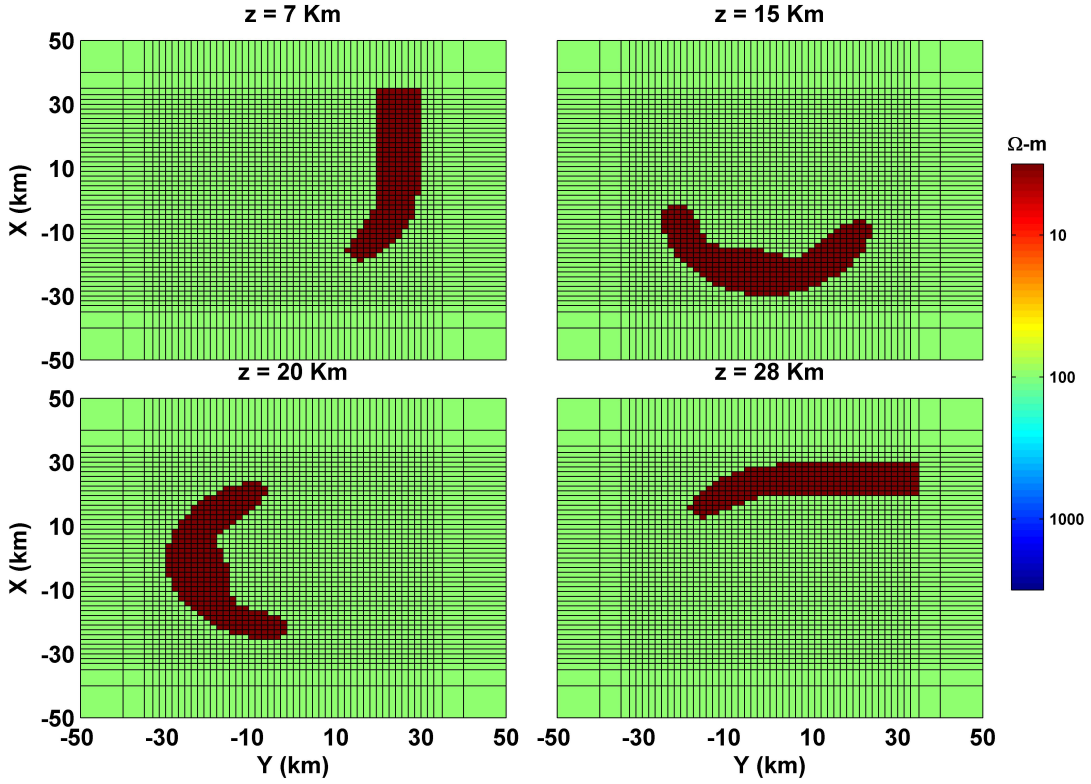

Figure 3.7: Model after grid discretization at various depth, (Clockwise from top left) at z = 7 km, 15 km, 20 km and 28 km.

3.5 Forward and Inverse Modeling

In this section, we describe the implementation of forward and inverse modeling code in detail. For efficient memory usage, a sparse storage is used where ever necessary like, storage of coefficient matrix, interpolation matrices, etc. MATLAB's inbuilt functions for dot-product, norm, Hadamard product, etc. are used for basic manipulation of various arrays. Each time a function is called, the required space is allocated for various temporary working arrays. Once the job is completed, all the temporary arrays are cleared and just the requisite solution is returned.

Before proceeding further, we give a brief discussion on the MT data. For the 3D MT problem, the complete data set ($\mathbf{d}^{obs}$) is multi-frequency, multi-observation site and multi-component. For example, there are set of periods, each requiring separate solutions for the two independent source polarizations. For each frequency, there will be number of observation sites and for each site there will be multi-component data, e.g., there are four complex components of impedance tensor and two in VTFs.

The data set is organized according to three attributes which are referred to as *transmitter*, *datatype* and *receiver* (observation sites). These three attributes are read from the data file and are stored as fields of a structured array defined as *CData* in the code. This structure array also stores other parameters like solver information (from parameter file), etc. The attribute *transmitter* consists of a list of frequencies to setup and solve the forward problem. Fields in attribute *data type* define the data functionals such as impedance tensor, VTFs, phase tensor, etc. and are accessed by name. The attribute *receiver* provides information about all the unique observation sites (Cartesian co-ordinate system). Thus, the full data vector ($\mathbf{d}^{obs}$) is an array corresponding to different frequencies. Each, in turn, stores all the components corresponding to all the data types for all the active *receivers*. A binary array is used to keep track of certain missing components, if any, in the data set.

The data vector structure allows for mixing of different data types and simplifies addition of new data types by adding new fields corresponding to the new data type. Addition of new data type simultaneously allows computing predicted data and/or sensitivity calculations for all the observation sites by solving the forward problem only once.

3.5.1 Forward Modeling

An accurate forward modeling code forms the core of inversion code. Apart from computing model responses it is heavily used for Jacobian based computations. Fig. 3.8 gives the flowchart of various steps involved in evaluation of responses at observation sites for a given set of frequencies. In these steps one deals with basic numerical grid, model parameters, boundary conditions, forward solver and interpolation functional. Table 3.2 list all the functions and their description used in the code. In *AP3DMT*, small functions deals with each of these components and they have no direct interaction with the inversion and Jacobian functions. Hence, new forward modeling scheme like Direct Current Resistivity (DCR), CSEM, etc. can be implemented with appropriate modifications. Such a modification was made while extending divergence correction to implement inversion of 3D DCR data.

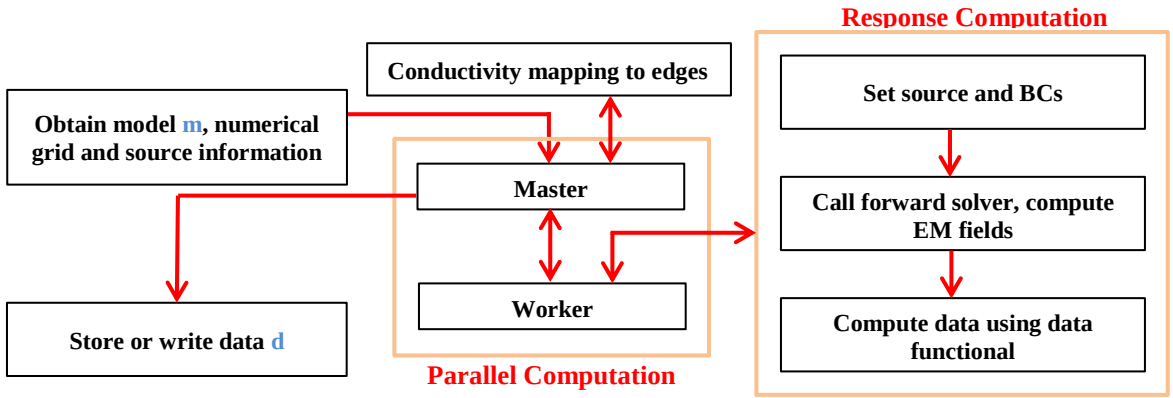

Figure 3.8: Schematic representation of *AP3DMT* for forward modeling. Frequency Parallelization is implemented for computing forward responses.

For 3D MT, the forward modeling is performed by the function, *ForwardResp*. This function computes the predicted data, **d**, and returns an array of EM solutions, **e**, (optional) corresponding to all the frequencies. These solutions can be saved and used for subsequent sensitivity calculations (e.g., gradient of the penalty functional). The major forward modeling function are,

(i) *AverageSigma* - conductivity mapping to edges,
(ii) *BoundCond* - setting up boundary conditions,
(iii) *DiscreteSolver* - computing EM field at grid nodes, and
(iv) *OutInterp* - predicted data at observation sites.

The detailed description of these functions follows.

AverageSigma maps the model parameters to the edges of numerical grid. Currently, the model parameters, σ, are defined as log conductivity. For each grid node the average conductivity, $\bar{\sigma}$, is defined as weighted average of $\exp(\sigma)$ of four surrounding cells to each edge. If any other mapping scheme is used, like harmonic mean instead of arithmetic mean, then only changes in this function are required with other portions remaining unchanged.

Once, model parameters are mapped to the edges, *BoundCond* is called where right hand side vector, **s**, is formed and stored as a column vector depending on the application. The array, **s**, is a matrix with two columns corresponding to the two source polarizations. Either the total field or the scattered field solution can be computed (the details are given in subsection 2.4.1). Arbitrary sources and the boundary data can be easily incorporated for another technique like CSEM.

In *DiscreteSolver* the coefficient matrix, with boundary conditions enforced, is formed and stored in a sparse format. The coefficient matrix is made symmetric simply by multiplying each row by integration volume elements and the implementation of the transposed solver is straightforward. The matrix equation solver, *bicgstab*, is called and the field solutions are obtained. This function includes divergence correction subroutines also.

The *OutInterp* uses the previously computed interpolation functional (discussed in the next paragraph) and electric fields to evaluate the desired data at all observation sites. For computation of sensitivity, additional components may be returned (e.g., all elements of impedance tensor) for some data types like apparent resistivity and phase and phase tensor. The whole process from *BoundCond* to *OutInterp* is looped over number of frequencies and all the data and solutions are assembled together in *ForwardResp*. To accommodate new data from new sources (for e.g., CSEM, DCR, etc.), the structure array may be used [4].

The data functionals $F(\mathbf{m})$ can be expressed in terms of (*i*) electric field; (*ii*) magnetic field (mapped from electric field); (*iii*) interpolation operator for electric and magnetic fields and (*iv*) functionals of the measured responses, e.g., impedance tensor and/or VTFs, phase tensor, etc. To evaluate this data functional, appropriate interpolation functional on the basis of observation sites are designed. The function *interpcoeff* is simple and it computes the basic interpolation functionals (λ_{ex}, λ_{ey}, λ_{bx}, etc. in sparse storage) required to evaluate $\mathbf{E}_x$, $\mathbf{E}_y$, $\mathbf{H}_x$, $\mathbf{H}_y$ and $\mathbf{H}_z$ at the observation site and applies these to solutions for both

[4] **d** can be made as **d**.MT, **d**.DCR, etc

polarizations. For computation of magnetic field from electric field, the transformation $\mathbf{h} = \mathbf{Te}$ ($\mathbf{T}$ is a transformation matrix) has been incorporated in the interpolation coefficients so that this functional can be applied directly to the field vector. The transformation matrix, $\mathbf{T}$, is scaled by $(-i\omega\mu)^{-1}$ to keep it real and frequency independent. After computing $\mathbf{h}$ fields, they are multiplied by $(-i\omega\mu)^{-1}$ to take care of the scaling factor. Once the field components are computed, the impedance tensor and/or the VTFs or any other transformations required for a particular data type (e.g., apparent resistivity or phase) can be computed. Multiple data types are supported with appropriate selection. The interpolation implemented in the current version is based on tri-linear interpolation coefficients. Since these interpolation coefficients are stored in a sparse format and do not affect other modules, other functions would remain unchanged (e.g., calculation of impedance). Further, addition of new data types or their combinations will not demand any changes to interpolation aspect.

For a different numerical discretization like, finite element method with a non-structured grid, changes in the construction of coefficient matrix and boundary conditions (if any) are required. Since the system matrix is explicitly formed and stored, any change in its structure will not affect the matrix solver which is not the case when the forward operator is implemented using a matrix-free approach [39, 71].

Table 3.2: Main functions used in *AP3DMT* for forward modeling. (* MATLAB's inbuilt function)

Functions	**Description**
FwdResponse	Compute predicted data and field solutions (optional)
Intercoeff	Compute interpolation coefficients i.e. transformation matrix
AverageSigma	Average volume weighted conductivity on the internal nodes
BoundCond	Set source and boundary conditions
DiscreteSolver	Forward solver $\mathbf{A^{-1}}$ with inbuilt divergence correction
divcorrection	Perform divergence correction, called by *DiscreteSolver*
bicgstab*	BiCGSTAB: matrix equation solver, called by *DiscreteSolver*
pcg*	Conjugate Gradient: matrix equation solver, called by *divcorrection*
ilu*	Perform Incomplete **LU** decomposition, called by *DiscreteSolver*
OutInterp	Compute predicted response

3.5.2 Inversion

In the inversion schemes, as discussed in section 2.5, the basic working vectors are: data, $\mathbf{d}^{obs}$; forward mapping, $F(\mathbf{m})$; model parameter vector, $\mathbf{m}$; MT field solution, $\mathbf{e}$; Jacobian, $\mathbf{J}$ and the model and data covariance matrices, $\mathbf{C_m}$ and $\mathbf{C_d}$ respectively. During the inversion, various operations are performed like, multiplication of data by $\mathbf{C_d}^{-1/2}$, multiplication of model parameters by $\mathbf{C_m}$, $\mathbf{C_m}^{1/2}$ or $\mathbf{C_m}^{-1/2}$, multiplication of $\mathbf{J}$ with arbitrary model space vector and $\mathbf{J^T}$ with arbitrary data space vector. In *AP3DMT*, these are standalone operators with no direct interaction with the inversion function, hence, these can easily be used in any other new inversion algorithm. Symmetric covariance operators, both data and model, are standalone with input and output being the data vector or model parameter vector. For data covariances, full matrix can be provided. However, for our inversion tests, we have used diagonal error covariances into error functional for noisy measurements; these are generally based on the standard deviations of the measurements. The model covariances are much more complicated, like $\mathbf{C_m}^{-1}$ can be represented by $\mathbf{D}^T\mathbf{D}$, where $\mathbf{D}$ is a finite difference approximation of the gradient or the Laplacian (∇^2) operator, as in Occam inversion [25], or as defined by Kelbert et al. [70]. Since this function is completely independent of other functions, various covariance operators can be easily be designed and implemented and tested for 3D MT problems. Thus the code is flexible and adoptable for the new development.

Among all the operations involving forward mapping, Jacobian, data and model covariances, Jacobian is the most time consuming. Explicit formation of the Jacobian is avoided by computing its product with a vector. Since, the Jacobian is represented as product of three components as $\mathbf{J} = \mathbf{L}\mathbf{A}^{-1}\mathbf{P}$ (as explained in subsection 2.5.3), the multiplication is performed using these three components. Fig. 3.9 gives the flowchart of various steps involved and Table 3.3 gives the list of main functions used in inversion. The interaction among various components of Jacobian is efficiently and smoothly managed in sensitivity function *JacobJob*. Details of how these components are formed for a single frequency, f, are given below.

3.5.2.1 Sensitivity functional L

The function, *LpJobs*, implements the sensitivity functionals for use in sensitivity computations. This function performs the linearization of the data functional with respect to variations in the EM solution $\mathbf{e}$. Here, the $\mathbf{L}_f$ matrix is not explicitly formed, rather the

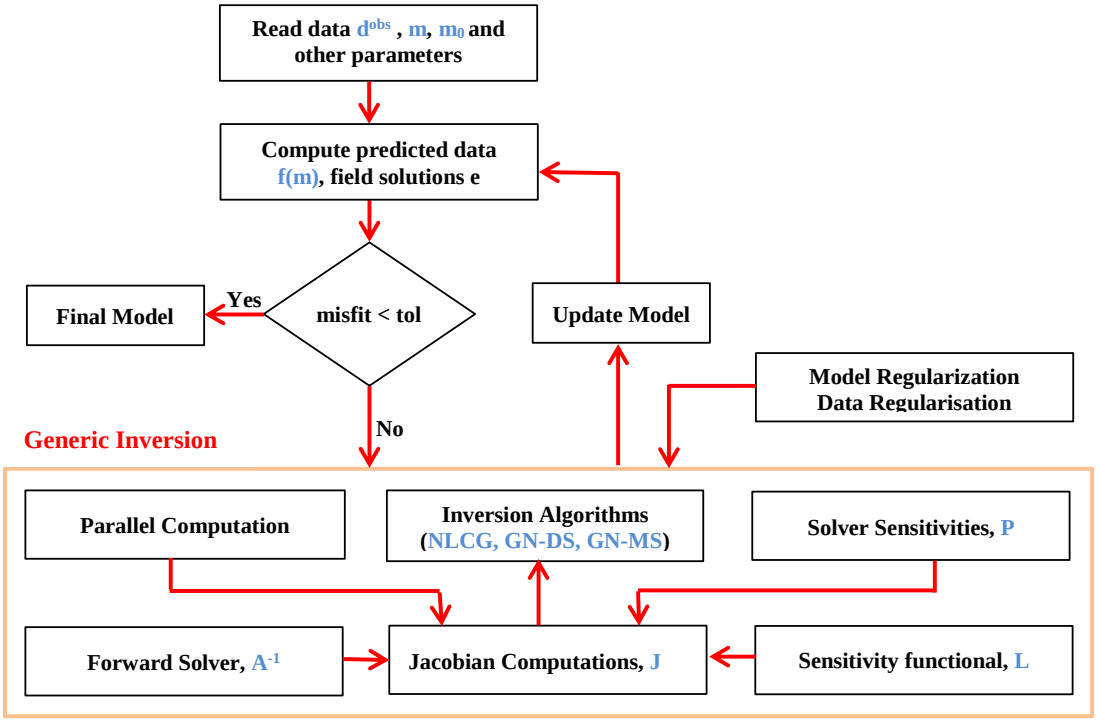

Figure 3.9: Schematic representation of *AP3DMT* for inverse modeling. Frequency Parallelization is implemented for multiplication of $\mathbf{J}$ and $\mathbf{J}^T$ with model and data space vectors respectively.

product of $\mathbf{L}_f$ and vector $\mathbf{v}_{\mathbf{e}f}$ or of $\mathbf{L}_f^T$ and vector $\mathbf{v}_{\mathbf{d}f}$ is computed using the components of $\mathbf{L}_f$. If the measured component is EM field itself, then $\mathbf{L}_f$ simply computes the interpolated values of the fields (using $\mathbf{E}_x$, $\mathbf{E}_y$, etc.) at the observation sites and stores them in a vector. For *datatype* like impedance tensor or VTFs the components of $\mathbf{L}_f$ include the interpolation and transformation matrices (λ_{ex}, λ_{ey}, λ_{bx} etc.), the magnetic field and the impedance values at the observation sites (Appendix B for detailed expression of $\mathbf{L}_f$). For other *datatype* viz. apparent resistivity and phase or phase tensor, first the perturbations in impedance are computed and therefrom the perturbations in the desired *datatype* are computed. *LpJobs* performs the product of $\mathbf{L}_f$ with $\mathbf{v}_{\mathbf{e}f}$ to form $\mathbf{v}_{\mathbf{d}f} = \mathbf{L}_f\mathbf{v}_{\mathbf{e}f}$. *LTpJobs* multiplies this vector by the appropriate data components to form $\mathbf{L}_f^T\mathbf{v}_{\mathbf{d}f}$. Here, instead of implementing $\mathbf{L}_f^T\mathbf{v}_{\mathbf{d}f}$, we have implemented $(\mathbf{v}_{\mathbf{d}f}^T\mathbf{L}_f)^T$. This does away with the need to implement $\mathbf{L}_f^T$.

For the addition of new data type or mixing different data types, appropriate changes are required in *OutInterp*, *LpJobs* and *LTpJobs* for computing the predicted data and the sensitivity calculations.

3.5.2.2 Forward Solver $\mathbf{A}_\mathbf{m}^{-1}$

The function *DiscreteSolver* is common to the forward modeling, and sensitivity computations. This function includes the standard BiCGSTAB iterative solver (MATLAB library) for solving eq 2.4.1 with the incomplete **LU** decomposition for pre-conditioning and the pre-conditioned conjugate gradients for divergence correction. For each frequency, the output vector, $\mathbf{v_e}$, consists of two sets of solutions corresponding to the two source polarizations, requiring two solver calls with different right hand side vectors (boundary conditions) for each frequency. The *DiscreteSolver* implements the general solver, allowing for boundary conditions and solution for both the forward problem and its transpose. Addition of new sources can be easily incorporated by appropriate boundary conditions and additional solver calls for new solutions. Also, it is very easy to replace one solver with another like replacing BiCGSTAB with QMR (Quasi-Minimal Residual). With MATLAB inbuilt solvers and preconditioners (like incomplete Cholesky decomposition, Incomplete **LU** decomposition, Modified Incomplete **LU** decomposition etc.), one can test various numerical schemes easily.

3.5.2.3 Solver Sensitivity P

One of the main components of Jacobian is $\mathbf{P}_f$, which defines the product of the sensitivity of the coefficient matrix $\mathbf{A}_{\mathbf{m},f}$ to the model parameters (log conductivity) with the electric field. eq 2.5.11 provides the derivative of the field with respect to the parameters. This derivative can be interpreted as the field due to a collection of sources described as the product of derivatives of $\mathbf{A}_{\mathbf{m},f}$ and the field vector $\mathbf{e}$. As discussed in previous chapter, the solver sensitivity $\mathbf{P}_f$ consists of two components. The first part provides the scaled derivative of the coefficient matrix with respect to the model parameters (see eq 2.5.11). For a FD approximation, only the diagonal elements of the matrix $\mathbf{A}_{\mathbf{m},f}$ depend on conductivity elements in the form of $i\omega\mu\bar{\sigma}$. Here, $\bar{\sigma}$ is the volume average of the conductivity of nearby cells. Hence, each parameter contributes to twelve internal nodes. This results in a maximum of twelve non-zero elements per parameter. The derivatives of $\mathbf{A}_{\mathbf{m},f}$ are made real and frequency independent by simply scaling these with the factor $(i\omega\mu)^{-1}$. Once these derivatives are obtained, these are multiplied with the model parameter vector to perform the operation $\bar{\mathbf{P}}\mathbf{v}_\mathbf{m}$. This product can be viewed as $\sum_{i=1}^{M} m_i \mathbf{A}'_{m_i,f}$ where $\mathbf{A}'_{m_i,f}$ is a column vector of length N_e with only 12 non-zero elements with their indices defined by the mapping. These

non-zero elements are $-V_i exp(m_i)$ where, V_i is the integration volume for the i^{th} cell having parameter m_i defined as log conductivity. Afterwards, this vector is multiplied element by element with the electric field solution, $\mathbf{e}$, corresponding to frequency, f, and finally by the factor $i\omega\mu$ to obtain the final solution $\mathbf{P}_f\mathbf{v_m}$ in *PpJob*. For the transpose case, these steps are reversed in *PTpJob*. In this function, first the input vector $\mathbf{v}_{\mathbf{e}f}$ is multiplied element by element with the electric field $\mathbf{e}_n$ resulting in $\mathbf{v}_{\mathbf{e}f} = (\mathbf{v}_{\mathbf{e}f}^a, \mathbf{v}_{\mathbf{e}f}^b)$ (solutions for two source polarizations). Finally, the operation $i\omega\mu[\bar{\mathbf{P}}^T(\mathbf{v}_{\mathbf{e}f}^a + \mathbf{v}_{\mathbf{e}f}^b)]$ is performed resulting in a model parameter vector $\mathbf{v}_{\mathbf{m}f}$.

Once the necessary product of Jacobian (or its transpose) with vector is computed the updated model is computed (using the eq 2.5.4 or 2.5.5 or 2.5.6) and next inversion iterations begin. This process continues till one of the condition is satisfied,

(i) desired data misfit is reached,
(ii) local minima in case of *NLCG*,
(iii) data misfit starts to increase (in case of *GN*),
(iv) data misfit change is less than the threshold (in case of *GN*),
(v) maximum inversion iteration reached.

Table 3.3: Main functions used in *AP3DMT* for inverse modeling.

Functions	**Description**
NlcgSolver	non-linear conjugate gradient after Kelbert et al. [71]
lineSearch	perform line search
FullPenality	computes full penality functional
GradientCal	computes gradient of penality functional
CG_Data_Space	data space conjugate gradient
CG_Model_Space	model space conjugate gradient
JacobJob	multiply Jacobian or its transpose by vector
LpJob	multiply Data Sensitivity, **L** by vector
LTpJob	multiply **L** transpose by vector
PpJob	multiply Solver Sensitivity **P**, by vector
PTpJob	multiply **P** transpose by vector
MatrixOccam	Occam matric (first or second order), Model Regularisation
CmFormulate	Model regularisation after Kelbert et al. [70]
FormCdInv	Data regularisation matrix (diagonal matrix)
Residual	Compute misfit between observed and predicted data

The task of multiplication of Jacobian (or its transpose) with a vector or computation of responses for a single frequency is independent of other frequencies hence, this task can be performed in parallel. We have implemented a coarse grain parallelization over frequency similar to that of Siripunvaraporn and Egbert [157]. This scheme is discuss in the following section.

3.6 Parallelization over Frequency

AP3DMT implements a coarse grained parallelization over forward problem. Implementation of parallelization over number of frequencies is quite simple. Fig. 3.10 explains the basic scheme of implementation. MATLAB's inbuilt function *parfor* is used which executes loop iterations in parallel. The command *matlabpool* (*parpool* in 2015a version) starts a pool, with the pool size specified by parallel preferences in the default setting or one can manually start a pool with desired number of workers. After initialization, one processor acts as the master, and the others act as workers. The master performs the main job of the code until a step requiring parallel computations (like forward problem or multiplication of Jacobian with a vector) is reached i.e. *parfor* loop over number of frequencies. At this point the master sends a message indicating nature of the task and the necessary input data to all the active workers. The worker performs the task for a single frequency by calling appropriate functions like *ForwardResp* or multiplication of Jacobian with a vector. When the task has been performed, it returns the solution to master. These functions are same as called in the sequential version [5]. We describe the multiplication of Jacobian (or its transpose) with a vector as this is the major task while performing inversion.

For the multiplication of $\mathbf{J}$ with the model parameter vector $\mathbf{v_m}$, master sends a copy of $\mathbf{v_m}$ and other necessary information to each worker. Each worker computes $\mathbf{v_{d}}_f = \mathbf{L}_f \mathbf{A}^{-1}_{\mathbf{m},f} \mathbf{P}_f \mathbf{v_m}$. The product $\mathbf{v_s}_f = \mathbf{P}_f \mathbf{v_m}$ results in a matrix with two columns for the two source polarizations. Hence, for $\mathbf{v_e}_f = \mathbf{A}^{-1}_{\mathbf{m},f} \mathbf{v_s}_f$, the matrix solver is called twice for different right hand side vectors. Finally, the product $\mathbf{v_d}_f = \mathbf{L}_f \mathbf{v_e}_f$ is obtained using the linear combination of product of each block of $\mathbf{L}_f$ matrix and column of $\mathbf{v_e}_f$ (eq 2.5.16). If the operation $\mathbf{J}^T_f \mathbf{v_d}_f$ is required (like $\mathbf{J}^T\mathbf{J}$ in *GN-CG*) then the same worker performs the operation $\mathbf{v_m}_f = \mathbf{P}^T_f \mathbf{A}^{-1}_{\mathbf{m},f} \mathbf{L}^T_f \mathbf{v_d}_f$. If only the gradient is required, like in *NLCG*, then the

[5] Note, that in absence of any pool *parfor* is equivalent to *for* with the difference that last frequency will be solved first

master sends the residual vector $\mathbf{r}_f$, corresponding to each frequency, to each worker and the workers compute the gradient. Once the gradient is computed for each frequency, the linear sum is computed and final $\mathbf{J}^T\mathbf{v_d}$ is formed. *JacobJob* implements all these three tasks viz. $\mathbf{Jv_m}$, $\mathbf{J}^T\mathbf{v_d}$ and $\mathbf{J}^T\mathbf{Jv_m}$ computations depending on the call. $\mathbf{JJ}^T\mathbf{v_d}$ is computed using the combination of $\mathbf{J}^T\mathbf{v_d}$ and $\mathbf{Jv_m}$. Hence, for $\mathbf{JJ}^T\mathbf{v_d}$ two parallel loops are required, one for computing $\mathbf{J}^T\mathbf{v_d}$ and another for $\mathbf{Jv_m}$. This is in contrast to $\mathbf{J}^T\mathbf{J}$ where the product is computed in a single parallel loop. Fig. 3.11 gives the details for these computations.

The communication between master and worker can be minimized by transferring some fixed components or data which are used very frequently to all the active workers using *WorkerObjWrapper*. It manage persistent state on all active *matlabpool* workers thus avoids transfer of data from master to worker during each parallel call.

Parallel

Master:

⋮

Parfor over number of frequencies

1. Send : Copy of model parameters
2. Send : Frequency index number
3. Receive : Solution from workers after completion of the task

end

⋮

Worker:

1. Receive copy of model parameters
2. Receive the frequency
2. Used previously stored information (optional)
3. Perform necessary task like, $F(\mathbf{m})$, $\mathbf{J}^T\mathbf{r}$, etc.
4. Send solution to master.

Serial/Parallel version Inversion Algorithm

⋮

If parallel

Compute penality functional or gradient over frequency

else

parfor loop just acts as for loop over frequency

end

⋮

Figure 3.10: Pseudo-code for the parallelization over frequencies of a task like forward response.

Parallel for computations of $\mathbf{J}^T\mathbf{v_d}$; $\mathbf{J}^T\mathbf{J}\mathbf{v_m}$ and $\mathbf{J}\mathbf{J}^T\mathbf{v_d}$

Master (Computation of $\mathbf{J}^T\mathbf{v_d}$):
Send : data space vector, $\mathbf{v_{d}}_f$, and frequency index, f
Parfor over number of frequencies (N_f)
Receive : data space vector, $\mathbf{v_{d}}_f$, and frequency index, f
Compute : $\mathbf{v_{m}}_f = \mathbf{J}_f^T\mathbf{v_{d}}_f$
Send : send moddel space vector, $\mathbf{v_{m}}_f$, to master
end
Receive : all the solution vectors $\mathbf{v_{m}}_f$
Compute : $\mathbf{J}^T\mathbf{v_d} = \mathbf{v_m} = \sum_{f=1}^{N_f} \mathbf{v_{m}}_f$

Master (Computation of $\mathbf{J}^T\mathbf{J}\mathbf{v_m}$):
Send : model space vector, $\mathbf{v_m}$, and frequency index, f
Parfor over number of frequencies (N_f)
Receive : model space vector, $\mathbf{v_m}$, and frequency index, f
Compute : $\mathbf{v_{d}}_f = \mathbf{J}_f\mathbf{v_m}$
Compute : $\mathbf{v_{d}}_f = \mathbf{J}_f^T\mathbf{v_{d}}_f$
Send : send data space vector, $\mathbf{v_{d}}_f$, to master
end
Receive : all the solution vectors $\mathbf{v_{d}}_f$
Compute : $\mathbf{J}^T\mathbf{J}\mathbf{v_m} = \mathbf{v_m} = \sum_{f=1}^{N_f} \mathbf{v_{d}}_f$

Master (Computation of $\mathbf{J}\mathbf{J}^T\mathbf{v_d}$):
Send : data space vector, $\mathbf{v_{d}}_f$, and frequency index, f
Parfor over number of frequencies (N_f)
Receive : data space vector, $\mathbf{v_{d}}_f$, and frequency index, f
Compute : $\mathbf{v_{m}}_f = \mathbf{J}_f^T\mathbf{v_{d}}_f$
Send : send model space vector, $\mathbf{v_{m}}_f$, to master
end
Receive : all the solution vectors $\mathbf{v_{m}}_f$
Compute : $\mathbf{J}^T\mathbf{v_d} = \mathbf{v_m} = \sum_{f=1}^{N_f} \mathbf{v_{m}}_f$
Send : model space vector, $\mathbf{v_m}$, and frequency index, f
Parfor over number of frequencies (N_f)
Receive : model space vector, $\mathbf{v_m}$, and frequency index, f
Compute : $\mathbf{v_{d}}_f = \mathbf{J}_f\mathbf{v_m}$
Send : send data space vector, $\mathbf{v_{d}}_f$, to master
end
Receive : all the solution vectors $\mathbf{v_{d}}_f$

Figure 3.11: Pseudo-code showing the multiplication of Jacobian (and/or its transpose) with a vector in parallelization over frequencies as implemented in *GN-CG* (model space and data space) and *NLCG*.

3.7 Summary

We have presented the development of a MATLAB based code, *AP3DMT*, for modeling and inversion of 3D MT data. The capability of this program for performing inversion is demonstrated through two synthetic test models. The key feature is its compactness and simplicity and its potential as a tool for rapid development and testing of new applications. Basic functions are simple and readily available. Thus, it would be easy to replace one scheme with another one like replacing *NLCG* with quasi-Newton [e.g. 111, 114] or one iterative solver with another one. *AP3DMT* provides a natural platform for development and comparison of not only inversion search algorithms but also various modeling schemes or its components like iterative solvers, preconditioners and modified finite differences schemes.

Use of MATLAB platform and its in-built functions such as BiCGSTAB, incomplete **LU**, etc. may be considered both as its strength and weakness. Use of these global functions make the code more versatile, compact and user friendly which will open opportunities for development and experimentation. It may be mentioned that the MATLAB functions have multiple checks and are therefore slower than the corresponding routines of a Fortran code. However, in a MATLAB code the implementation, parallelization etc. are much easier and one can use a mixed language environment for bottleneck computations.

Chapter 4

NUMERICAL EXPERIMENTS WITH CODE : *AP3DMT*

4.1 Preamble

The performance of the code must be verified for its accuracy, efficiency and robustness and versatility over a variety of standard models. A rigorous testing should include different synthetic models from literature as well as real field data case study to demonstrate robustness of the algorithm for real field scenario. For this purpose, we present here two synthetic models from literature [39, 102] and a real field data inversion. The field data set was acquired over the past decade using Broadband MT survey by our group along the Roorkee-Gangotri profile. The 2D inversion of this dataset has been performed earlier [62, 103]. This dataset was inverted using *ModEM* [39] by Devi et al. [32] and we have used *AP3DMT* to invert the same dataset and compared the results with *ModEM* inverted model.

4.2 Validation of Forward Modeling Code

The forward modeling has been tested and validated over a variety of 3D standard models available in literature. For demonstration, we have selected one synthetic test model [210]. The test model is the 3d2 model from Comparison of Modeling in Electromagnetic Induction (COMMEMI) project [210]. The model consists of three layers with resistivities of 10, 100, and 0.1 Ω-m. The upper two layers are 10 and 20 km thick, respectively. Two adjacent blocks of dimension $X:\ 40 \times Y:\ 20 \times Z:\ 10$ km are embedded in the top layer (Fig. 4.1). The resistivity contrast ratio in this model is up to 1000 which is the main difficulty for

the calculations is the presence of a strong resistivity contrast at the surface. The irregular grids were designed and used for this model. The smaller cells were taken to be near the adjacent planes and the cell sizes extended to large distances from the blocks boundaries. The model was composed of $54 \times 48 \times 34$ cells. The observation sites were taken along the central profile $x = 0$ (north) for y (east) between ± 60 km. These sites coincides with the grid nodes. Fig. 4.2 shows the comparison of numerical results from *ModEM* [39] and our code *AP3DMT* in terms of apparent resistivities and phases and Fig. 4.3 shows relative error and Fig. 4.4 shows absolute error among the solutions. There is an excellent overall agreement between responses computed using *ModEM* and *AP3DMT*. The maximum absolute relative error in amplitude, Z_{xy}, is less than 4% while maximum absolute difference are below 1 Ω-m while the corresponding values for amplitude, Z_{yx}, are 2.5 % and 1 Ω-m respectively. The absolute relative error in phase, Z_{xy}, is below 2% and the absolute differences are under 1 degree. The absolute relative error and absolute differences in phase, Z_{yx}, are below 1.5% and below 0.75 degree. The misfit mainly occur at the resistivity contrast and over resistivity body. Table 4.1 summaries the results.

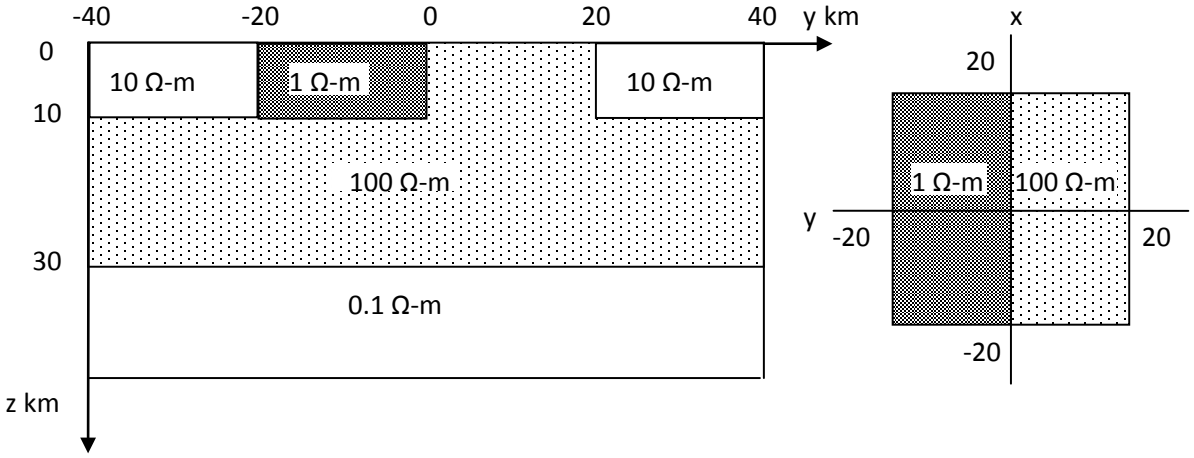

Figure 4.1: The 3d2 COMMEMI model [210]. (left) cross-section, and (right) plan view.

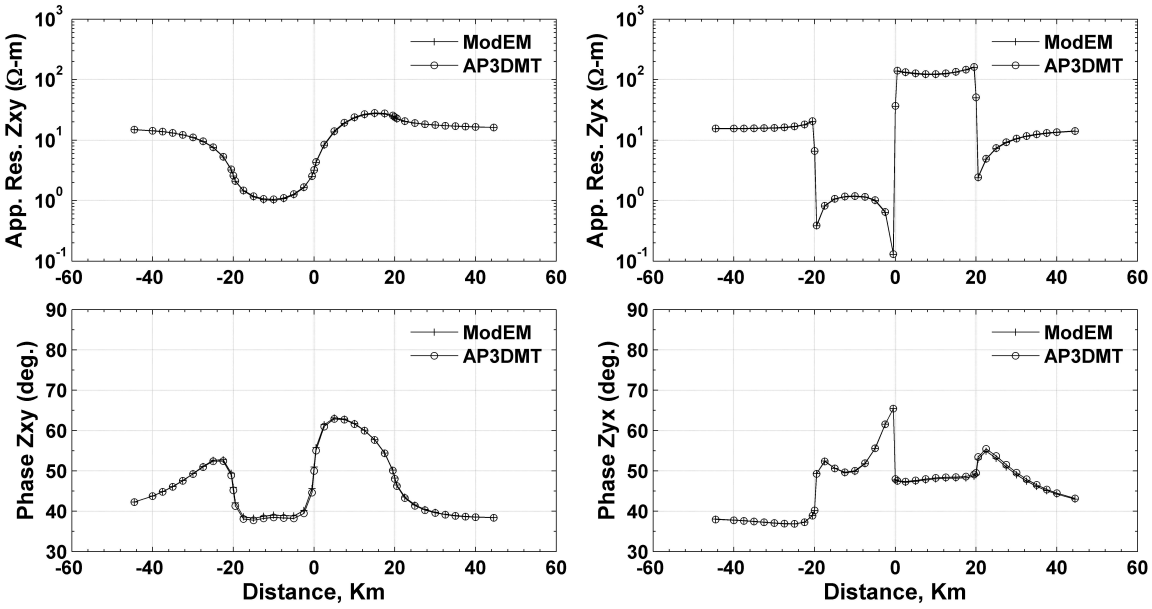

Figure 4.2: The 3d2 COMMEMI model simulation at 100 sec. (Top clockwise) Panels are apparent resistivities Z_{xy} and Z_{yx}, phases of impedances Z_{yx} and Z_{xy}. The comparison are between results from *ModEM* and our code *AP3DMT*.

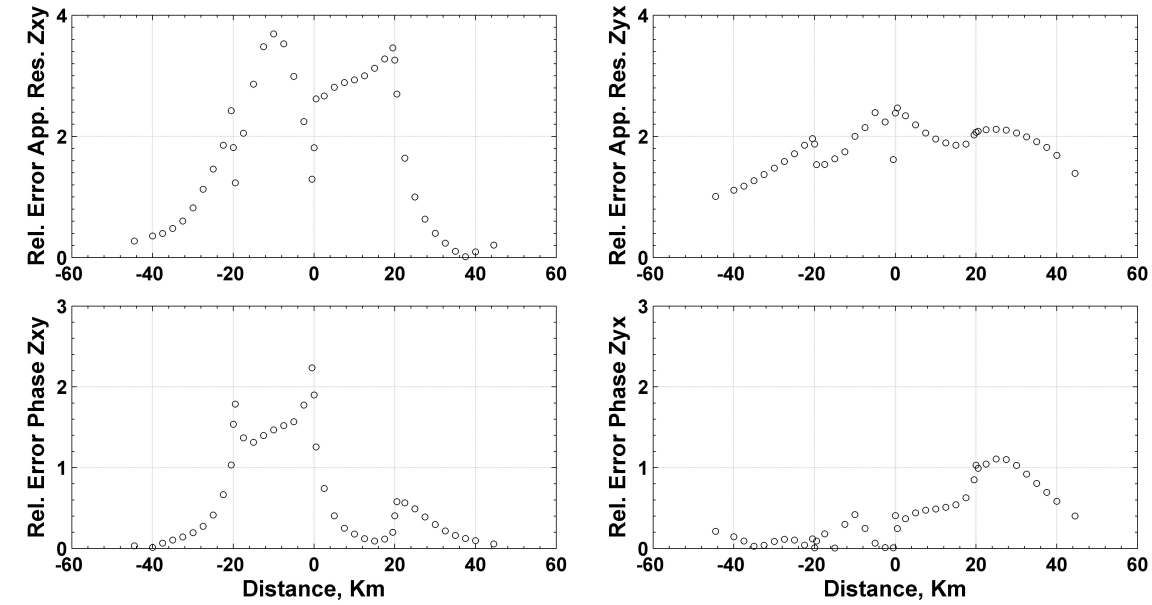

Figure 4.3: (Top clockwise) Absolute relative deviation between the numerical results of *ModEM* and *AP3DMT* for apparent resistivities Z_{xy} and Z_{yx}, phases of impedances Z_{yx} and Z_{xy}.

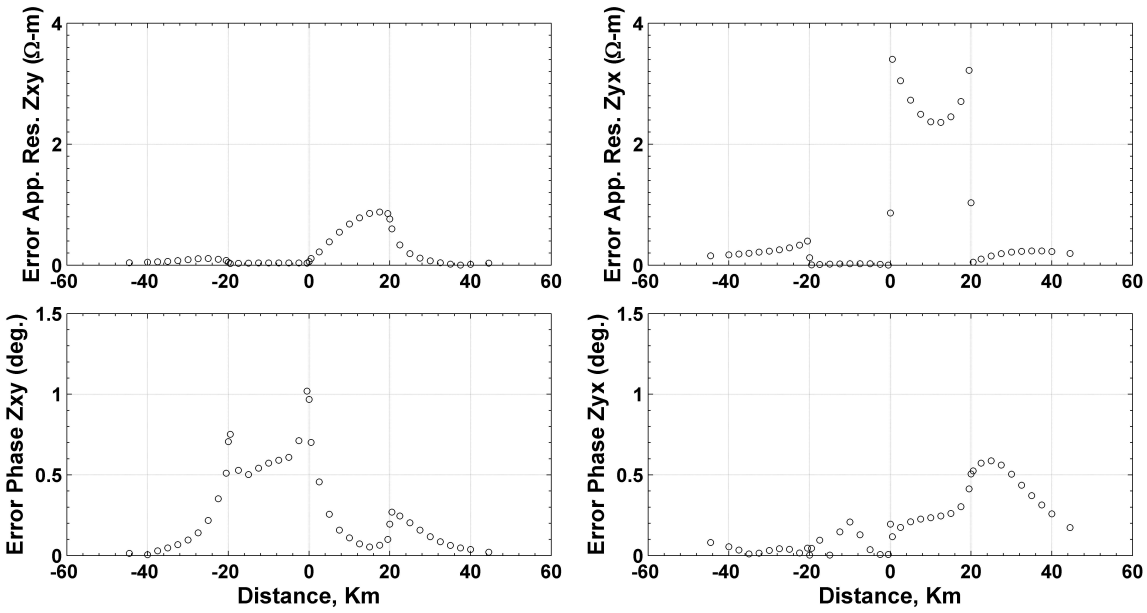

Figure 4.4: (Top clockwise) Absolute deviation between the numerical results of *ModEM* and *AP3DMT* for apparent resistivities Z_{xy} and Z_{yx}, phases of impedances Z_{yx} and Z_{xy}.

Table 4.1: Table showing information about absolute and relative absolute error in amplitude and phase among the solution obtained using *ModEM* and *AP3DMT*.

Error	**amplitude** (Z_{xy})	**phase** (Z_{xy})	**amplitude** (Z_{yx})	**phase** (Z_{yx})
Abs. Error (max)	0.88	1.02	3.40	3.40
Abs. Error (min)	0.00	0.01	0.00	0.00
Rel. Abs. Error, % (max)	3.69	2.23	2.47	2.47
Rel. Abs. Error, % (min)	0.02	0.01	1.01	1.01

4.3 Validation of Inverse Modeling Code

4.3.1 Comparison of three Different Inversion Algorithms

To demonstrate the working of the three developed algorithms, we consider a two block model [101]. The model consists of 10 and 1000 Ω-m blocks in a homogeneous 100 Ω-m half-space (left panel of Fig. 4.6 - 4.8). Each block has the dimension $X:\ 13 \times Y:\ 7 \times Z:\ 2$ km. The top of the blocks is located at the earth's surface ($z = 0$). There are 81 sites distributed on a regular 2D array with 3 by 3 km spacing in $x-$ and $y-$ directions respectively (marked as black dots on the first panel). The model was discretized into $39 \times 39 \times 30$ cells (plus 7 air layers above the earth surface), with nominal resolution of 1 km horizontally. The

thickness of first layer is 50 m and increases vertically with depth by a factor of 1.3. The data set for inversion consists of full impedance tensor computed for 19 periods, logarithmically spaced between $0.001 - 1000$ s using *ModEM*. We have added 2% error to the computed responses to simulate real data. An error floor of 2% of $|\mathbf{Z}_{xy}\mathbf{Z}_{yx}|^{1/2}$ is assigned to the impedance tensor. For inversion, homogeneous 50 Ω-m half-space discretized into same number of cells, is used as apriori and initial model. Constant smoothing of 0.3 was used in $x-$, $y-$ and $z-$ directions for all the models.

Before performing *GN* inversion, we have tested the effect of number of iterations in *CG*. All the inversions were run on HP Z620, Intel Xeon E2643 3.30 GHz, 32 GB with 8 cores for parallelization and the starting normalized root-mean-square (nRMS) error was 15.34. The starting regularisation parameter, λ was calculated on the basis of largest row sum of $real(\mathbf{J}^T\mathbf{J})$ during first iteration and decreased by a factor of 2 in subsequent iterations. A total of four inversion runs were performed. In the first, second and third run 3, 6 and 9 *CG* iterations were used respectively. For the fourth run varying *CG* iterations (increase with inversion iteration) were used. Fig. 4.5 shows the convergence plot for these four runs. It is observed that when the *CG* iterations were 3 and 6, the inversion stalls after 6 inversion iterations with nRMS error 1.50 and 1.27 respectively. The nRMS error reaches to desired level after 4 inversion iterations when 9 iterations were used for *CG*. It was observed that in the first 2 or 3 inversion iterations less *CG* iterations can be used and later more number of *CG* iterations. In the fourth run such approach was used. The nRMS error reaches desired level after 6 iterations. In these fours runs, Jacobian calls [1] were 47, 83, 79 and 71 respectively. Hence, less number of Jacobian calls were used in fourth run as compared to third run (one with 9 *CG* iterations) with the condition that nRMS error reached desired level. Table 4.2 summaries the results. Hence, for the next runs of *GN* we follow the fourth approach.

[1] total calls for one inversion iteration are twice of *CG* iterations plus two calls for forming right hand side vector plus one call second inversion iteration onwards (as we are using the result of previous iteration as initial guess)

Table 4.2: List of inversion results. Beside the run name, information about CG iterations, number of inversion iterations, nRMS error (* target reached), Jacobian calls are given.

Inversion	*CG calls*	*Iterations*	*nRMS Error*	*Jacobian Calls*
Run 1	3	6	1.50	47
Run 2	6	6	1.27	83
Run 3	9	4	0.93*	79
Run 4	varying	6	0.99*	71

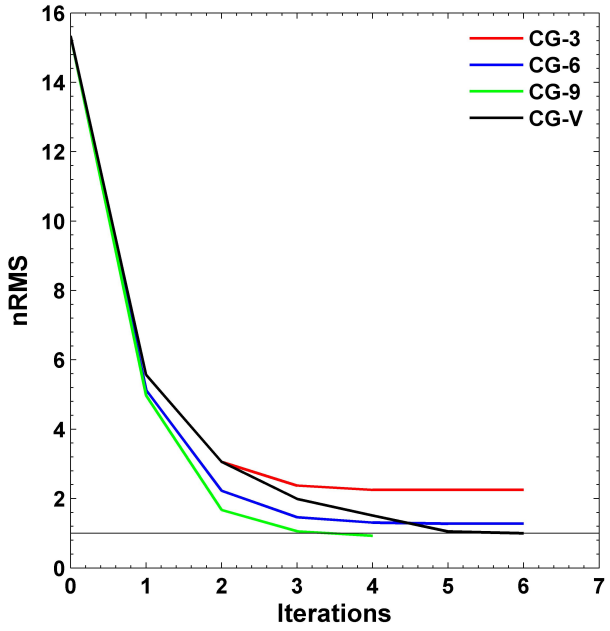

Figure 4.5: Convergence plot for *GN-MS* inversion algorithm for different number of iteration in *CG* routine. CG-3: three iterations for CG; CG-6: iterations for CG; CG-9: nine iterations for CG ; CG-V: varying iterations for CG such that they increase with inversion iteration.

First, inversion was performed using *NLCG*. In 47 iteration nRMS error reduced from 15.34 to 0.99. The inverted model (plane view slices) is shown in the second panel of Fig. 4.6 - 4.8. For *GN-MS* the nRMS error reduced to 1 in 6 iterations resulting in inverted model shown in the third panel of the same figures. Finally, using *GN-DS* the nRMS error reached 1.1 in 9 iterations. The model obtained is shown in fourth panel of the same figures. For all the three runs, same model regularisation parameters were used. All the three inversion runs gives similar results.

The result shows that in contrast to the conductive block, the top and the bottom of the resistive block are not well resolved. Furthermore, the result also shows that the bottom

of the conductive block smears to a depth of ∼3 km. While performing inversion using, *WS3DMTINV* [2] [159], for the same model Meqbel [101] observed four conductive blobs in the model at depths greater than 4 km located at the western and the eastern corners of the conductive and the resistive blocks, respectively. However, no such spurious features appears in our results. Table 4.3 gives the details of comparison of the three inversion runs.

Fig. 4.9 shows the convergence plot for all the three inversion. In general, a comparison between inversion result reveals that the *GN* requires less iterations due to its quadratic behaviour to converge to the desired error level. The quadratic behaviour, however, implies longer total run time because of the time consuming mathematical operations used in this algorithm. Further, *GN-MS* requires less iteration as compared to *GN-DS*. On the contrary, the *NLCG* algorithm requires more iterations to reach the global minimum, however, less total run time. Hence, for the subsequent inversion runs in this thesis we have used *NLCG*.

Table 4.3: List of inversion results. Beside the inversion algorithm, information about number of iterations, nRMS error, target tolerance (TT) and run time is given. These three inversions were run on HP Z620, Intel Xeon E2643 3.30 GHz, 32 GB with 8 cores for parallelization.

Inversion	*NLCG*	*GN-MS*	*GN-DS*
iterations	47	6	9
nRMS Error	0.99	1.00	1.1
TT (forward)	10^{-6}	10^{-6}	10^{-6}
TT (sensitivity)	10^{-6}	10^{-5}	10^{-5}
Time (hrs)	5:10	6:02	9:26

[2] inversion based on data-space variant of the Occam approach

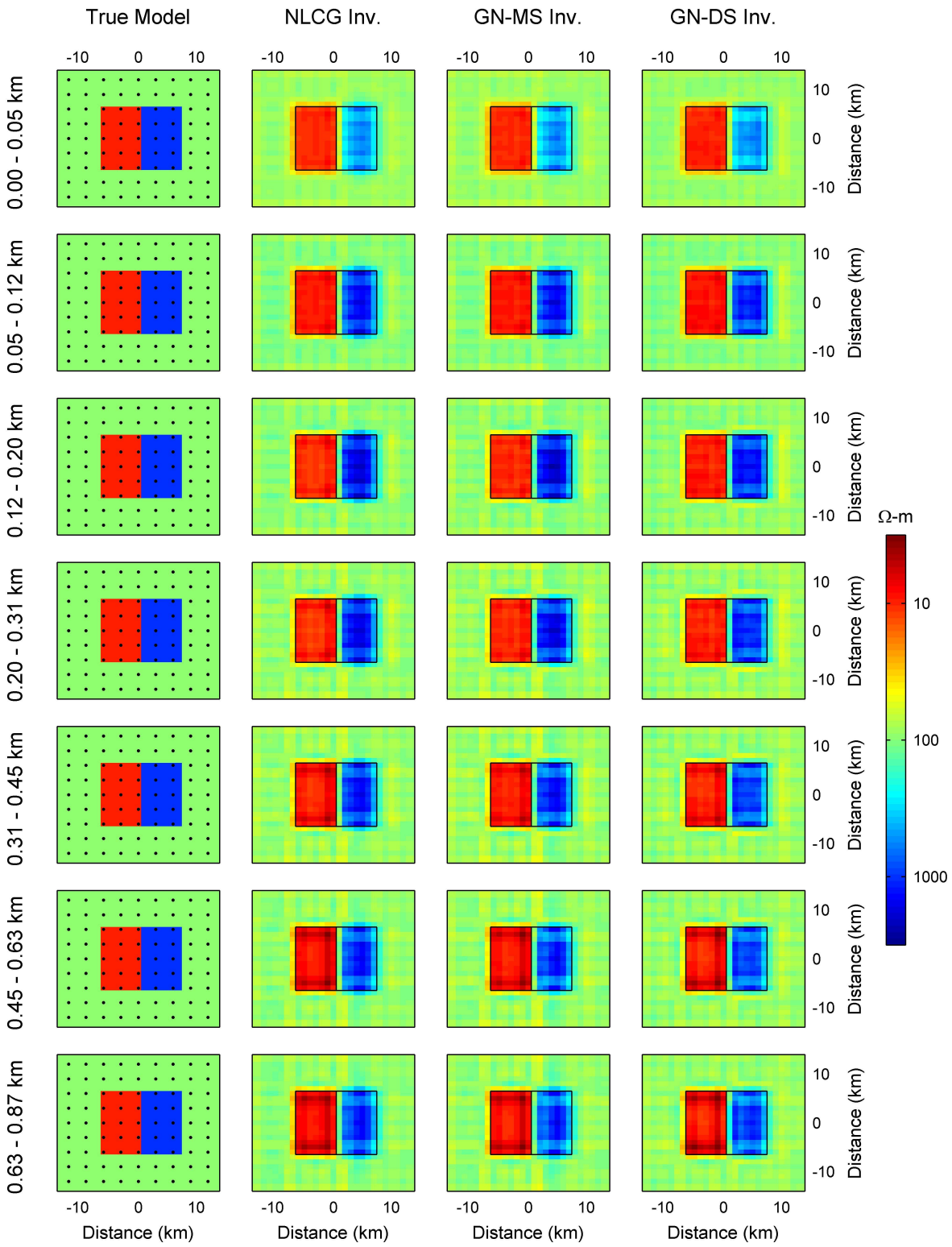

Figure 4.6: A 3D model (left panel) used to test the developed inversion schemes. The 3D inversion results, as a plane view slices (for depth 0 to 0.87 km), for *NLCG*, *GN-MS* and *GN-DS* are shown in second, third and fourth panels. The rectangles indicate the position of the original conductive and resistive blocks to generate the synthetic data. Black dots represents the observations sites.

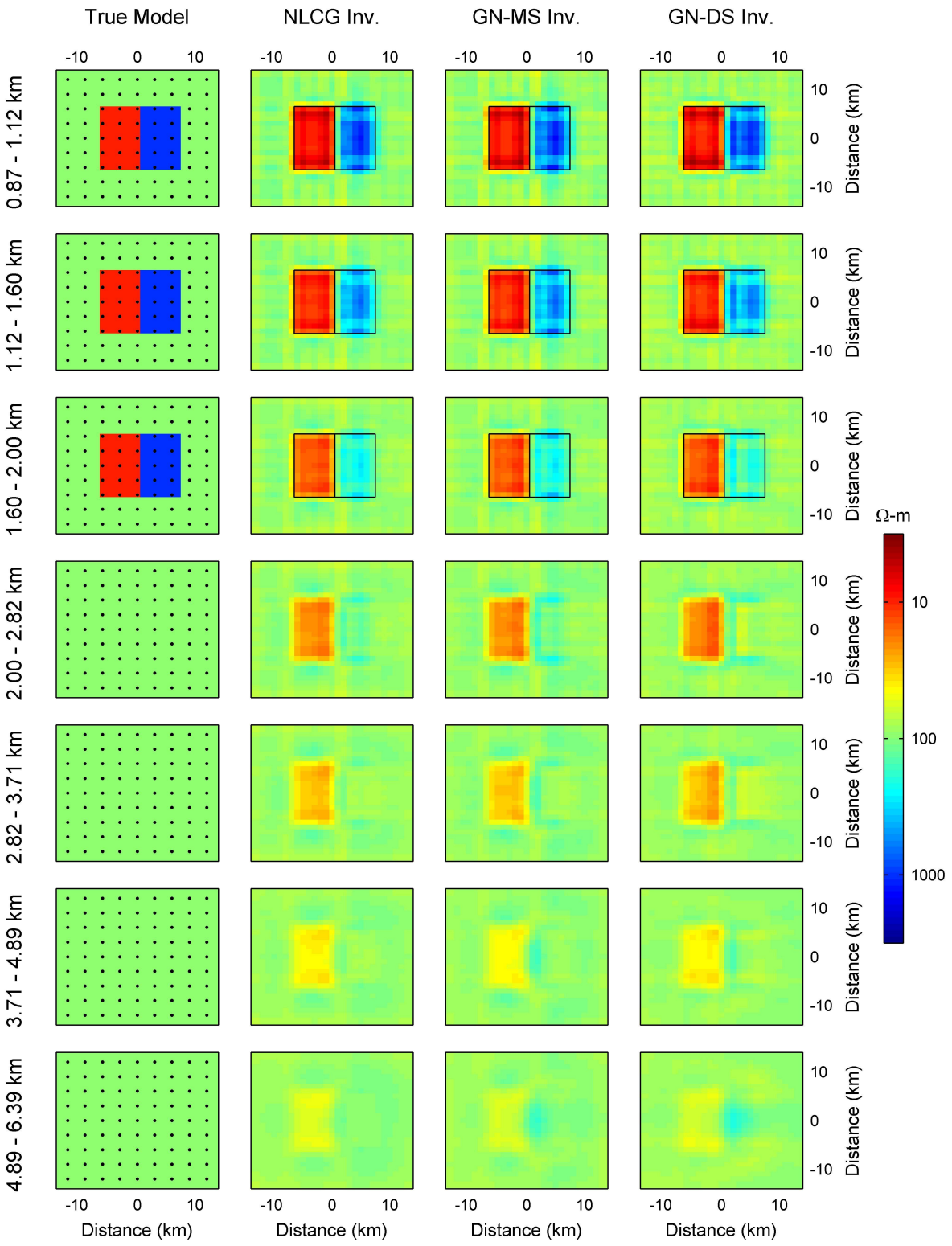

Figure 4.7: The 3D inversion results, as a plane view slices (for depth 0.87 to 6.39 km), for *NLCG*, *GN-MS* and *GN-DS* are shown in second, third and fourth panels. The rectangles indicate the position of the original conductive and resistive blocks to generate the synthetic data.

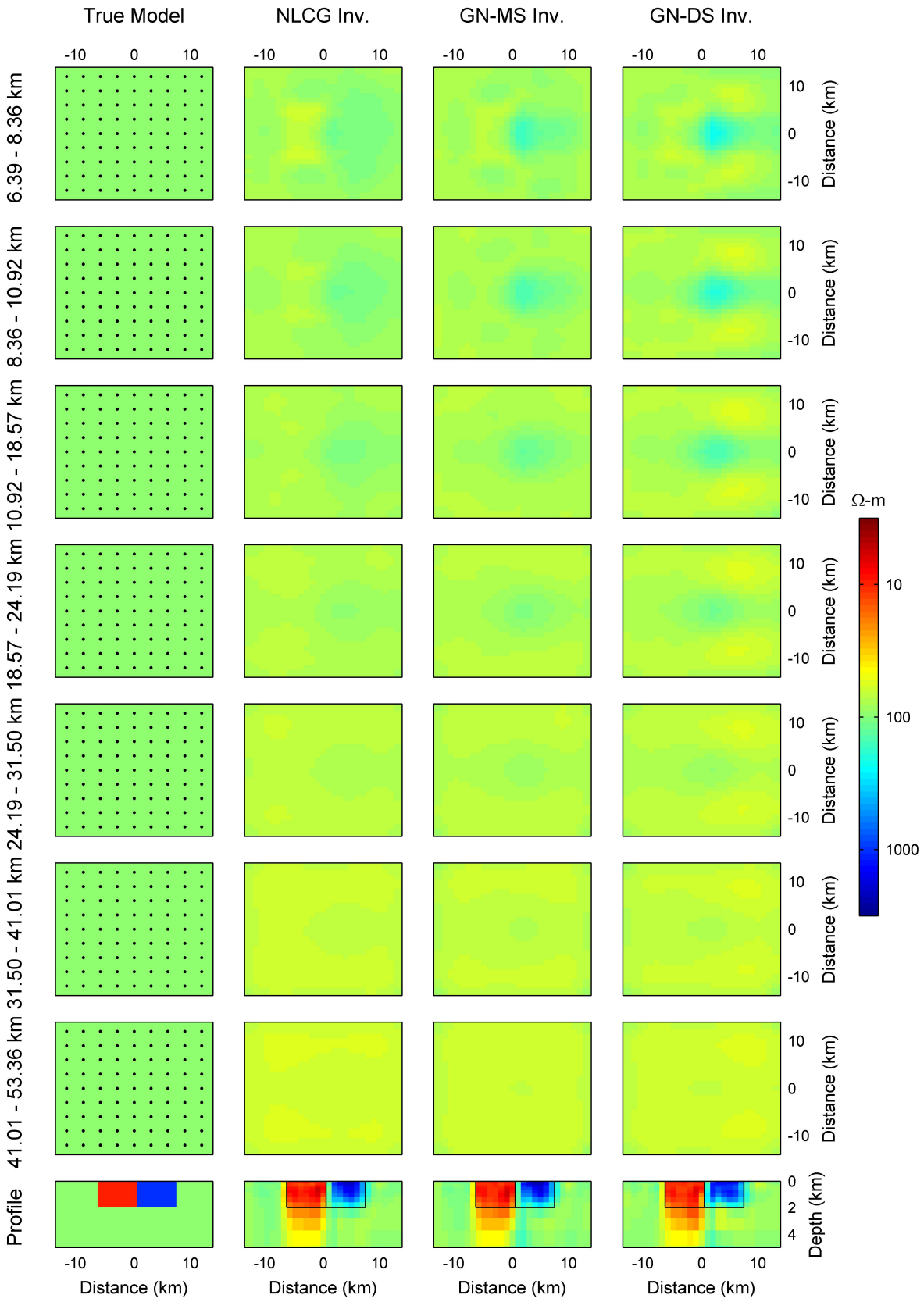

Figure 4.8: The 3D inversion results, as a plane view slices (for depth 6.39 to 53.36 km), for *NLCG*, *GN-MS* and *GN-DS* are shown in second, third and fourth panels. The rectangles indicate the position of the original conductive and resistive blocks. The last row of images shows the cross-section of the model at $x = 0$ profile.

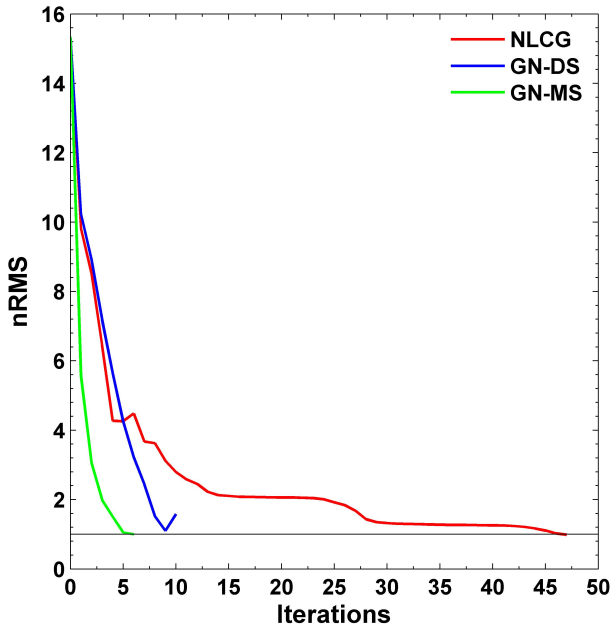

Figure 4.9: Convergence plot for all the three inversion algorithms. The *GN* requires less iterations due to its quadratic behaviour to converge to the desired error level. However, *GN-MS* requires less iterations as compared to *GN-DS*.

4.3.2 Synthetic Models

The *AP3DMT* code has been tested and validated over a variety of 3D standard models available in literature. For demonstration, we have selected two synthetic test models. The first test model is a 'checkerboard' conductivity distribution, **Rubick Model** [39] and the second is Dublin Test Model 2 (**DTM2**) [51, 102]. These two data sets, generated for these models were inverted using *NLCG* algorithm on HP Z620, Intel Xeon E2643, 32 GB RAM with 8 cores for parallelization. Other inversion details for each model is discussed in the the following.

Model 1

The model 1 consists of 10 and 1000 Ω-m blocks in a homogeneous 100 Ω-m half-space (Fig. 4.10a). These blocks are placed in a checkerboard pattern (alternate resistive and conductive). There are three layers of blocks termed as L1, L2 and L3 in the depth range $0-10$, $18-50$ and $58-106$ km respectively. Each layer has nine blocks. For comparison of inversion results, the synthetic data is taken from the test files provided along with the *ModEM* code [39]. This data set includes full impedance (all four complex components),

plus VTFs sampled at 225 sites distributed on a regular 2D array with 40 and 80 km spacing in $x-$ and $y-$ directions respectively for 12 periods, logarithmically spaced between $10 - 10000$ s. The data has 3% noise added to it and error floor set as 3% of $|\mathbf{Z}_{xy}\mathbf{Z}_{yx}|^{1/2}$ for impedance tensor and 0.03 for VTFs. The data was computed by discretized the model into $67 \times 67 \times 60$ cells (excluding cells in air), with nominal resolution of 20 km horizontally. For inversion a homogeneous model of 100 Ω-m, discretized into same number of cells, is used as apriori and initial model. In 58 *NLCG* iterations the nRMS error reduced from 12.38 to 1.06. The inverted model thus obtained is shown in Fig. 4.10b. All the conductive features are resolved. To analyse and study the inversion result ratio between model parameters [3]of the inverted models as obtained by *AP3DMT* and *ModEM* is shown in Fig. 4.11. Table 4.4 shows the ratio of model parameters in terms of percentage in various ratio range. Further, two set of 1D sections are drawn and compared with the true model. One set is along $z-$ direction (at three different $(x,\ y)$ points) while another is along $x-$ direction (at three different $(x,\ z)$ points) (Fig. 4.12 and 4.13). Also a histogram of inverted resistivity values for all model parameters lying in the volume occupied by all the resistive and conductive blocks in L1, L2 and L3 is drawn separately and shown in Fig. 4.14 and 4.15. Table 4.5 shows the information about resistivity values for true and inverted model along with minimum and maximum resistivity for the model parameters belonging to the volume occupied by each block. From Fig. $4.11 - 4.15$ and Table 4.5 following is inferred.

The inverted model obtained using *AP3DMT* is similar to that obtained using *ModEM*. Approximately 95% of the model parameters lies between $0.7 - 1.3$ ratio (ratio $r_i = {m_i}^a/{m_i}^d,\ i\ =\ 1,\ 2,\ ...\ N_p$, where ${m_i}^a$ and ${m_i}^d$ is the i^{th} model parameters of the inverted model obtained using *AP3DMT* and using *ModEM* respectively and N_p are the total number of model parameter inverted). The resistivity range reaches their true value for all the conductive blocks in L1, L2 and L3. From the histogram, (Fig. 4.14) it is clear that the resistive values for the cells corresponding to the conductive blocks cluster around their mean value. The top of all the conductive blocks in L1 and L2 are well defined however, this is not the case for blocks in layer L3. The MT data like all inductive EM techniques, is more sensitive to conductive structures than to resistive bodies and therefore, while the top of a conductor is well defined, its base is usually smeared out as seen for the blocks in L2 and L3. Among all the resistive blocks, only outcropping blocks are resolved. For

[3]model parameter in the code are defined as log of conductivity however, here and in subsequent discussion of results, model parameter is defined as resistivity of a cell of the model (either synthetic or inverted).

majority of model parameters corresponding to resistive blocks of L1, the resistivity lies in the range $350-450$ Ω-m. For model parameters corresponding to resistive blocks of L2 and L3, the resistivity hardly exceeds 200 Ω-m (Fig. 4.15). This is also corroborated by the 1D sections (Fig. 4.12 and 4.13). As seen from the histogram in Fig. 4.15 the resistivity of the model parameters for the blocks in L2 and L3 cluster around 100 Ω-m (background resistivity value).

The main model features present in the test model are recovered. As expected, there is degradation in the features below the conductive structures. To study the misfit, component and period wise misfits of the data and the predicted data (off diagonal impedances) for the inverse model is shown in Fig. 4.16 and 4.17. For the sites that lies on the edge (along $y-$ direction) of the conductive block (Fig. 4.16) there is mismatch between the observed and the computed responses especially in the phase Z_{yx}. There is good agreement between the responses for the sites that lies within the boundary of the block under consideration. For the sites that lies within the boundary of the resistive block, there is mismatch in phases, both Z_{xy} and Z_{yx} for period less than 100 s.

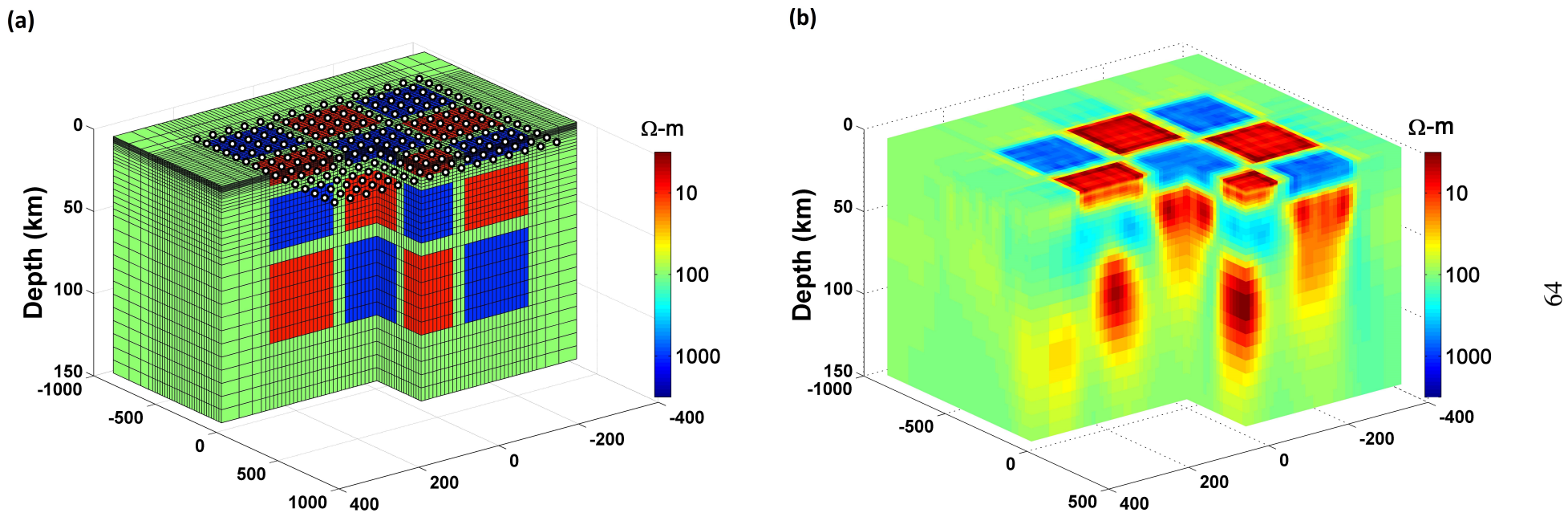

Figure 4.10: (a) Resistivity model for 3D test, with site locations shown at top Egbert and Kelbert [39], and (b) inverse solution obtained with *AP3DMT* after 58 iterations by fitting full impedances and vertical field TFs. In the cut-away view the upper surface shown is at 2 km depth, but the structures shown extend to the surface.

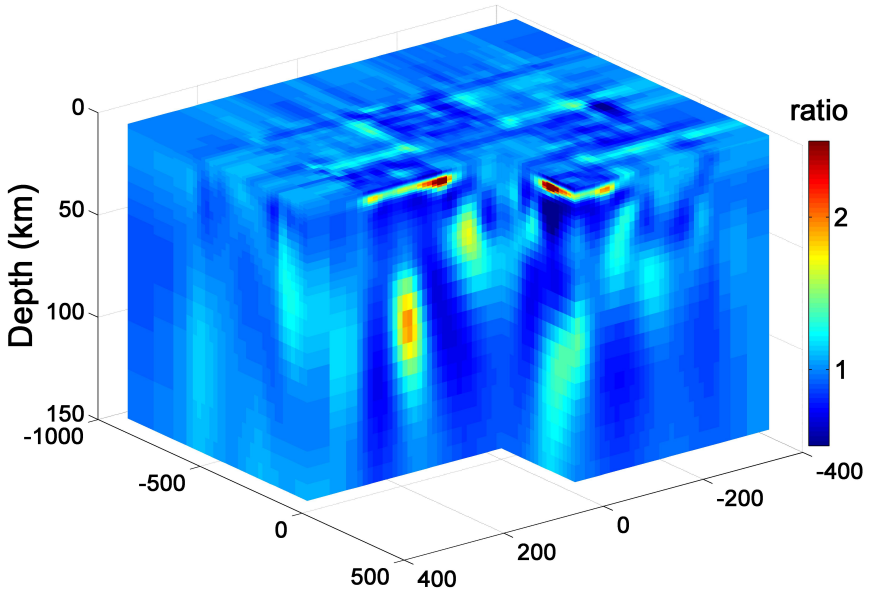

Figure 4.11: Ratio between model parameters of the inverse models as obtained by *AP3DMT* and *ModEM*.

Table 4.4: Table showing information about ratio between model parameters (resistivity value) of the inverted models as obtained by *AP3DMT* and *ModEM*.

Ratio Range	**Model Parameter (%)**
< 0.7	0.94
$0.7 - 0.9$	10.95
$0.9 - 1.1$	67.84
$1.1 - 1.3$	17.02
$1.3 - 1.5$	1.74
> 1.5	1.51

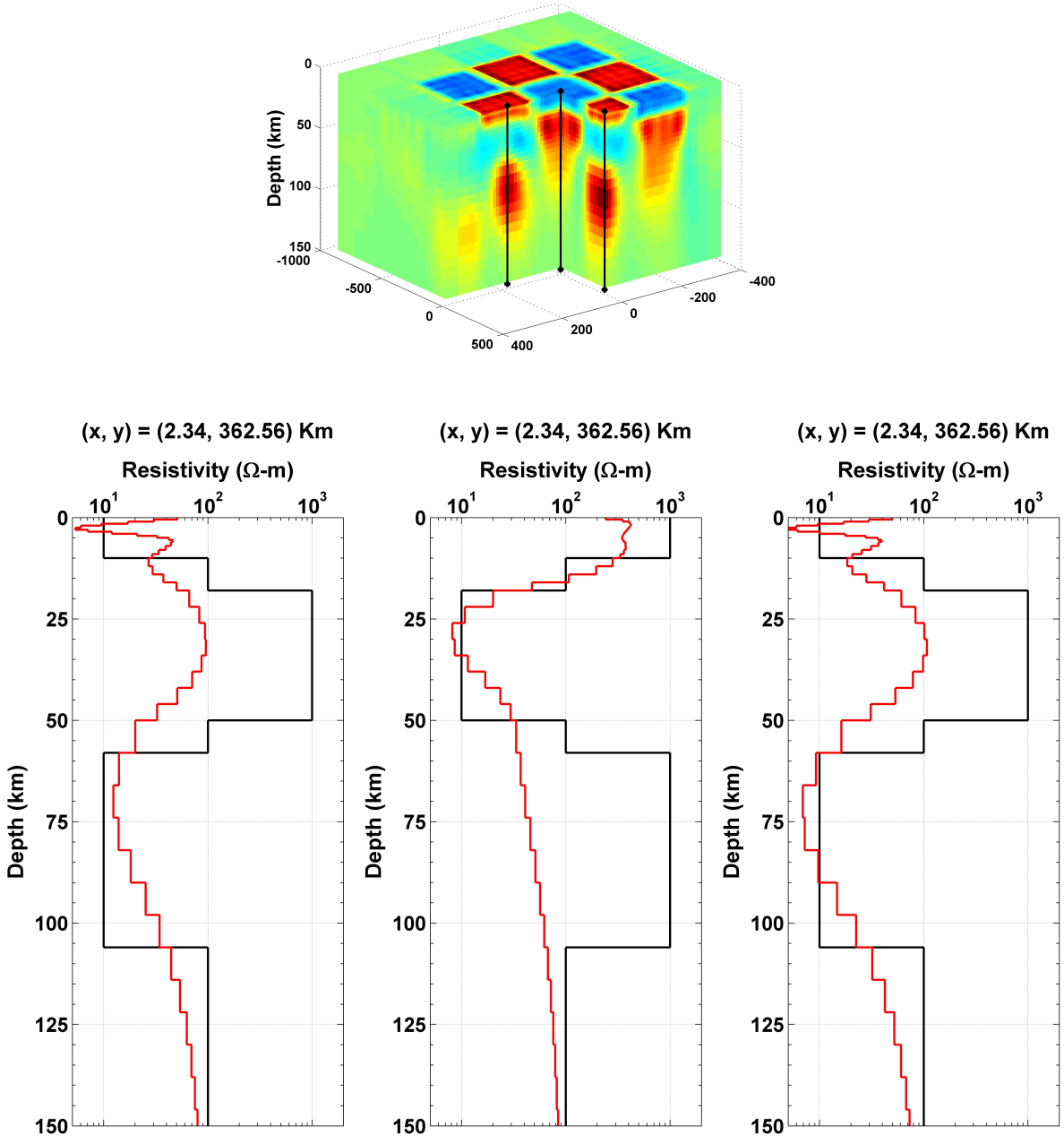

Figure 4.12: 1D section of True Model (black), and inverted model (red) at three different $(x,\ y)$ points.

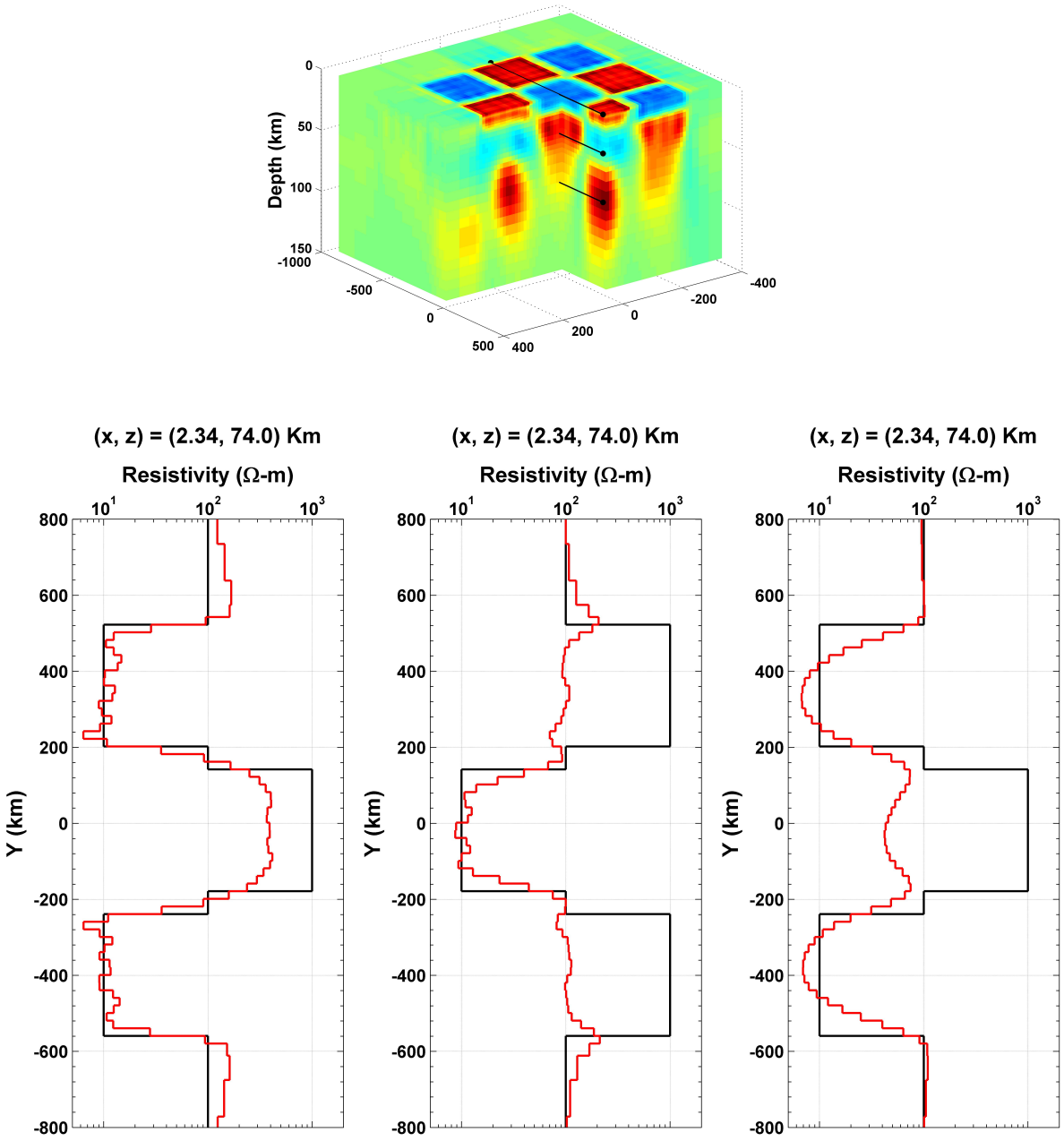

Figure 4.13: 1D section of True Model (black), and inverted model (red) at three different (x, z) points. This section passes through all the three different layers of anomalies

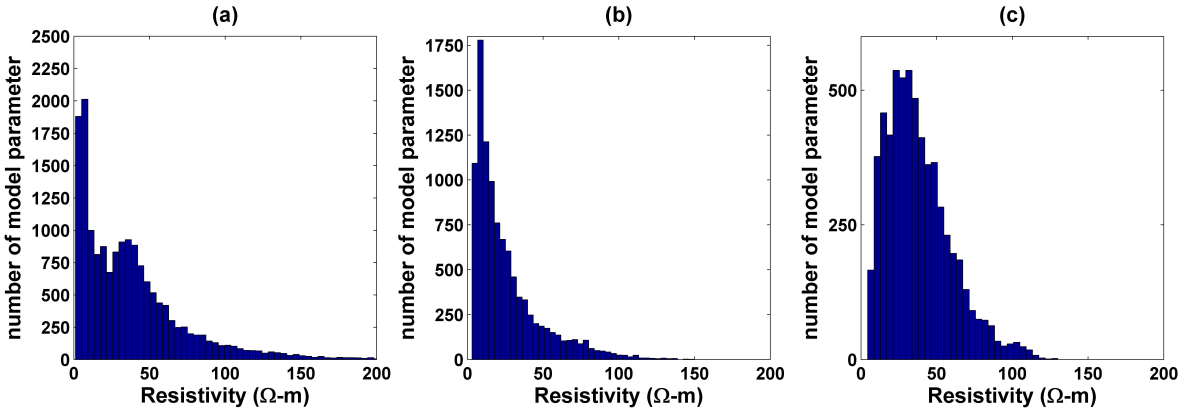

Figure 4.14: Histogram of model parameters lying in the volume occupied by all the conductive blocks in (a) L1, (b) L2 and (c) L3.

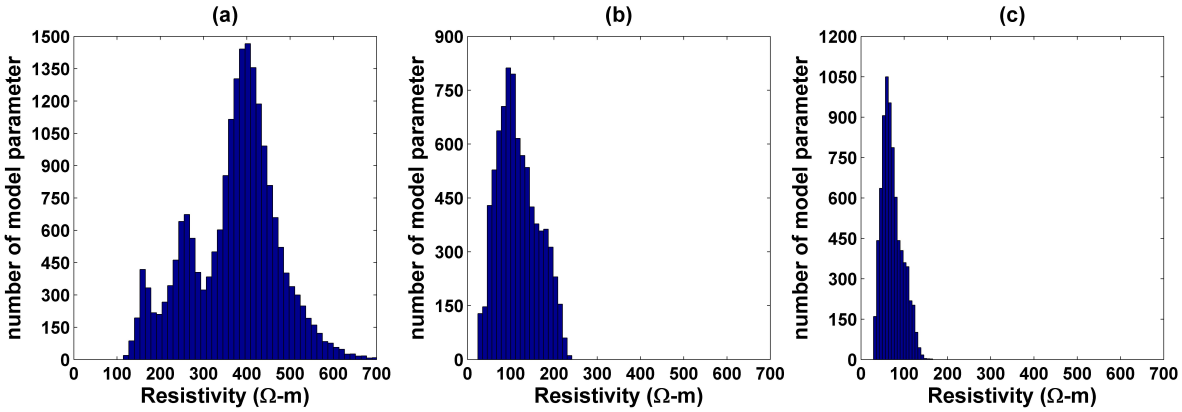

Figure 4.15: Histogram of model parameters lying in the volume occupied by all the resistive blocks in (a) L1, (b) L2 and (c) L3.

Table 4.5: Table showing information about resistivity values for true and inverted model. For each block minimum and maximum resistivity value is given along with the percentage of model parameters that lies less than 30 Ω-m for conductive blocks and greater than 400 Ω-m for the resistive blocks. Block no. $1-9$ lies in L1, $10-18$ lies in L2 and $19-27$ lies in L3.

Block No	**True Model** Avg. Res. (Ω-m)	**Inverted Model** Avg. Res.(Ω-m)	**Minimum** (Ω-m)	**Maximum** (Ω-m)	**No. of Cells** (%)
Block 1	1000.0	382.2	120.2	753.7	46.3
Block 2	10.0	35.6	1.3	309.1	50.5
Block 3	1000.0	379.6	121.2	692.6	48.8
Block 4	10.0	40.5	1.1	305.8	49.5
Block 5	1000.0	344.4	114.5	623.2	26.3
Block 6	10.0	40.2	1.1	287.4	49.9
Block 7	1000.0	376.1	115.4	699.9	45.8
Block 8	10.0	37.8	1.3	307.3	48.5
Block 9	1000.0	374.1	119.1	729.6	43.6
Block 10	10.0	24.8	3.1	132.3	73.0
Block 11	1000.0	118.5	30.1	236.6	–
Block 12	10.0	25.1	3.2	136.7	72.5
Block 13	1000.0	117.8	24.0	236.1	–
Block 14	10.0	29.2	3.5	144.5	65.5
Block 15	1000.0	114.9	24.5	241.4	–
Block 16	10.0	24.9	3.4	135.5	71.9
Block 17	1000.0	115.2	27.3	227.7	–
Block 18	10.0	25.0	3.6	149.7	72.3
Block 19	1000.0	76.6	29.1	139.6	–
Block 20	10.0	37.0	8.8	122.0	42.6
Block 21	1000.0	76.3	32.0	147.6	–
Block 22	10.0	39.0	4.5	130.1	42.0
Block 23	1000.0	59.7	30.0	123.6	–
Block 24	10.0	39.7	5.2	113.4	39.9
Block 25	1000.0	73.5	35.6	144.8	–
Block 26	10.0	37.0	8.9	115.1	40.6
Block 27	1000.0	75.5	30.4	166.0	–

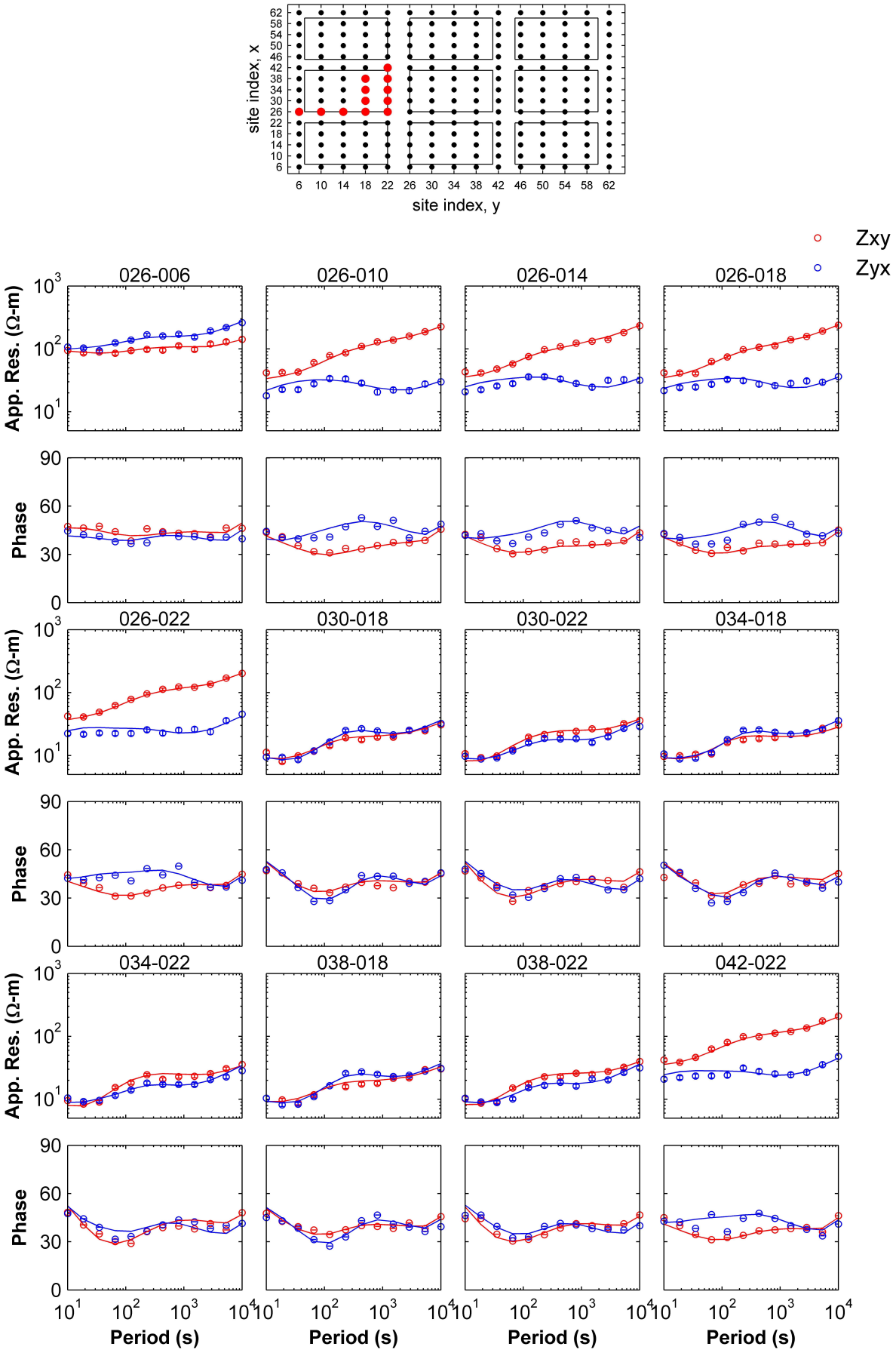

Figure 4.16: Comparison of forward versus inversion responses (apparent resistivity and phases). For the example sites (marked in red colour), the data of the inversion (lines) are shown in comparison to the forward responses (circles). The sites lies inside and on the edge of the conductive block and the site name are formed by the combination of indexes in $x-$ and $y-$ direction.

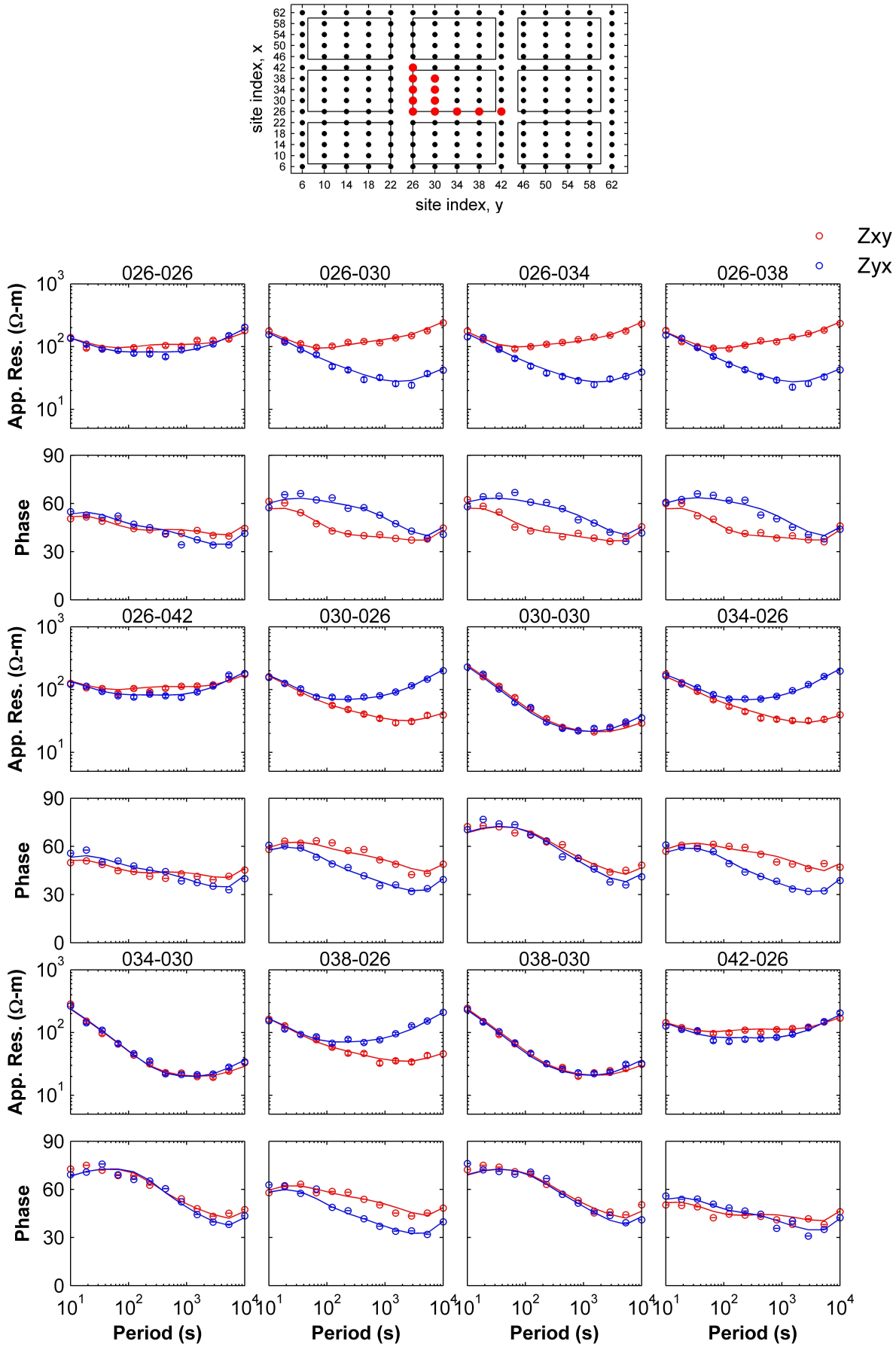

Figure 4.17: Comparison of forward versus inversion responses (apparent resistivity and phases). For the example sites (marked in red colour), the data of the inversion (lines) are shown in comparison to the forward responses (circles). The sites lies inside and on the edge of the resistive block and the site name are formed by the combination of indexes in $x-$ and $y-$ direction.

Model 2

The second model **DTM2**, Fig. 4.18 (left panel), consists of a hemisphere of 10 Ω-m and radius of 5 km directly beneath the surface of a homogeneous 300 Ω-m half space. The origin of the coordinate system has been taken as the center of the hemisphere. For proper representation of the target body, the model is discretized into 76 × 76 × 39 cells with horizontal cell dimensions of 250 m. Cell thicknesses are 50, 100 and 250 m in the upper 5 km of the model and then increase logarithmically downwards. *ModEM* is used to compute the impedance tensor for 11 periods logarithmically spaced between $0.01 - 1000$ s at 49 sites, located on circles of different radii from the center of the hemisphere except one at the center. 5% gaussian noise was added to the data. An error floor of 5% of $|\mathbf{Z}_{xy}\mathbf{Z}_{yx}|^{1/2}$ is assigned to the impedance tensor. For inversion, again a homogeneous model of 100 Ω-m, discretized into 54 × 54 × 37 cells (500 m × 500 m × 50 m, vertical extent increasing with depth) is used as apriori and initial model. In 47 *NLCG* iterations the nRMS error reduced from 20.03 to 0.98. The inverted model thus obtained is shown in Fig. 4.18 (right panel). The circular shape of the hemisphere and the correct order of resistivity were recovered however, the deeper part of the structure is not delineated properly as it is shielded by the shallower conductive structure. The correct representation of the circular body is limited due to the use of rectangular meshing. The nRMS error reaches less than 1 however, it will be more useful to study the fit of responses for different sites and different periods. Fig. 4.19 illustrate the misfits of the observed data and the predicted data for the inverse model shown in Fig. 4.18 (right panel). These misfits are for the sites lying on the line $x = y$. Site number 000 lies at the center of the body and sites 002, 010, 018, 026 and 034 in the first quadrant (increasing towards positive y) while other sites are in the third quadrant. For sites that lie away from the edge of the hemisphere, there is an excellent agreement between the observed and the computed responses (for e.g., 046, 038, 000 etc.). For sites that are close to the edge of the hemisphere (for e.g., 010 and 014) there is mismatch in the responses (Fig. 4.19). The reason why the responses differ is the different meshes used for computing synthetic data and for inversion. In general, the data of site 18 (outside the hemisphere) is fit better than for site 10 (inside the hemisphere) particularly phases (Fig. 4.20). The results fit the longer periods well and have a shortcoming on the shorter periods. This also suggest that a single nRMS value is not a satisfying way to represent data fit. For more discussion on this model, one can refer Miensopust et al. [102].

Table 4.6 summarizes the results of these two models along with the run time and other details. For both these models and for the other models tested, the edges of the target bodies are not represented by a sharp boundary as in the true model. The resistivity gradually changes across the boundary of a structure resulting in smearing out of structures. This is so because the current code is based on smooth model approach.

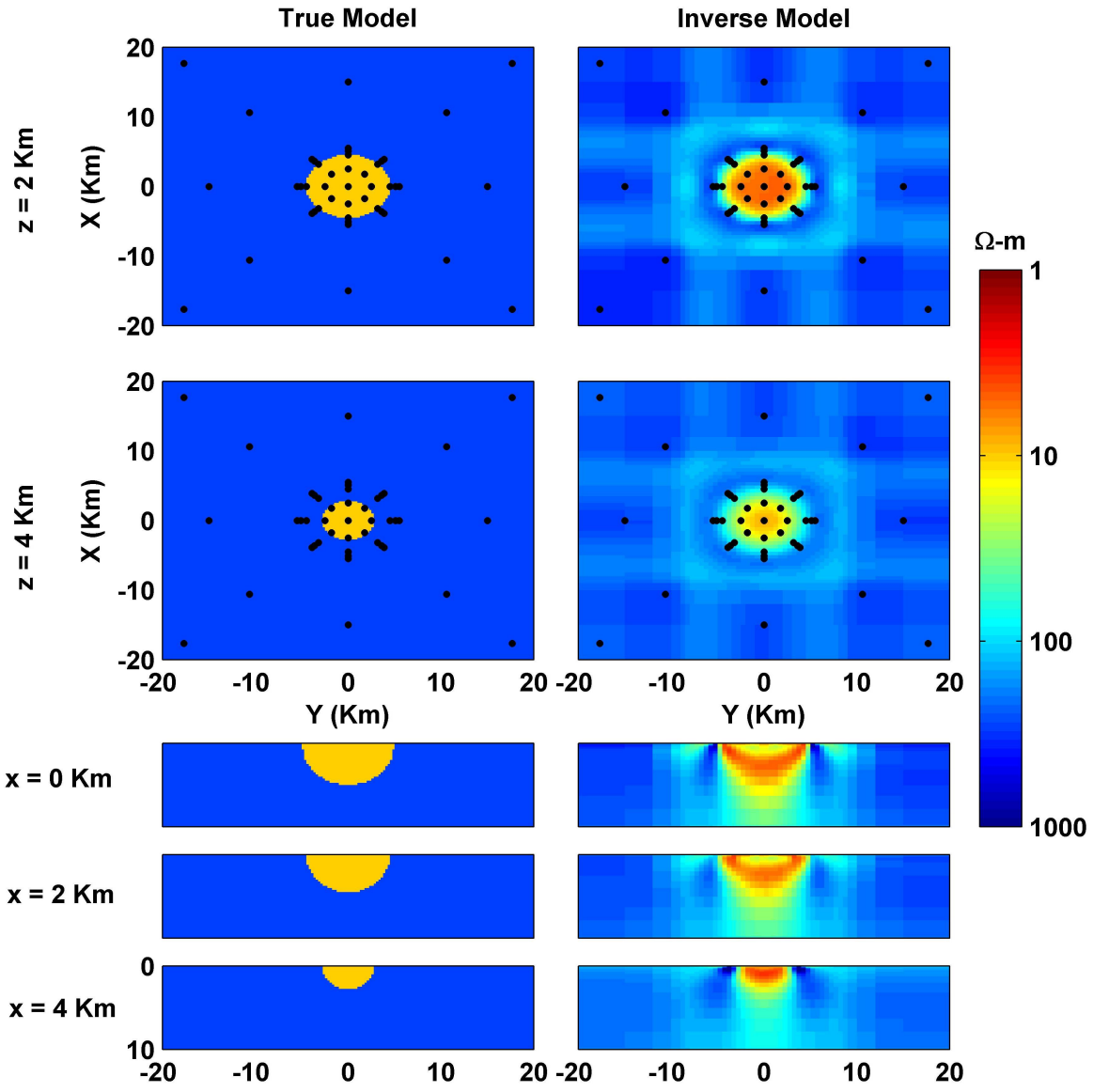

Figure 4.18: A plan view of true model (left panel) and the inverted model (right panel). The black dots represent the used sites.

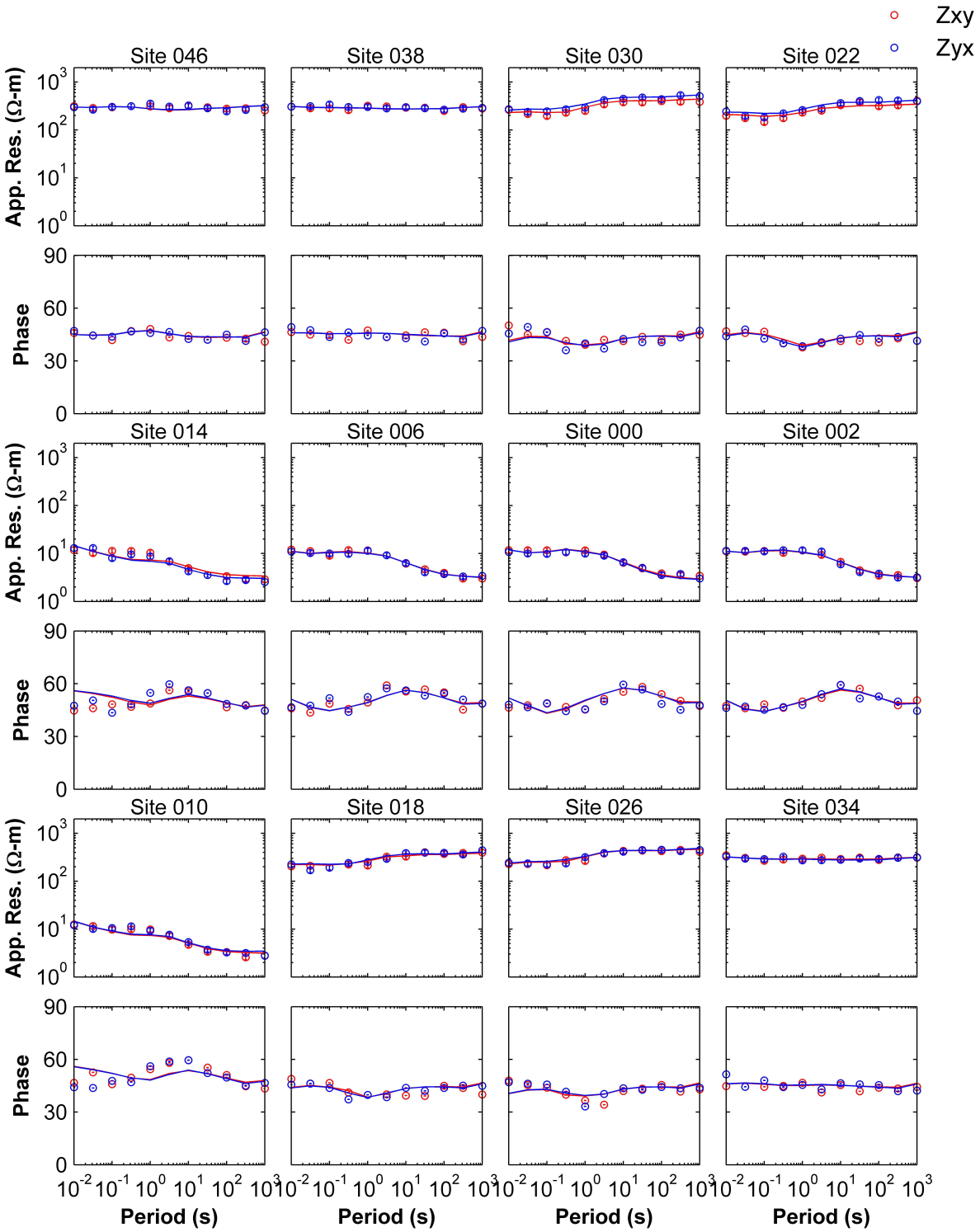

Figure 4.19: DTM2 comparison of forward versus inversion responses (apparent resistivity and phases). For the 12 sites ($x = y$), the data of the inversion (lines) are shown in comparison to the forward responses (circles).

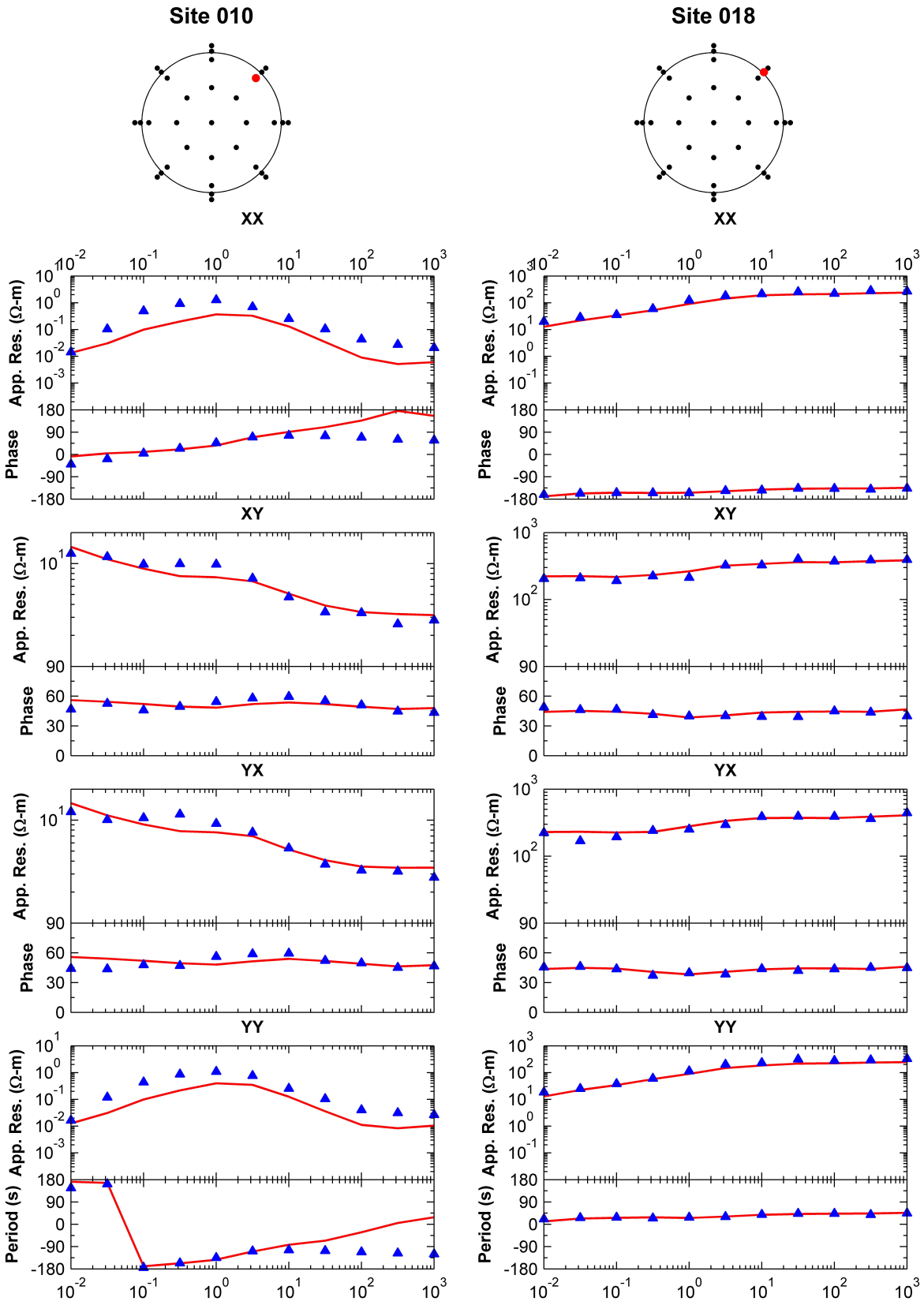

Figure 4.20: DTM2 comparison of forward versus inversion responses (apparent resistivity and phases). For the example sites 010 (left) and 018 (right), the data of the inversion (red) are shown in comparison to the forward responses (blue).

Table 4.6: List of inversion results. Beside the model name, information about the data, mesh and inversion parameters are given as well as the number of iterations, RMS value (initial/final) and target tolerance (TT).

Model Name	*Info*	*Data*	Mesh
Rubick	58 iterations nRMS error 12.38/1.06 TT = 10^{-6} (forward & gradient) 49.25 hrs, 9.9 GB (8 cores) HP Z620, Intel Xeon E2643 3.30 GHz, 32 GB	225 sites Impedance & VTF 12 periods ($10 - 10000$ sec) Error floor 3% of $\lvert\mathbf{Z}_{xy}\mathbf{Z}_{yx}\rvert^{1/2}$ for $\mathbf{Z}$ and 0.03 for VTF	$67 \times 67 \times 60$ cells (same as for forward) $\sim$9939 m $\times$ $\sim$20011 m $\times$ $\sim$500 m centre cell starting model: 100 Ω-m half-space
DTM2	42 iterations nRMS error 20.03/0.98 TT = 10^{-6} (forward & gradient) 12.26 hrs, 5.8 GB (8 cores) HP Z620, Intel Xeon E2643 3.30 GHz, 32 GB	49 sites Impedance 11 periods ($0.01 - 1000$ sec) Error floor 5% of $\lvert\mathbf{Z}_{xy}\mathbf{Z}_{yx}\rvert^{1/2}$ for $\mathbf{Z}$	$76 \times 76 \times 39$ cells (forward) $54 \times 54 \times 37$ cells (inversion) $\sim$500 m $\times$ $\sim$500 m $\times$ $\sim$50 m centre cell (vertical extent increasing with depth) starting model: 100 Ω-m half-space

4.3.3 Validation on Field Data from Garhwal Himalaya

4.3.3.1 Regional Geology

The Himalaya is one of the youngest and highest mountain range, which originated from continental collision tectonics and underthrusting of the Indian Plate beneath the Eurasian Plate. Regional N-S compression, resulting from horizontal movement of rock masses along the north dipping thrust planes, caused crustal shortening, horizontal extrusion and lithospheric delamination [79, 105]. In this process, leading upper brittle portion of the subducting Indian crust has been sliced and stacked up southwards to form the Himalayan mountain belt. The Main Frontal Thrust (MFT) is the southernmost thrust in the study area, to its north is the Main Boundary Thrust (MBT) which hades northwards [74, 73] at 30° to 40°. In the study area, the MCT zone is bounded by the Munsiari Thrust (MT) in the south and Vaikrita Thrust (VT) in the north [18, 19, 93, 184]. The northernmost region of the profile is the Higher Himalayan (HH) crystalline zone containing most of the famous peaks of the mountain range and it has an average elevation of 4500 m. In addition to these major Himalayan thrusts, a number of faults, ridges and other structural features having strike oblique and transverse to the main Himalayan thrusts are mentioned in the literature [46, 74, 184]. To map the geometry of various faults and thrusts system magnetotelluric method has been used along a few transects Himalayan collision zone [47, 61, 83]. These studies are mainly based on 1D or 2D inversion approximation of realistic 3D complex geological model. In presence of strong 3D features, 1D/2D inversion may add significant error in realistic 3D model. At the same time for conducting 2D inversion, locations of MT sites need be projected along a profile line. To overcome this approximation and error, 3D inversion must be performed.

4.3.3.2 Description of MT data

Broadband MT survey was conducted in the Garhwal Himalayan Corridor along Roorkee-Gangotri (RG) profile passing through the major Himalayan thrusts zones. The MT system used in this survey was Metronix system, with ADU-06 data logger, MFS-06 induction coil and EFP-06 electrodes. The recorded time domain data were transformed to the frequency domain impedance tensor using the magnetotelluric processing code, MAPROS [45]. Subsequently, the data were reprocessed using a robust noise-suppressing

code [162]. The stable impedance tensor was thus estimated for 35 sites in the period range of 0.01 to 2048 s. Since the induction coil for vertical magnetic field component couldn't be buried in rocky sites in tough Himalayan terrain, the estimated tipper values were noisy at such MT sites, and reliable tipper responses were available only at 19 sites in the period range from 22 − 2048 s. Locations of impedance and tipper sites are shown in Fig. 4.21. However, for demonstrating the performance of ur code, we have inverted full impedance only. Topography variations are not taken into consideration as Kumar and Manglik [78] demonstrated through a 2D study that there is no effect of Himalayan topography on interpretation of MT data.

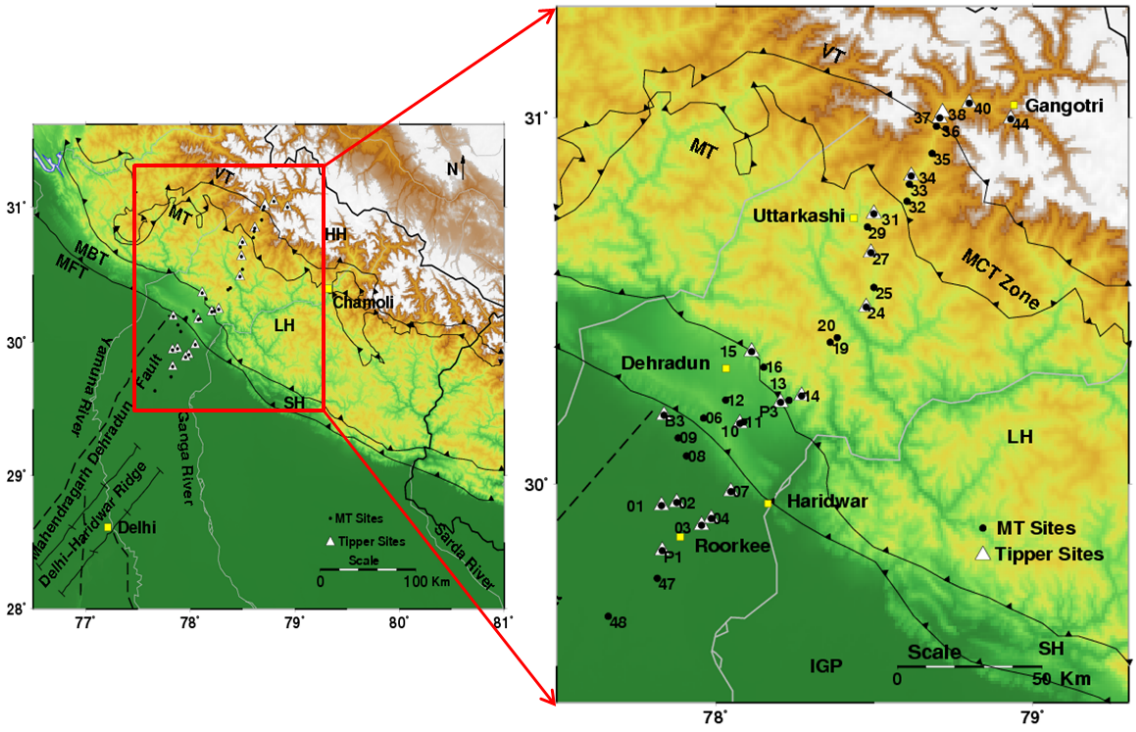

Figure 4.21: Simplified tectonic map of the study area and MT sites locations (compiled from Mahesh et al. [93] and Valdiya [185]).

4.3.3.3 Inversion of MT data

For the 35 sites of full impedance tensor (Z_{xx}, Z_{xy}, Z_{yx}, Z_{yy}) data, the 16 periods, lying in the period range of 0.015 to 512 s, were selected. The data error floor was set to 10% of $|\mathbf{Z}_{xy}|$ for $|\mathbf{Z}_{xx}|$ and $|\mathbf{Z}_{xy}|$ and 10% of $|\mathbf{Z}_{yx}|$ for $|\mathbf{Z}_{yx}|$ and $|\mathbf{Z}_{yy}|$ respectively.

The 3D guess model, a homogeneous half space of 100 Ω-m, was discretized into 83 × 90 × 42 cells (with 10 cells in air). Below the surface, the top layer thickness was 50 m and the thickness of each subsequent layer increased by a factor of 1.2, extending up to 500

km. Horizontal grid spacing in $x-$ and $y-$ direction was 3 km and 3 km respectively in the central zone of the model domain. Total of 24 cells were padded around the central region, six in each horizontal direction with increasing distance. With this grid, the dimension of the model domain has become $800 \times 800 \times 500$ km in $x-$, $y-$ and $z-$ directions, respectively.

From the starting misfit of 7.21 the nRMS error reduced to 1.79 in 117 *NLCG* iterations using *ModEM* whereas it reduced from 7.21 to 1.78 in 120 iterations using *AP3DMT*. Fig. 4.22 and 4.23 depicts the inversion results obtained using two different inversion codes (*ModEM* and *AP3DMT*) at different depth planes. Fig. 4.24 shows the ratio between model parameters of the inverted models as obtained by these inversion codes. Fig. 4.25 shows the comparison of apparent resistivity and phase of inverted models for representative sites in different lithotectonic domain along the profile. From the Fig. 4.22 $-$ 4.25 it is observed that the inverted model obtained using *AP3DMT* closely matches the model obtained using *ModEM*.

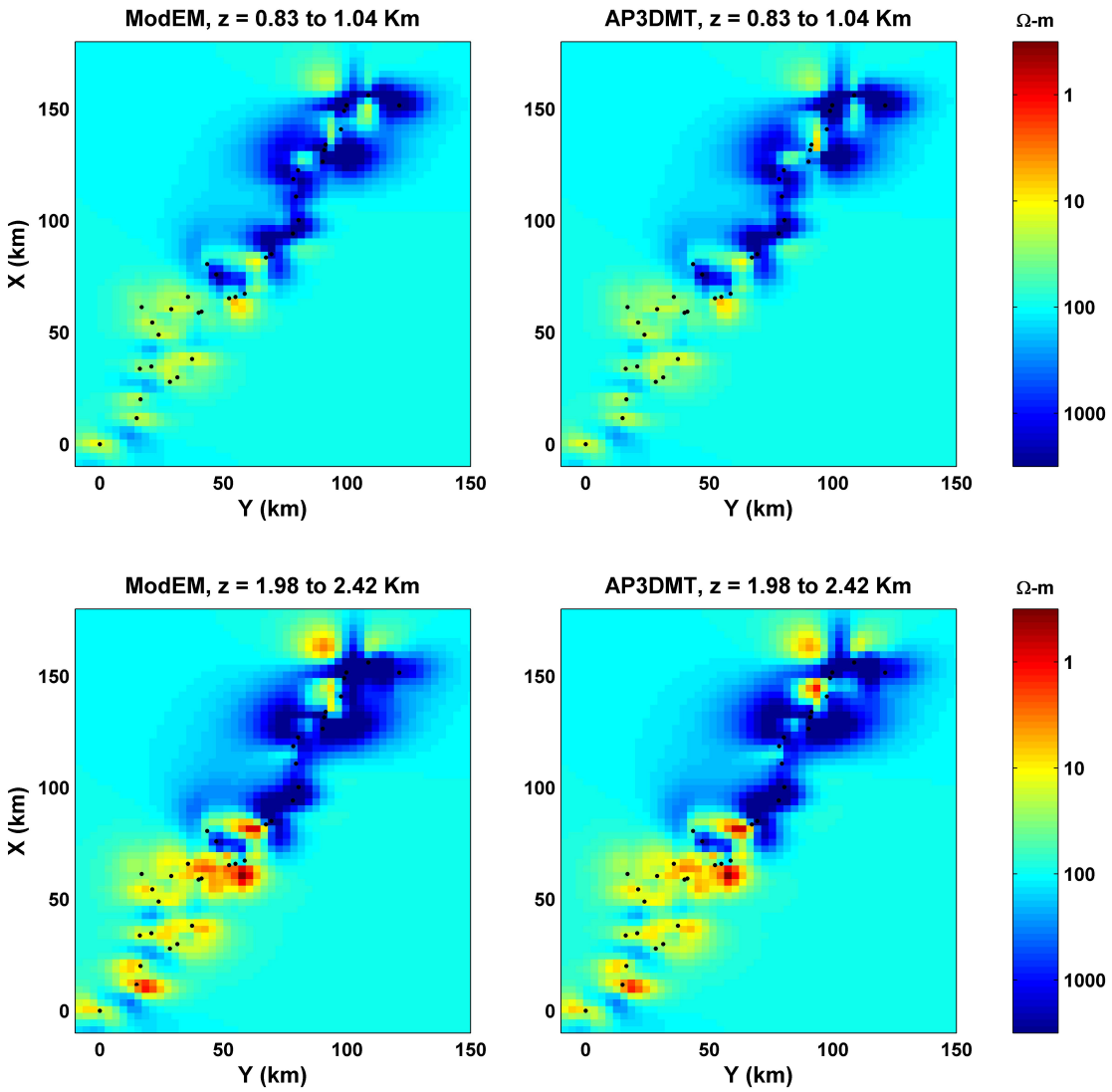

Figure 4.22: Depth plane slices of 3D inverted model obtained using full impedance using *ModEM* and *AP3DMT*.

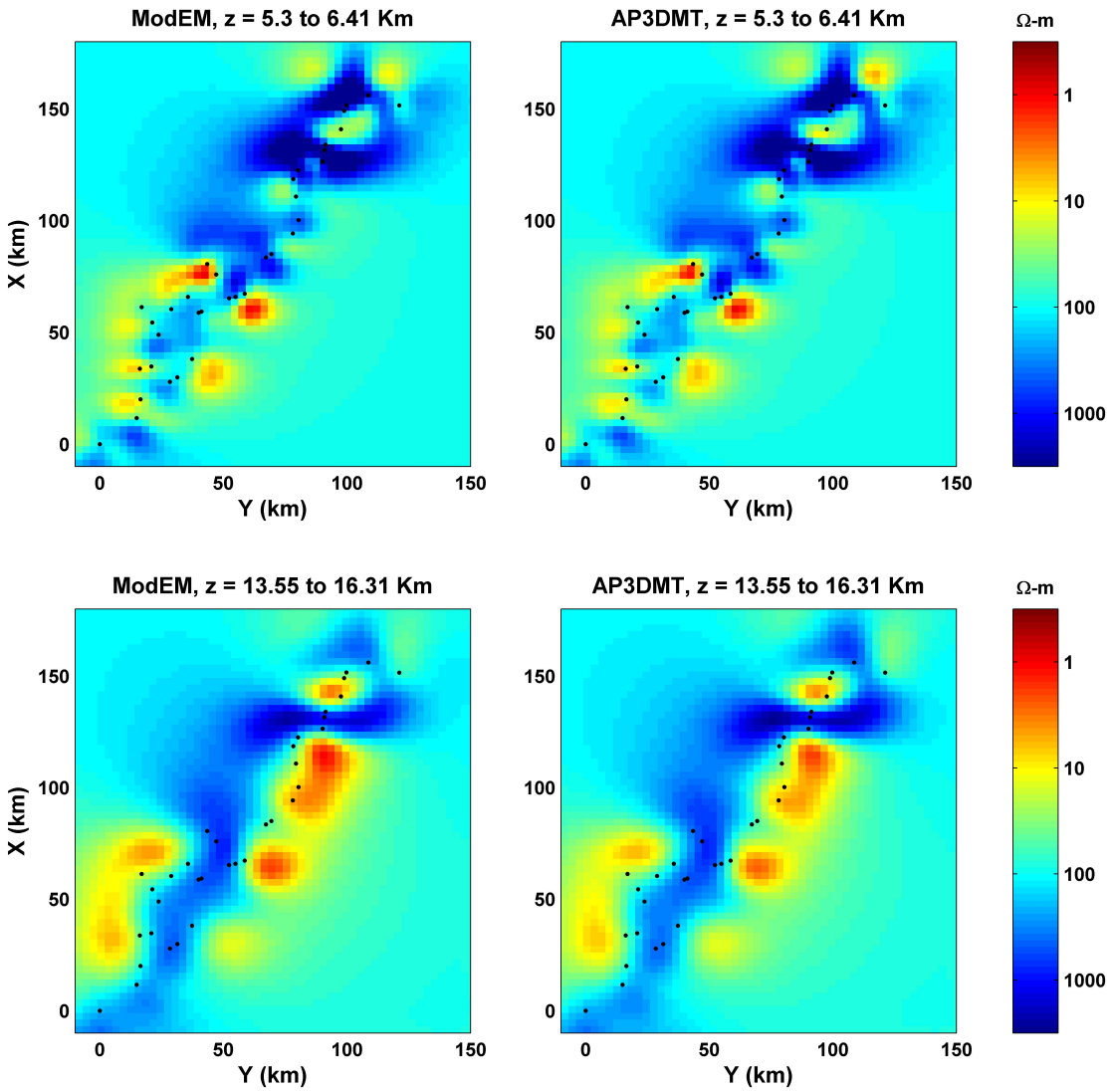

Figure 4.23: Depth plane slices of 3D inverted model obtained using full impedance using *ModEM* and *AP3DMT*.

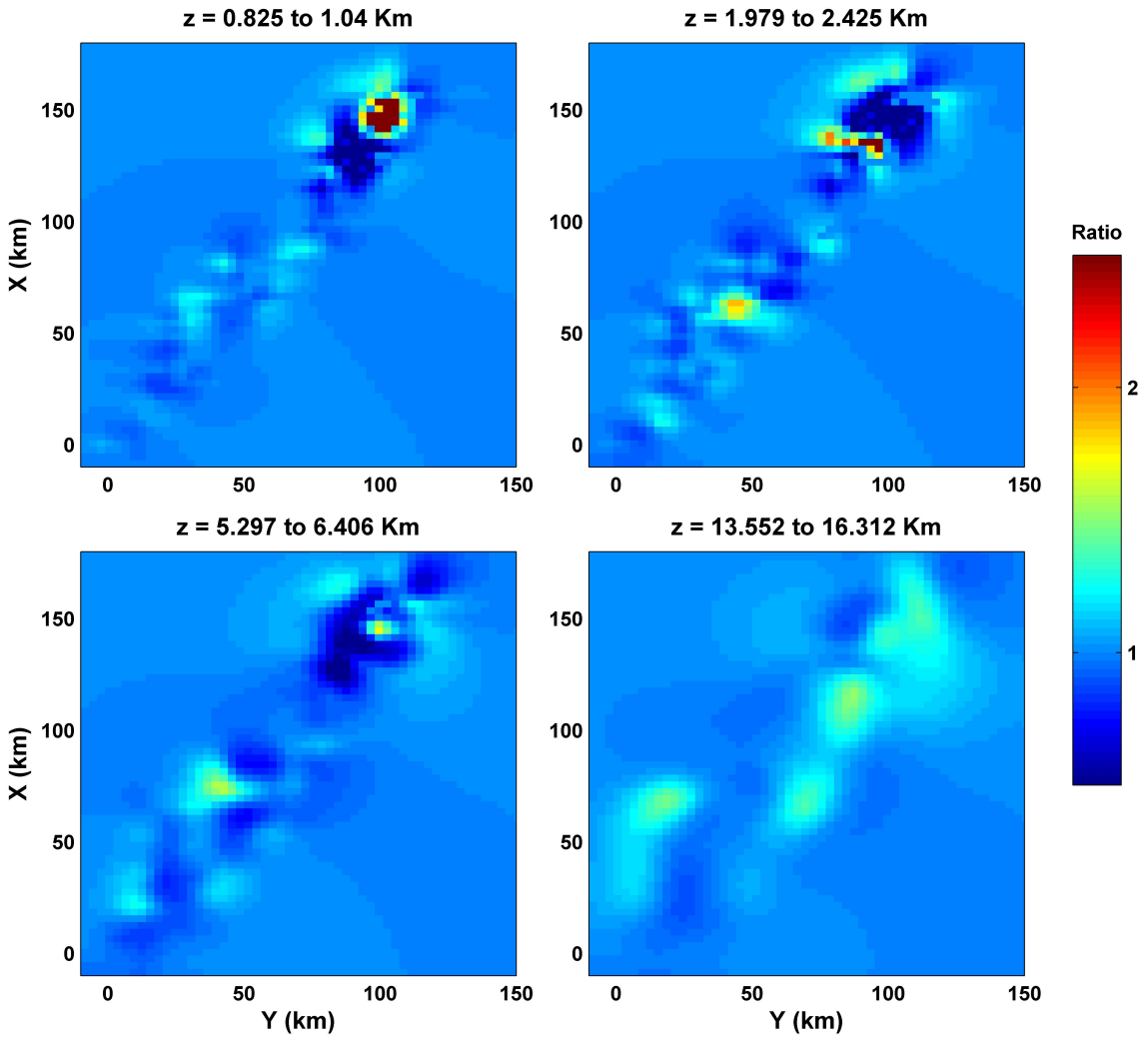

Figure 4.24: Ratio between model parameters (resistivity of cells) of the inverse models as obtained by *AP3DMT* and *ModEM*.

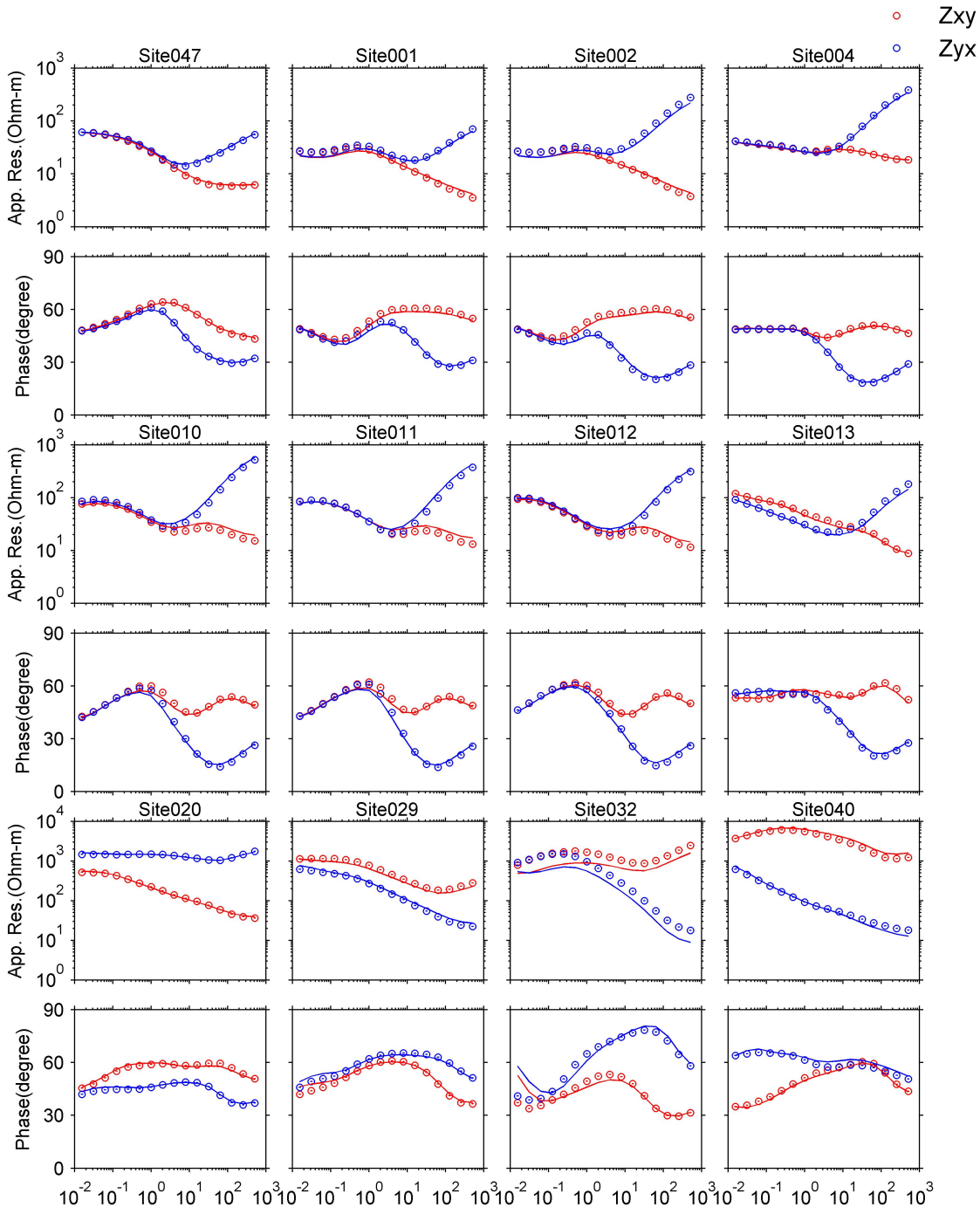

Figure 4.25: Comparison of apparent resistivity and phase of inverted models obtained using by *AP3DMT* and *ModEM* for representative sites in different lithotectonic domain along the profile. Red and blue circles represents the data for *ModEM* whereas red and blue lines represents the data for *AP3DMT*.

4.4 Summary

The accuracy and versatility of the developed inversion algorithm is demonstrated through two synthetic models. The inverted models obtained using *AP3DMT* closely matches the inverted models published in the literature. For these and others models (not discussed here) it is difficult to reproduce the resistivity contrasts of the true model as the code is based on the smooth inversion approach. Also, the nRMS value reaches close to 1 but some periods and components are fitted better than others (for e.g., site 010 and 018 for DTM2). This is further evident from the misfit of site no 032 and 036 for real data. Thus, one should not just rely on single nRMS value but more carefully examine the misfit over all data space.

The 3D inversion of Roorkee-Gangotri field MT data is performed and comparison of inverted model with the model obtained using *ModEM* is presented. From the inverted models and comparison of apparent resistivity and phase for the inverted models we demonstrate robustness of the algorithm for real field data.

Chapter 5

3D DCR INVERSION

5.1 Preamble

Normally the Direct Current Resistivity (DCR) field data are interpreted using the assumption of simple models such as 1D or 2D structures. However, in the presence of 3D structures, 1D or 2D approaches are not correct near the boundaries of 3D bodies. Thus, the use of 3D model is essential for meaningful and comprehensive interpretation for complex geological situations which are truly 3D in nature. Resistivity data were interpreted using 3D modeling and inversion techniques developed by various authors [e.g. 40, 52, 87, 118, 126, 139, 148]. In these studies, the forward solutions are generally obtained by IEM, FDM or FEM and inversion are based on Gauss–Newton (or its variant) technique. For the 3D case the problem is large hence, explicit computation and storage of the Jacobian (sensitivity) matrix is avoided [e.g. 204].

In this chapter, first I will briefly discuss 3D DCR theory and the governing partial differential equation (PDE). Next, various aspect of forward modeling like model discretization, boundary conditions, matrix solver (both iterative and direct) will be discussed in detail. The necessary modifications made in *AP3DMT*, to incorporate DCR inversion will be discussed in the detail. Finally validation of the code for DCR case, both forward and inverse modeling will be demonstrated over two synthetic models.

5.2 Formulation of 3D DCR problem

Let us consider a half space with arbitrary conductivity $\sigma(x,y,z)$ distribution. The electric field, **E**, can be written as,

$$\mathbf{E} = -\nabla\phi, \tag{5.2.1}$$

where, ϕ is the electric potential. For current density, j, and source term, Q, the equation of continuity is defined as,

$$\nabla \cdot j = Q, \tag{5.2.2}$$

with the source term,

$$Q = I\delta(x-x_q)\delta(y-y_q)\delta(z-z_q) \tag{5.2.3}$$

with I as the current and δ is the Dirac delta function. Using $j = \sigma E$, we get,

$$\nabla \cdot [\sigma(x,y,z)\nabla\phi(x,y,z)] = -I\delta(x-x_q)\delta(y-y_q)\delta(z-z_q). \tag{5.2.4}$$

The eq 5.2.4 is the governing differential equation and relates the potential field to the input current source. The analytical solutions for the above equation are available for selected class of simple models. Hence, for a general conductivity distribution the equation is solved numerically, using FD, FE or FV approximation.

5.3 Forward Modeling Scheme

This section describes how the numerical solutions of the governing PDE (eq 5.2.4) are computed using FD method using primary and secondary field formulation.

5.3.1 Primary and Secondary Field Formulation

In the vicinity of source or at source, numerical approximations using eq 5.2.4 typically give poor results because of steep gradient of field. The popular approach is to remove the effect of the singular potential by using primary/secondary field formulation [88]. In this approach, the potential ϕ in eq 5.2.4 is considered as the superposition of the primary potential, ϕ_p, and

secondary potential, ϕ_s, as,

$$\phi(x,y,z) = \phi_p(x,y,z) + \phi_s(x,y,z). \tag{5.3.1}$$

The primary potential are caused by the current source for simple model like half space or layered model having conductivity σ_p. If σ_a represents residual conductivity of the medium such that the total conductivity, σ, can be written as,

$$\sigma = \sigma_p + \sigma_a. \tag{5.3.2}$$

Using eq 5.3.1 and 5.3.2 the eq 5.2.4 is decomposed into two equations as,

$$\nabla \cdot [\sigma_p(x,y,z)\nabla\phi_p(x,y,z)] = -I\delta(x-x_q)\delta(y-y_q)\delta(z-z_q). \tag{5.3.3}$$

$$\nabla \cdot [\sigma(x,y,z)\nabla\phi_s(x,y,z)] + \nabla \cdot [\sigma_a(x,y,z)\nabla\phi_p(x,y,z)] = 0. \tag{5.3.4}$$

Now, the forward problem is to be solved in two steps. First eq 5.3.3 is solved for primary potentials and later eq 5.3.4 is solved to compute the secondary potentials due to scattered source. Finally, total potentials are computed as superposition of primary and secondary potentials.

Primary Potential

The primary model should be chosen such that the simulation of primary potential is easy and straightforward and the secondary (scattered) source vanishes in close proximity of primary source. The eq 5.3.3 is sloved for primary potentail for a background model such that σ_p at the source position is same as σ. This means σ_a should be zero around the source point. For a uniform half-space and a current source located at $(x_q,\ y_q,\ 0)$, the potential at $(x,\ y,\ z)$ is given as,

$$\phi_p = \frac{I}{2\pi\sigma_p\sqrt{(x-x_q)^2+(y-y_q)^2+z^2}}, \tag{5.3.5}$$

where, while σ_p is the conductivity of the media at the source point [205] and not the mean conductivity as suggested by Lowry et al. [88]. If the background model is layered earth then, the primary potential is assumed to be caused by the current source in a layered earth. The potential of a vertical contact can be calculated very easily using the method of images.

The recurrence formulae for all kernel functions for a layered earth are given by Yuguo and Spitzer [203]. In the following, we shall discuss numerical solutions of eq 5.3.4.

5.3.2 Numerical Solution of Secondary Potentials

To obtain the numerical solution for arbitrarily shaped 3D structures using FD method first step is discretization of model. For accurate solutions this a very crucial step. Dey and Morrison [33] used four nodes between two adjacent electrodes, while Park and Van [118] used only one node. To strike a balance between number of nodes and errors in the calculated potentials Loke and Barker [87] used two nodes. In this study, we have used two nodes between adjacent electrodes however, this choice will depend on the availability of computational resources. For vertical discretization, the first vertical layer thickness is fixed at around one-fourth of the minimum electrode spacing. The thickness of the subsequent layers is then increased at a rate 1.2–1.5 times that of the previous one. Towards the side and bottom of the mesh, the spacing between two grids is progressively increased to simulate the 'infinitely distance' edges of the model. After model discretization, unknown potential are assigned on a normal grid. The discrete FD equations using normal grid are given in Appendix C.

The forward problem is solved using the appropriate boundary conditions. The total potential ϕ and the normal component of the current density $j_n = \sigma \partial\phi/\partial n$ must be continuous at boundaries. At the air-earth interface there is no current flow hence,

$$\frac{\partial \phi_s}{\partial n} = 0. \tag{5.3.6}$$

On the outside boundaries, mixed boundary conditions [33] are applied,

$$\frac{\partial \phi_s}{\partial n} + \frac{cos(r,n)}{r}\phi_s = 0, \tag{5.3.7}$$

where r denotes the radial distance from the source location to the boundary.

Now, after collecting all the discrete FD equations for all nodes and applying BCs, for a current source, f, a matrix equation is formed as,

$$\mathbf{A_m}\phi_a = \mathbf{s}, \tag{5.3.8}$$

where $\mathbf{A}_m$ is a source independent $N_v \times N_v$ sparse matrix with seven non-zero elements per row (Fig. 5.1); $\mathbf{s}$ is the N_v dimensional source vector and ϕ_a is the N_v dimensional vector representing secondary potential at the N_v internal nodes. This sparse linear system is solved iteratively using BiCGSTAB. The incomplete **LU** decomposition of the matrix is used for preconditioning. However, for a large number of right hand side vector, the system matrix can be solved using direct solver. The **LU** decomposition of the matrix is done along with a row permutation matrix and a column reordering matrix and a diagonal scaling matrix. Once the decomposition is complete the solutions are obtained simply by backward and forward substitution and multiplication. Such an strategy is quite useful for large number of electrodes, which is commonly used for multi-electrode system for data acquisition. However, if the system matrix is large then **LU** decomposition is time consuming and storage of decomposed matrices requires large memory space. Once the eq 5.3.8 is solved for the secondary potentials these are added to primary potentials and the responses are computed for arbitrary electrode array/geometry.

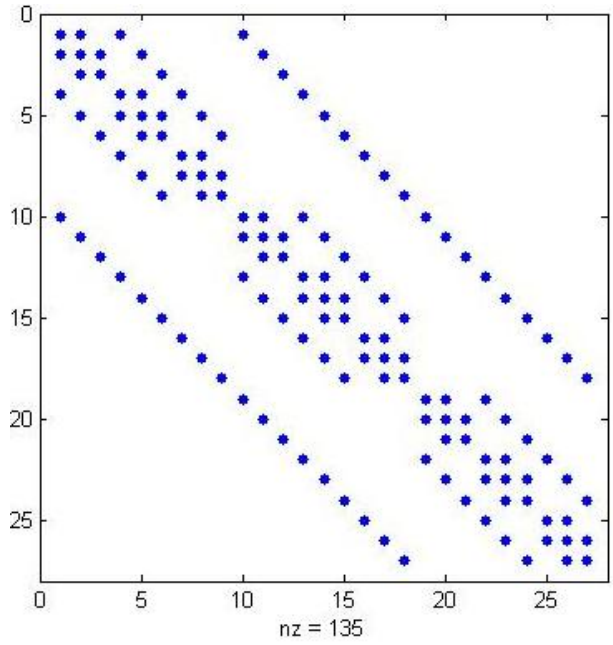

Figure 5.1: Coefficient matrix $\mathbf{A_m}$ for $3 \times 3 \times 4$ grid.

5.3.3 Response Computation

Once the eq 5.3.8 is solved and secondary potentials are computed, the total nodal potentials are computed as sum of primary and secondary potentials. Since the computed potentials are for a single source, they are superimposed to form the potentials due to a dipole. Such

approach is very efficient as individual electrodes are used more than once. For e.g., consider a 48 electrode system. For Dipole-Dipole configuration there are 168 dipoles corresponding to 7 dipole lengths but only 48 current sources.

The computation of apparent resistivity from potentials is straight forward, which requires a simple transformation matrix. The transformation matrix interpolates the nodal potential to the potential electrode location and then computes the potential difference between the measuring pair of electrodes. To compute the responses from the nodal potential ϕ, a transformation matrix, $\mathbf{T}$, is used to get the predicted response $\mathbf{d}^{pred}$ as:

$$\mathbf{d}^{pred} = \mathbf{T}\phi. \tag{5.3.9}$$

Generally, the observed field data is potential difference. From these apparent resistivity can be computed as:

$$\rho_a = K \times \frac{\delta\phi}{I}, \tag{5.3.10}$$

where, K is the geometric factor that depends only on the position of current and potential electrodes.

5.4 Numerical Implementation

In inversion algorithm the main operators are forward operator, data and model covariance matrices and Jacobian. As mentioned earlier in subsection 3.5.2, these are standalone operators in *AP3DMT* code with no direct interaction with the inversion function, hence it is very easy to incorporate necessary changes in the corresponding sub-programs or functions without disturbing basic structure of code. To include 3D DCR inversion into *AP3DMT* code, necessary modification were made in forward operator and Jacobian. Therefore, in the following we will only discuss the modifications in forward modeling, Jacobian and its various components.

5.4.1 Forward Modeling

The forward modeling function, *ForwardResp_DC*, a standalone function, includes (*i*) parameter mapping to edges, *AverageSigma*, (*ii*) setting up boundary conditions, *BoundCond_DC*, (*iii*) computing potentials at grid nodes, *DiscreteSolver_DC* and (*iv*)

the predicted data at observation points, *OutInterp_DC*. These functions were added in *AP3DMT*. Similar to MT problem, parallelization is implemented over number of current electrodes.

Since, the coefficient matrix $\mathbf{A_m}$ is independent of the source (current electrode) the construction of coefficient matrix, with boundary conditions enforced, is formed only once and stored in a sparse format in *ConstructMatrix*. The **ILU** or **LU** decomposition (depending on the type of matrix solver used) is also performed only once in the same function. To avoid overhead, while transferring matrices from master to workers during parallel implementation, these matrices are transferred to all active workers using *WorkerObjWrapper* which manage persistent state on *matlabpool* workers by avoiding re-transfer of arrays which are used very frequently. Such an approach is beneficial when the problem is solved for large number of current sources.

For a specific source, *BoundCond_DC* is called right hand side vector, **s**, is formed and stored as a column vector. *BoundCond_DC* includes analytic computation of primary potential for the primary model. Once primary potential are computed, vector, **s** is forming by computing $-\nabla \cdot (\sigma_a(x,\ y,\ z)\nabla\phi_p(x,\ y,\ z))$ (eq 5.3.4). The matrix solver, in *DiscreteSolver_DC*, is called and the secondary potential solutions are obtained. These potentials are superimposed to compute the potentials according to the array configuration. The *OutInterp_DC* uses the previously computed interpolation functional (discussed in next section) and potential difference to evaluate the response (data) at all desired measuring pair of electrodes. For computation of sensitivity, potential solutions are also returned. The whole process from *BoundCond_DC* to *OutInterp_DC* is looped over number of sources and all the data and solutions are assembled together in *ForwardResp_DC*.

5.4.2 Jacobian Computations

In general, there are three ways to compute the sensitivity for the DC resistivity [167], *(i)* the perturbation method, *(ii)* the sensitivity forward calculation and *(iii)* the potential approximation. We have used sensitivity forward calculation. Similar to MT, computation of predicted data $\mathbf{d}_f$ (forward mapping $F_f(\mathbf{m})$), for a source f, involves two steps; (*i*) solving the governing PDE (eq 5.2.4) for potentials ϕ and, (*ii*) computing the predicted data like apparent resistivity at observation sites using these potentials. The predicted data set at the

observation points can be written as,

$$\mathbf{d}_f^j = F_f^j(\mathbf{m}) = \psi_j(v(\mathbf{m}), \mathbf{m}) \qquad j = 1, \ldots, nobs. \tag{5.4.1}$$

where, ψ_j is generally a non-linear function of ϕ (and possibly $\mathbf{m}$). The expression for the Jacobian matrix of the forward mapping $F_f(\mathbf{m})$, in the block form is written as,

$$\mathbf{J}_f = \mathbf{L}_f \mathbf{A}_\mathbf{m}^{-1} \mathbf{P}_f, \tag{5.4.2}$$

where, symbols has the same notation as before (Section 2.5.3).

In the DCR problem, the dataset is real hence, all the computations of Jacobian are efficiently performed using real arithmetic unlike MT where complex arithmetic was used. As discussed earlier, the Jacobian is not formed explicitly rather product of Jacobian (ot its transpose) with an arbitrary vector is performed using its three components as shown in eq 5.4.2. The details of these components are discussed in the following.

5.4.2.1 Sensitivity functional

The function *LpJobs_DC* performs the linearization of the data functional with respect to variations in the potential solution ϕ. The $\mathbf{L}$ matrix of dimension $N_{obs} \times N_v$ is explicitly formed and stored in sparse storage. The rows of $\mathbf{L}$ matrix interpolates the potential at electrode locations and compute the potential difference. If the measured component is resistance, then $\mathbf{L}$ simply computes the interpolated values at the electrode locations. For *datatype* apparent resistivity the product $\mathbf{L}\mathbf{v}_\phi$ is multiplied element by element by another vector whose elements are the geometric factors. *LTpJobs_DC* multiplies this vector by the appropriate data components to form $\mathbf{L}^T \mathbf{v}_{\mathbf{d}f}$. Here, instead of implementing $\mathbf{L}^T \mathbf{v}_{\mathbf{d}f}$, we have implemented $(\mathbf{v}_{\mathbf{d}f}^T \mathbf{L}_f)^T$. This does away with the need to implement $\mathbf{L}^T$.

5.4.2.2 Forward Solver

The function *DiscreteSolver_DC* includes two types of solvers, *(i)* the standard BiCGSTAB iterative solver for solving eq 5.3.4 with the incomplete $\mathbf{LU}$ decomposition for preconditioning, and *(ii)* direct solver using $\mathbf{LU}$ decomposition along with a row permutation matrix, a column reordering matrix and a diagonal scaling matrix. For each source, the output vector, $\mathbf{v_e}$ consists of a set of solution corresponding a right hand side vector for each source.

5.4.2.3 Solver Sensitivity

The third component of Jacobian is $\mathbf{P}_f$, which defines the product of the sensitivity of the coefficient matrix, $\mathbf{A_m}$, to the model parameters (log conductivity) with the potential field. The derivative of the field with respect to the parameters is similar to MT case. As discussed earlier, the operator $\mathbf{P}_f$ consists of two components. The first part provides the derivative of the coefficient matrix with respect to the model parameters. For a FD approximation, all the seven elements of a row of the matrix, $\mathbf{A_m}$, depend on conductivity elements in the form of $\bar{\sigma}$ (volume weighted average of four nearby cells). Hence, each parameter contributes to eight internal nodes. This results in a maximum of thirty-two non-zero elements per parameter (for each node three off diagonal elements and a diagonal element). These non-zero elements are $-V_i exp(m_i)$ where, V_i is the integration volume for the i^{th} cell having parameter m_i defined as log conductivity. Once these derivatives are obtained, these are multiplied with potential solution vector ϕ, corresponding to a source f. Afterwards, this vector is multiplied element by element with the arbitrary vector and all such vectors are summed together. For the transpose case, these steps are reversed in *PTpJob_DC*.

5.5 Validation of Forward Modeling

The developed forward modeling code is tested over a variety of models. For demonstration we have selected two models from literature. First model is a vertical dike model [166] and the second is a two-layer model with an embedded cube [166].

Model 1

In this model the vertical dike, 5 m wide, extends to infinity in $\pm x$ and $+z$ direction. The dike is at 20 m offset from the origin of the model. The resistivity of the dike and the half-space are 10 Ω m and 100 Ω m, respectively (Fig. 5.2). The Schlumberger soundings, with two current sources at $(0,-1,\ 0)$ m and $(0,+1,\ 0)$ m are carried out over the structure along the y axis. The grid information was taken from the Spitzer [166]. Accordingly, the model was discretized into $72 \times 72 \times 39$ irregular cells and the modeling domain boundaries are located at $X : \pm 5500$, $Y : \pm 5500$ and $Z : 5500$ m. The system of equations was solved using iterative solver with a tolerance of 10^{-7}. Fig. 5.3 shows the variation of apparent resistivity and absolute relative error in apparent resistivity versus half current electrode spacing for

Schlumberger configuration along the $y-$ axis i.e. perpendicular to the dike. The absolute relative error in the apparent resistivity is less than 1%.

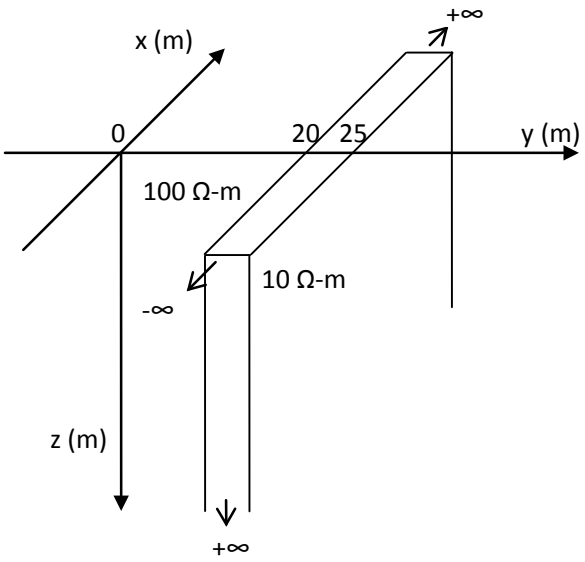

Figure 5.2: A vertical dike model [166].

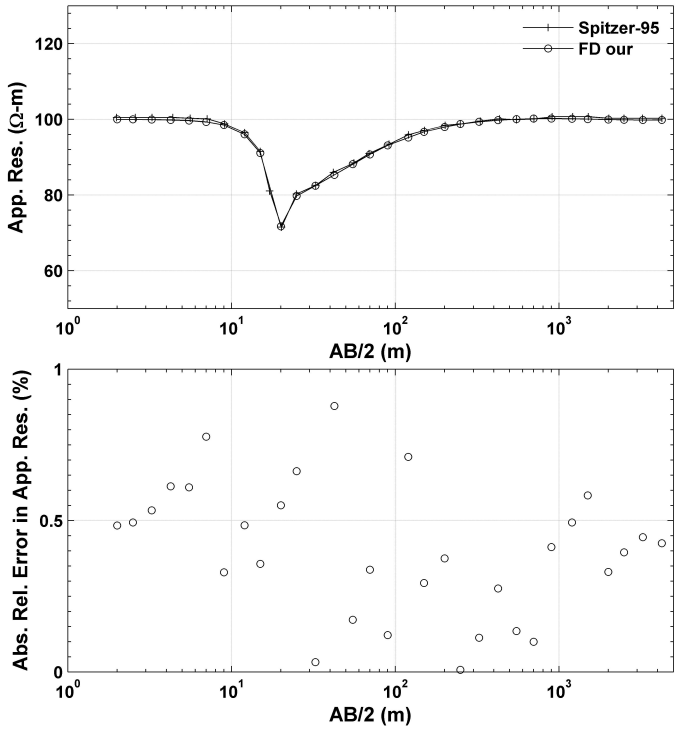

Figure 5.3: (top) Comparison of the apparent resistivity obtained from analytic solution and FD method *AP3DMT* versus half current electrode along the $+y$ axis perpendicular to the dike, and (bottom) absolute relative error in apparent resistivity versus half current electrode.

Model 2

The second synthetic model consists of a cube embedded in a two layer earth [166]. The first layer with resistivity 100 Ω-m extends upto 3 m while the resistivity of the underlying half-space is 10 Ω-m. A cube of resistivity 10 Ω-m and side length 2 m is embedded in the first layer with its centre at $(x,\ y,\ z) = (0,\ 2,\ 1.5)$ m (Fig. 5.4). The Schlumberger soundings with two current sources at $(0, -0.1,\ 0)$ m and $(0, +0.1,\ 0)$ m are carried out over the structure along the $y-$ axis. Fig. 5.5 shows the variation of apparent resistivity and relative error in apparent resistivity versus half current electrode spacing for Schlumberger configuration along the y axis. For majority of the points the absolute relative error in the apparent resistivity is less than 3% and it reaches 4% at large distance.

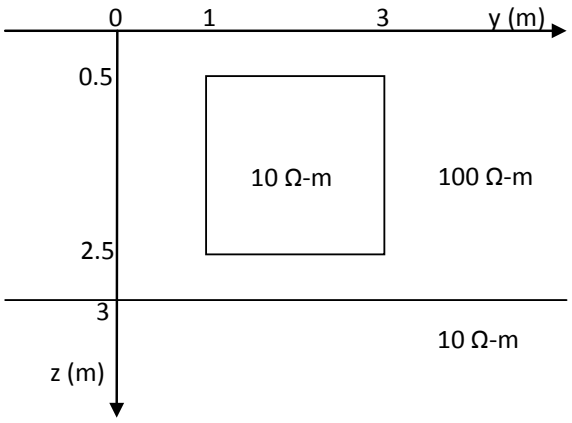

Figure 5.4: A buried cube in a two layer earth model [166].

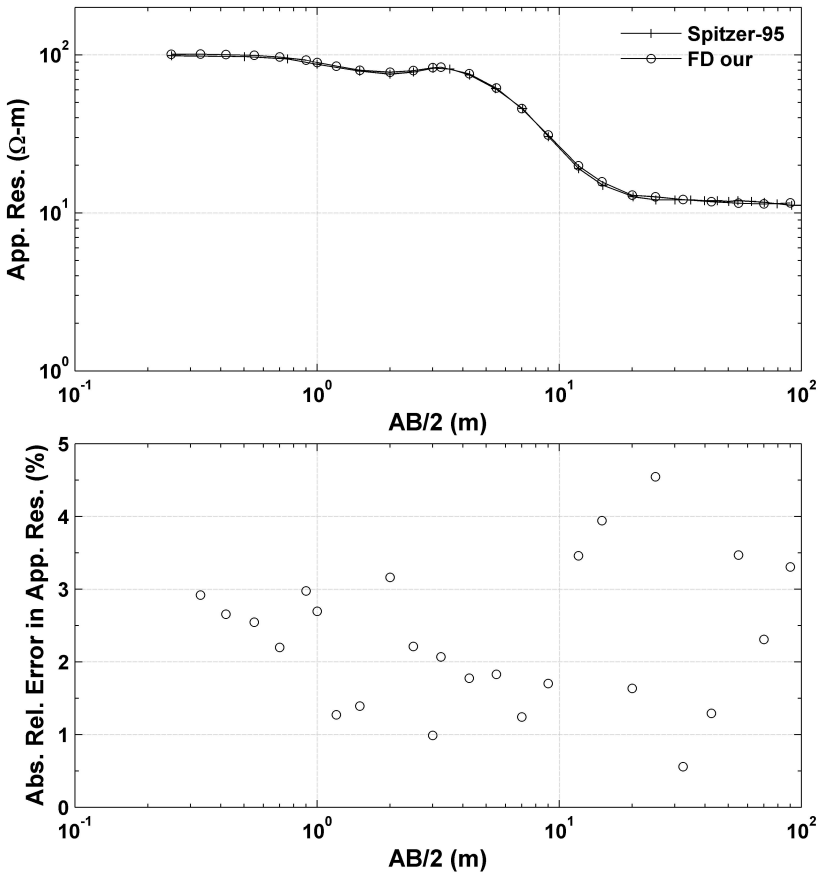

Figure 5.5: (top) Comparison of the apparent resistivity obtained from FD Spitzer [166] and *AP3DMT* versus half current electrode spacing for Schlumberger configuration along the $+y$ axis, and (bottom) absolute relative error in apparent resistivity versus half current electrode.

5.6 Validation of Inverse Modeling

The developed code has been tested and validated over a variety of 3D standard models available in literature. For demonstration, we have selected two synthetic test model. Further, versatility and robustness of the code will again be demonstrated in next chapter. The first test model is a two block model with a contrast resistivity model (modified after Boonchaisuk et al. [13]). The second model is a representation of a mine dump model (similar to Günther et al. [52]). These synthetic data sets were inverted using *NLCG* algorithm on HP Z620, Intel Xeon E2643, 32 GB with 8 cores for parallelization.

Model 1

The model consists of two blocks, one conductive (1 Ω-m) and one resistive (100 Ω-m). Each block has a dimension $60 \times 20 \times 12.5$ m and are next to each other. These blocks are embedded in half-space of resistivity 10 Ω-m. The top of the blocks is 2.5 m below the surface. Three data sets, of the apparent resistivity, using wenner, Schlumberger and Dipole–Dipole configurations are computed along 5 profile, with inter profile spacing of 20 m. Each profile has 31 electrodes with an inter electrode spacing of 5 m and a separation factor of n = $1-10$ for Wenner, n = $1-14$ for Schlumberger and n = $1-15$ for Dipole–Dipole (Fig. 5.6). This results in total of 725, 1050 and 1575 data points for Wenner, Schlumberger and Dipole–Dipole respectively. For computing accurate synthetic data the model was discretized into $78\times$ 132 $\times$ 22 cells with horizontal cell dimensions of 1.25 m. Five percent Gaussian noise was added to the data. Errors floor is set as 5% of the apparent resistivity to accommodate the discretization errors from using different meshes for the inversion and the forward modeling. For inversion, a homogeneous model of 9.5, 9.7 and 9.8 Ω-m for Wenner, Schlumberger and Dipole–Dipole data set respectively (based on the average of observed apparent resistivities) was used as an initial guess model. The model was discretized into $45\times$ 71 $\times$ 22 cells with horizontal cell dimensions of 2.5 m. For the three datasets *GN-MS* was used for inversion. The **LU** of the coefficient matrix (along with permutation matrix, column reordering matrix and a diagonal scaling matrix) was performed only once and the matrix equation is solved simply by backward and forward substitution and multiplication. In 9 iterations the nRMS error reduced from 10.7 to 0.90 using Wenner data. For Schlumberger array data, nRMS error reduced from 33.5 to 0.95 in 12 iterations while for Dipole–Dipole

array it reduced from 52.1 to 0.95 in 13 iteration.

The inverted models, for three different datasets, thus obtained are shown in Fig. 5.7 (second, third and last panel). In all the three inverted models, shape of the blocks were successfully recovered. Also, the top surfaces of the two blocks are clearly seen at a depth around 2.5 m. However, the bottom of both the blocks are poorly resolved for the Wenner data as compared to Schlumberger and Dipole-Dipole data. Fig. 5.8 – Fig. 5.10 shows comparison of observed and predicted data for a profile at $x = 0$ in pseudosection. Fig. 5.11 shows the convergence plot of the three datasets.

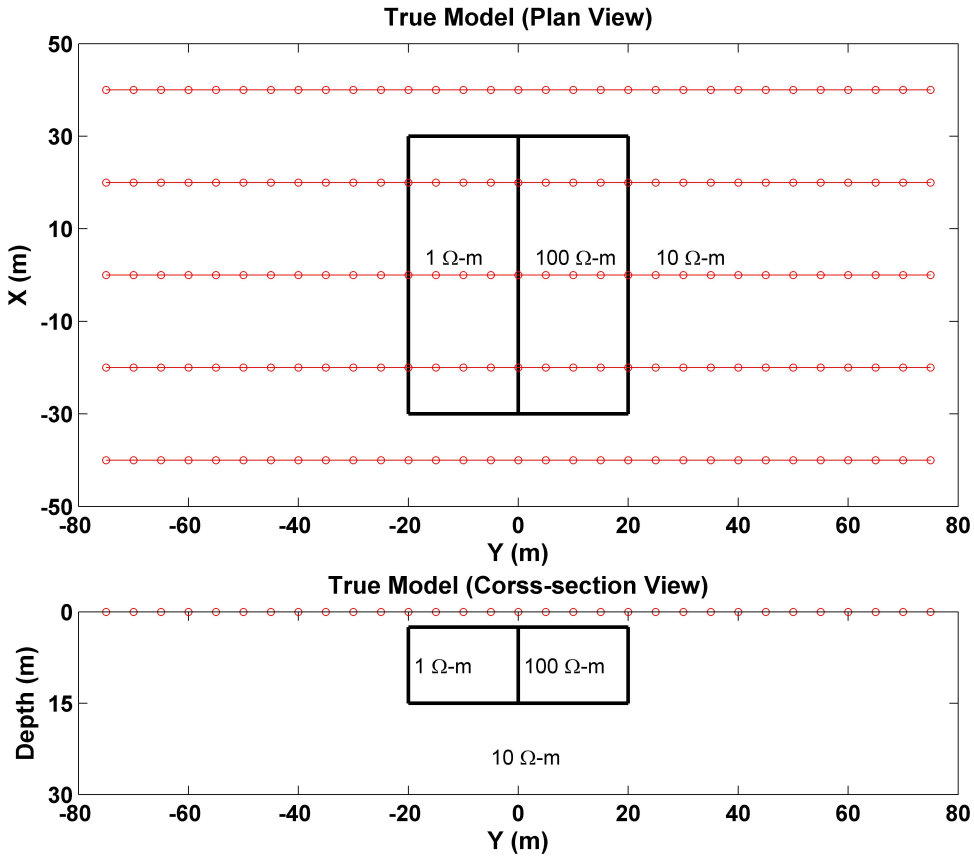

Figure 5.6: (Top) A plan view of true model with electrodes positions, and (below) cross section at $x = 0$ m with circles representing the electrodes (modified after Boonchaisuk et al. [13]).

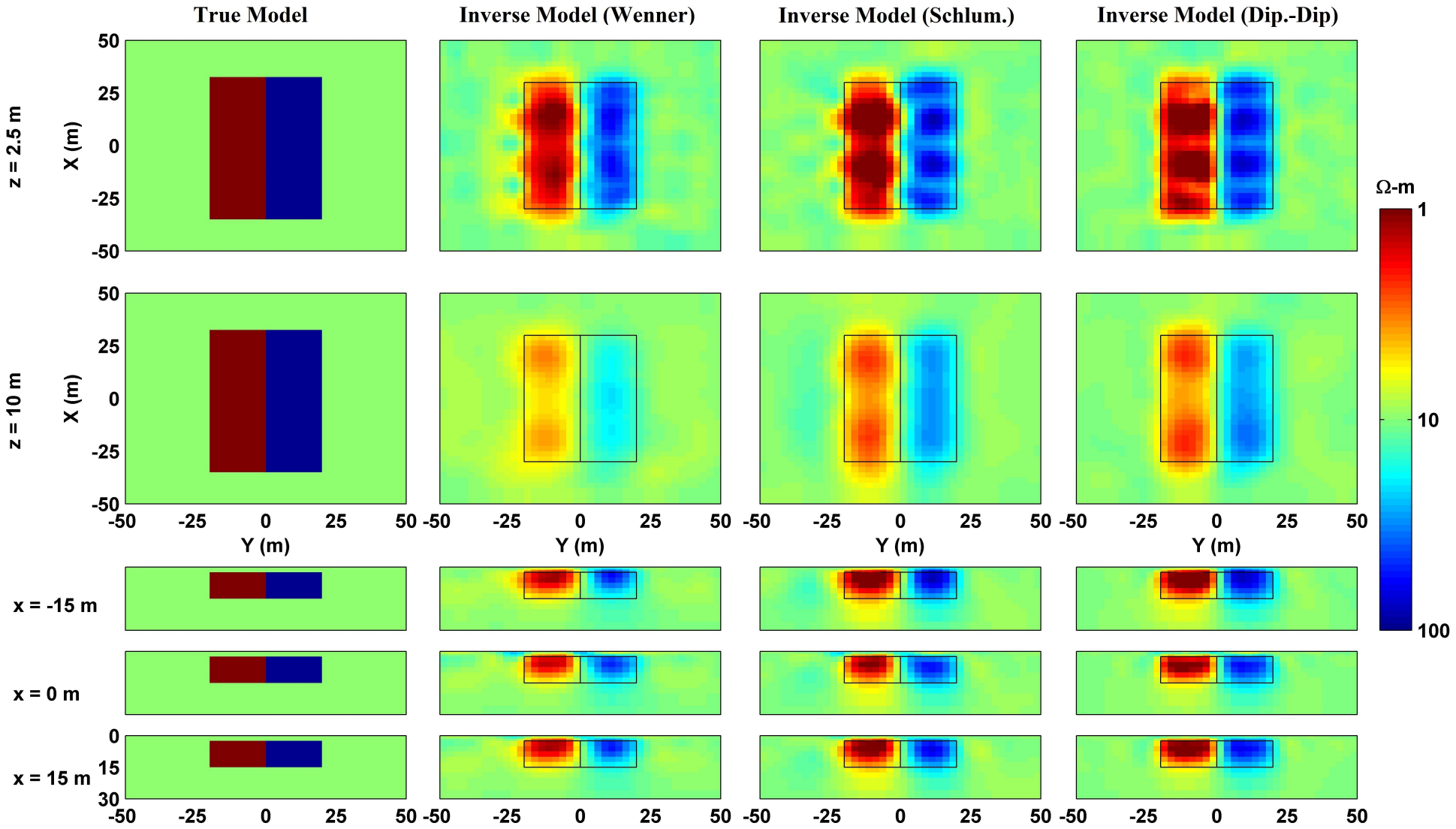

Figure 5.7: A plan view of true model (first panel) and the inversion model using Wenner array data (second panel), Schlumberger array data (third panel), and Dipole-Dipole array data (last panel). The black lines, second panel onwards, indicates the true location of the two blocks.

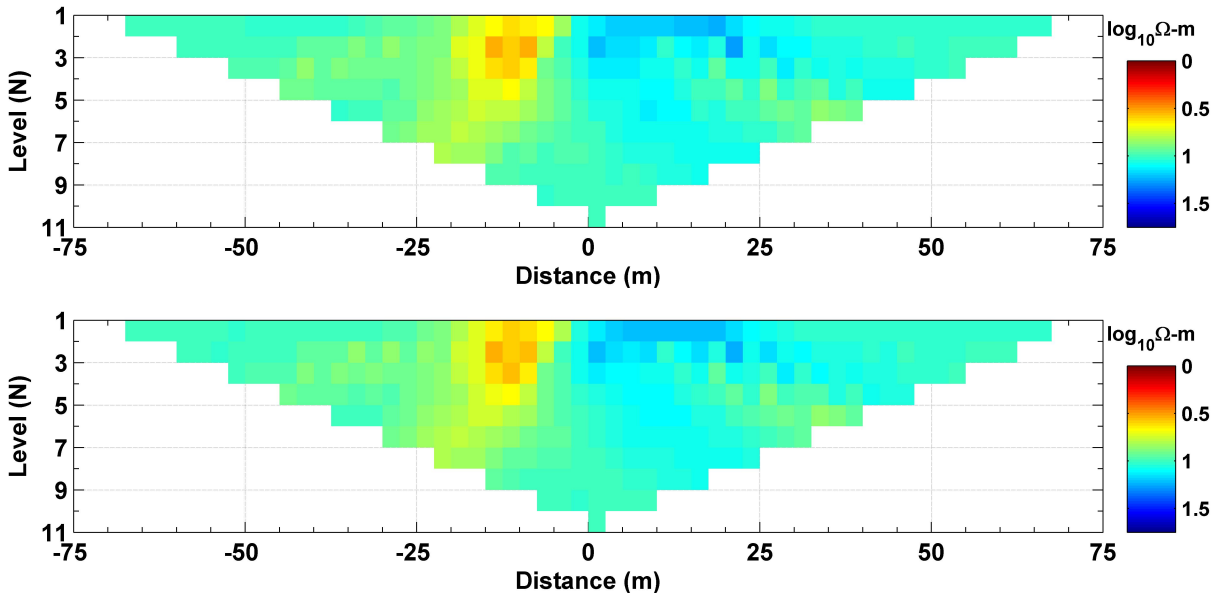

Figure 5.8: (top) Synthetic Wenner array data shown in pseudosection, and (below) Predicted Wenner array data of the inverse model for $x = 0$ profile.

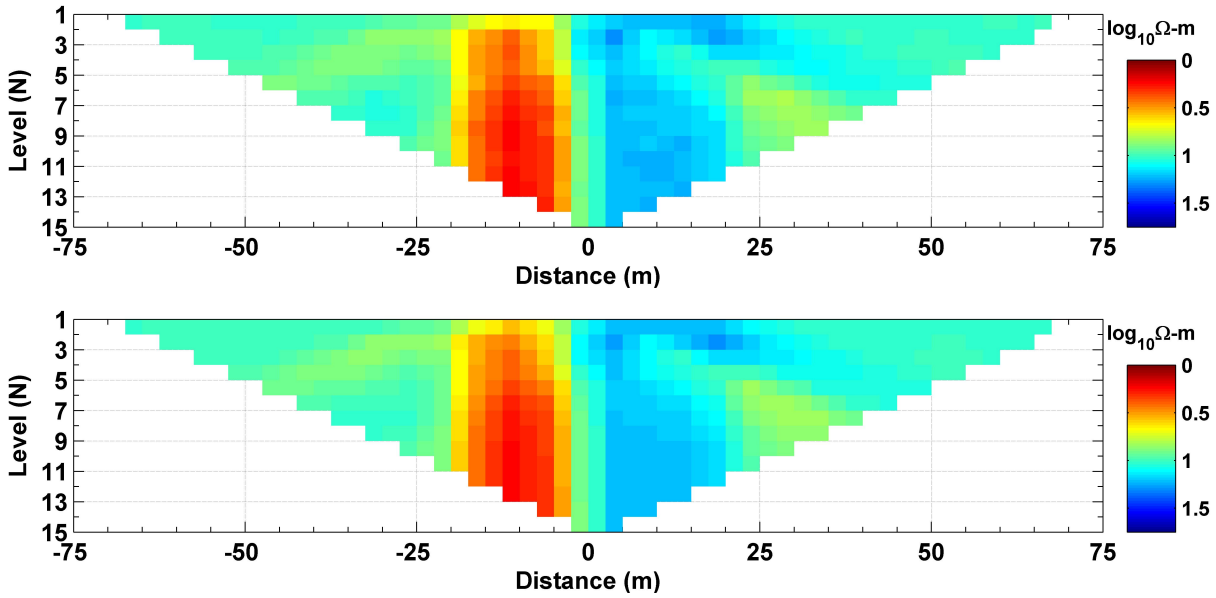

Figure 5.9: (top) Synthetic Schlumberger array data shown in pseudosection, and (below) Predicted Schlumberger array data of the inverse model for $x = 0$ profile.

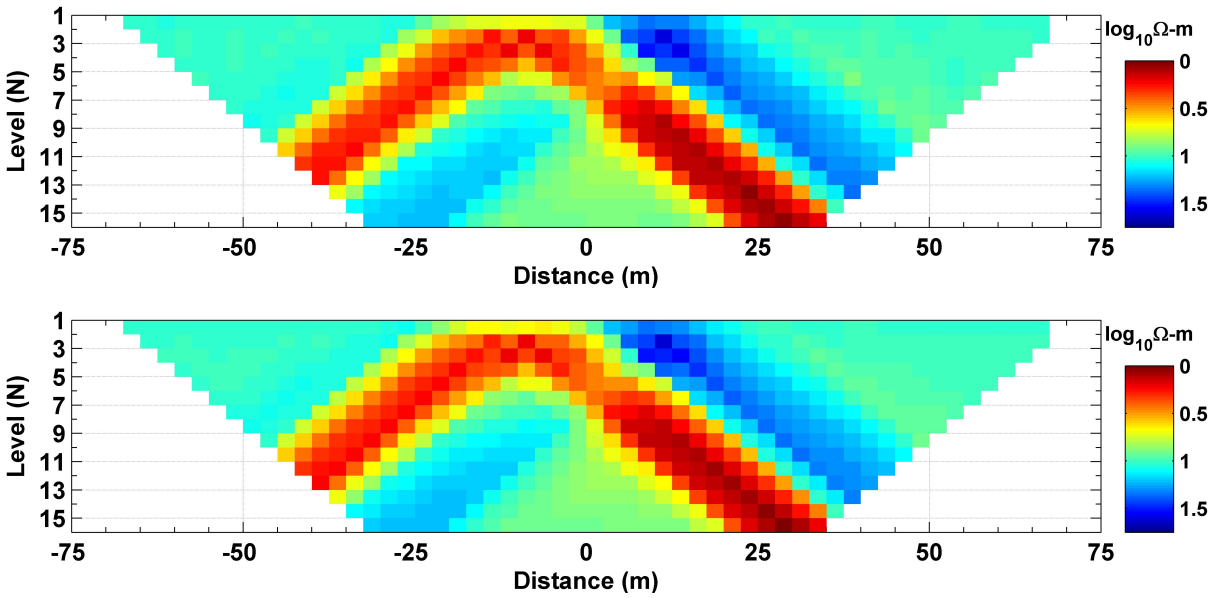

Figure 5.10: (top) Synthetic Dipole–Dipole array data shown in pseudosection, and (below) Predicted Dipole–Dipole array data of the inverse model for $x = 0$ profile.

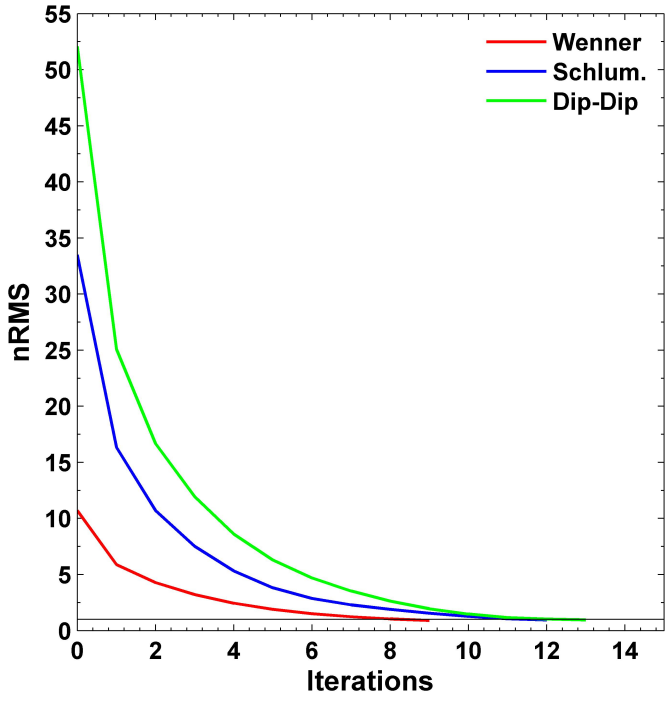

Figure 5.11: Convergence plot for all the three datasets.

Model 2

This model represents a complex geology, similar to a mine dump [52]. The model consists of a half ellipsoid with center at the centre of the model and semi-axis lengths 125, 40 and 25 m along $x-$, $y-$, and $z-$ direction respectively. The resistivity of the body was set at 15 Ω-m against the half space resistivity of 100 Ω-m. Four hardpans structures of arbitrary shape and varying depth ($2-3$ m) and resistivities ($400-800$ Ω-m) were generated (Fig. 5.12).

The DC resistivity measurements were made for Dipole–Dipole array with 13 profiles. Each profiles have electrode separations of 5 m. The inter profile distance is 20 m, which is four times the minimum electrode separation, thus, sufficient for 3-D reconstruction of the model. Two dipoles of length 5 and 10 m were used with a separation factor of n = $1-8$. With this electrode configuration 1800 data points were generated.

For the accurate representation of the anomalies, synthetic model was discretized into $62 \times 62 \times 42$ cells with horizontal cell dimensions of 5 and 2.5 m in $x-$ and $y-$ direction respectively. Five percent Gaussian noises were added to the synthetic data. Errors floor is set as 5% of the apparent resistivity. For inversion, a homogeneous model of 65 Ω-m (average of observed apparent resistivities) with $62 \times 62 \times 18$ cells was used as a guess model (5 m $\times$ 2.5 m $\times$ 1 m, vertical extent increasing with depth by a factor of 1.2). Fig. 5.13 shows the inversion result obtained after 13 iterations using *GN-MS* with nRMS error reduced from 52.01 to 0.96. The shape and the resistivity of the conductive dump material is very well delineated. The dimensions of the four hardpan were resolved with correct order of resistivities.

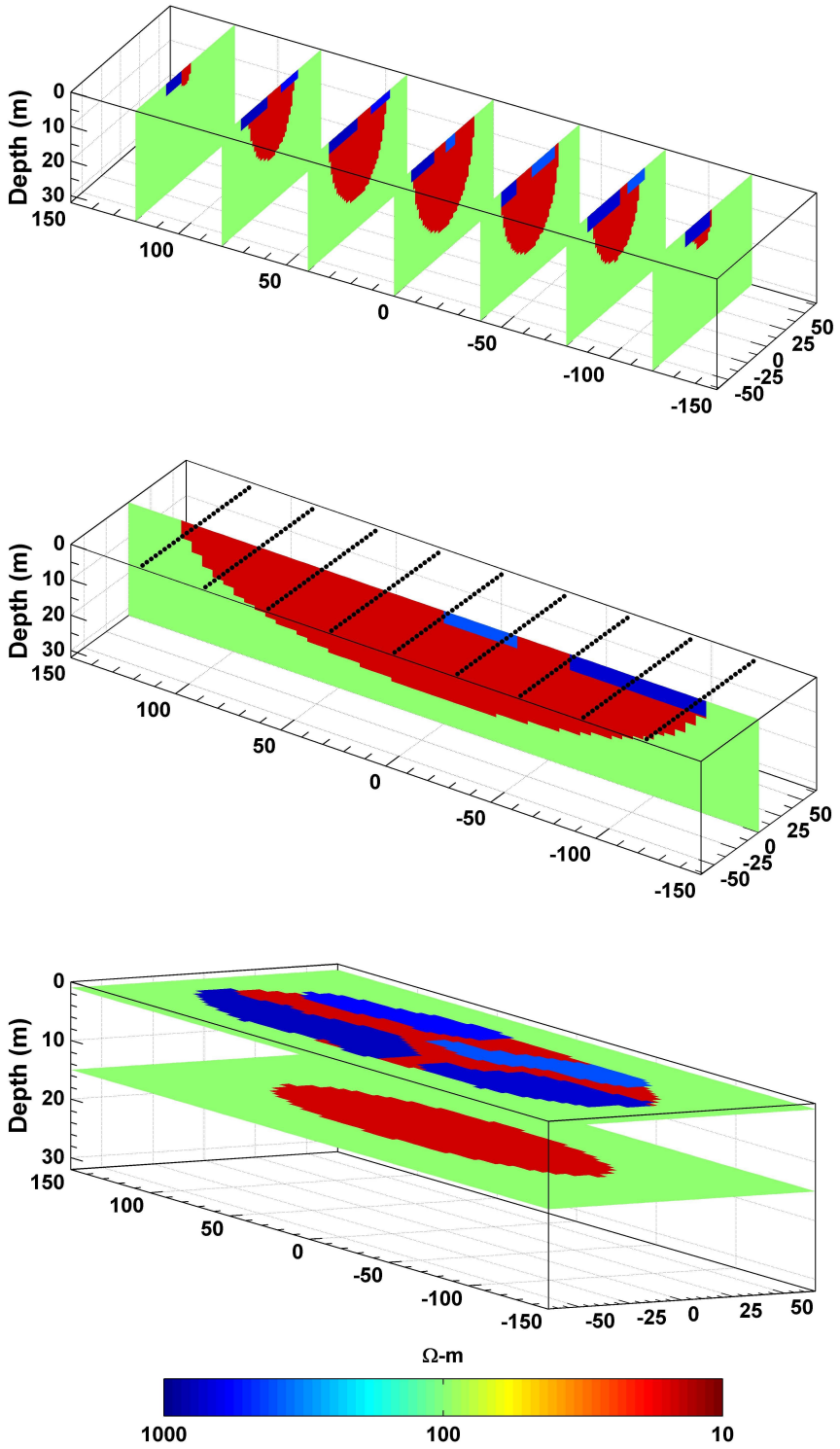

Figure 5.12: Dump model showing anomaly along with four hard-pan structures of varying depth and resistivities: (a) Slices parallel to profiles, (b) Slices perpendicular to profile; black circles represents the electrode position, and (c) Slices at z = 1 and z = 15 m depth.

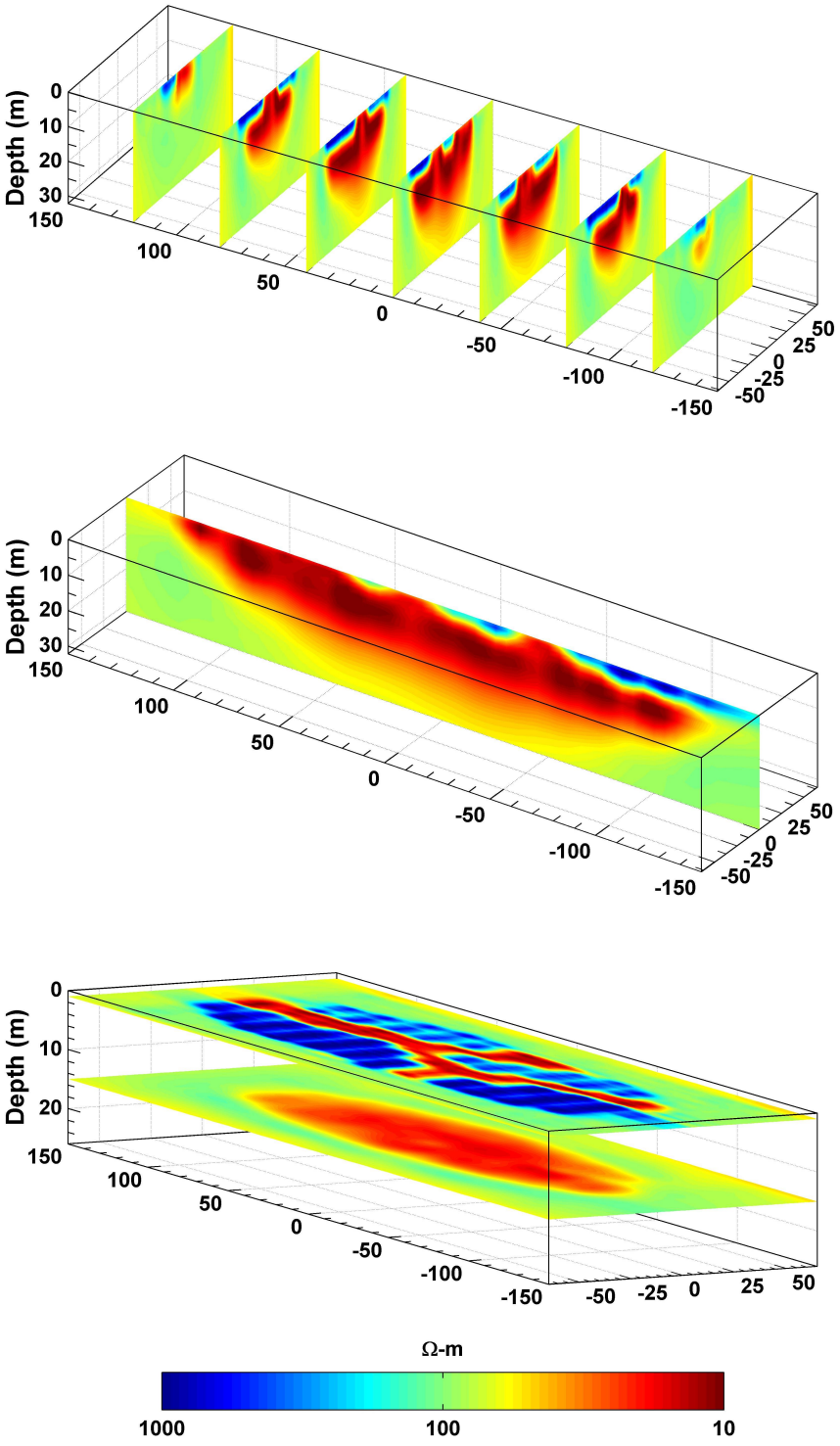

Figure 5.13: Inverse model after 40 iterations: (a) Slices parallel to profiles, (b) Slices perpendicular to profile, and (c) Slices at z = 1 and z = 15 m depth. Four harpan structures of 400, 500, 600 and 800 Ω-m are imaged.

5.7 Summary

The inversion of DCR data is successfully incorporated in *AP3DMT* code. In this chapter 3D DCR forward and inverse modeling algorithm are developed and tested over complex 3D geological models. The concept of partitioning of potentials is used to remove singularity at source. The primary potential are computed analytically for simple models like, uniform half-space. A efficient preconditioned iterative solver is used to solve the system matrix for secondary potentials. Further, the accuracy of the simulated responses are compared with published results.

Similar to MT case, instead of forming Jacobian explicitly, the multiplication of Jacobian (and its transpose) with an arbitrary vector is performed more efficiently by using the parts of Jacobian. Parallelization over number of sources is implemented for both forward modeling and Jacobian$-$vector product computations which helps in accelerating the computations for large data set. Inversion test over synthetic model demonstrates the versatility of the code.

Chapter 6

MT – DCR JOINT INVERSION

6.1 Preamble

The all types of the MT and DCR surveys yield cross-sections of resistivity from which the physical structure may be determined. They are used worldwide for many different purposes. In contrast to DCR surveys which use man-made sources, an audio -magnetotelluric (AMT) and MT surveys use natural EM signals for sensing from shallow to mid-crust and to deeper Earth (discussed in chapter 1). For very shallow surveys, remote radio transmitters operating in the VLF and LF replaced the natural EM signals of the Earth for the radio-magnetotelluric (RMT) method [e.g. 86, 175, 178].

Both MT and DCR data is sensitive to the same physical parameters resistivity, however, they differ in physical concept by which they are governed. MT is mainly governed by induction phenomena whereas in DCR current is injected in the zone of interest galvonically. Thus the sensitivity of these two methods (MT and DCR) for resistive and conductive structures may vary. It would be interesting to study performance of joint inversion of MT and DCR data set. There are two different approach for performing joint inversion. First, the two different data sets are inverted independently to produce 3D image of survey area and results are manually combined to produce a compatible interpretation [34, 146]. In second approach, the two different data sets are inverted together to constrain the interpretation [8, 20, 66, 143, 176, 138]. The advantages of using joint inversion is that the strength of one technique can help mitigate a drawback of the other technique. For example, MT data is sensitive to conductive structures while DCR data is sensitive to both conductive and resistive structures. Hence, joint inversion often results in better inverse models as compared

to individual inversion. Although, the joint inversion of the MT and DCR data yields better results, the joint (MT and DCR) survey is still not conducted regularly. One of the main reasons is the non-availability of joint inversion codes in public domain. 2D joint inversion were performed by Sasaki [138] followed by various authors [e.g. 20, 143, 146, 176]. Recently, Amatyakul et al. [2] presented *WSJointInv2D-MT-DCR* a joint inversion program for 2D MT and DCR data based on Occam's data space technique. However, in the presence of 3D bodies, 2D inversion algorithm may not be an accurate choice [138]. Keeping this is mind, we extended our code to implement 3D joint inversion of MT and DCR data.

In this chapter, we will discuss the additional features needed for implementation of joint inversion of MT and DCR data in *AP3DMT* code. The assemblage of data sets of MT and DCR and Jacobian are discussed. In addition, three crucial aspects, model discretization, data scaling and error floor, as used in joint inversion are also discussed. Finally individual and joint inversion of one synthetic test example is presented.

6.2 Joint Inversion of MT and DCR Data

In the following discussion, we will use the same notation as described earlier chapters. The penalty function for the joint inversion of MT and DCR data after transformation (see 2.5.1) is written as,

$$\phi(\mathbf{m},\mathbf{d}^{obs}) = (\mathbf{d}^{obs} - F(\mathbf{m}))^T(\mathbf{d}^{obs} - F(\mathbf{m})) + \lambda \mathbf{m}^T\mathbf{m} \tag{6.2.1}$$

For implementation of joint inversion, different data sets of MT and DCR must be combined as,

$$\mathbf{d} = \begin{bmatrix} \mathbf{d}_{MT} \\ \mathbf{d}_{DCR} \end{bmatrix} = \begin{bmatrix} F_{MT}(\mathbf{m}) \\ F_{DCR}(\mathbf{m}) \end{bmatrix}; \qquad \mathbf{d}^{obs} = \begin{bmatrix} \mathbf{d}^{obs}_{MT} \\ \mathbf{d}^{obs}_{DCR} \end{bmatrix}. \tag{6.2.2}$$

The data vector $\mathbf{d}^{obs}_{DCR} = [\rho^a]$ is the N_{DCR} dimension DCR data set obtained for a number of sources. The MT data vector $\mathbf{d}^{obs}_{MT}$ of dimension N_{MT} is obtained for a range of frequencies and can be represented as,

$$\mathbf{d}^{obs}_{MT} = \begin{bmatrix} \mathbf{Z}_{xx} \\ \mathbf{Z}_{xy} \\ \mathbf{Z}_{yx} \\ \mathbf{Z}_{yy} \end{bmatrix} \quad or \quad \begin{bmatrix} \rho^a_{xy} \\ \phi_{xy} \\ \rho^a_{yx} \\ \phi_{yx} \end{bmatrix} \tag{6.2.3}$$

As in case of MT different data types may be combined like Impedance and VTFs or Phase Tensor and VTFs, etc. Similarly, the total number of the data N_d for the joint inversion is the sum of the number of both data types, $N_d = N_{MT} + N_{DCR}$. The data covariance matrix, diagonal matrix in this case is also written such that its size is same as the number of both data types as,

$$\mathbf{C_d} = \begin{bmatrix} \mathbf{C_d}^{MT} & 0 \\ 0 & \mathbf{C_d}^{DCR} \end{bmatrix}. \tag{6.2.4}$$

A brief discussion on the other aspects like, model discretization, data scaling, error floor etc., as implemented in the joint inversion are given in the following.

6.2.1 Model Discretization

Since, both data types corresponds to the same sub-surface resistivity structure, the same model **m** can be used for joint inversion. Both techniques sense resistivity structures at different depth (DCR shallow depth while MT deeper depth) hence grid discretization should be such that it can accommodate both techniques. This is also necessary for high accuracy of solutions for the forward problem. For DCR method the grids are designed on the basis of the DCR configuration as discussed earlier. Since, MT may probe to greater depths, the grid discretization is vertically extended to greater depths. The grids are also extended horizontally on all four sides using larger grid spacing. This is to satisfy the boundary conditions and to ensure high accuracy for the MT forward problem. However, such grid discretization results in large number of model parameters. This increase in **m** will lead to a longer computational time for solving the forward problems for both types of data.

6.2.2 Data Scaling

In joint inversion, the two different data sets, corresponding to MT and DCR are combined together. Assuming other parameters being constant, like similar noise levels, the data set with more number of independent data points can have significant influence on the imaging. For example, consider a model with resistive subsurface features. Since, DCR method is more sensitive to resistive features as compared to MT, the DCR data set might bias the joint inverse model towards the DCR inverse model (obtained by DCR inversion only). To resolve this crucial issue two different approaches can be used. A trade-off parameter between the two data sets similar to the model regularization parameter λ may be considered. However,

instead of using additional parameter one can modify error bars and directly manipulate the influence of each data set or its subset thus reducing biasness of any data type. Following Commer and Newman [24] two different schemes are implemented.

In the first scheme, balancing is done solely on the basis of number of data points in each data set. Let us consider MT and DCR data set and call them as data set 1 and 2 respectively. Both of these set contain $N1$ and $N2$ data points, and assume $N1 > N2$. Then the set 2 is up-weighted by applying a factor g_w,

$$\tilde{w}_n = w_n g_w; \qquad g_w = \sqrt{\frac{N1}{N2}}, \tag{6.2.5}$$

where w_n are the inverse of the variances of the measurements. Since, data misfit ϕ_d depends quadratically on w_n hence, the square root. This scheme is likely to reduce the biasness of data set with more number of data points if both data sets are characterized by similar intrinsic sensitivities. If that is not the case then further up-weighting of a data set is required.

In second scheme, the re-weighting is done using the norms of the gradients of the data misfit of both data sets computed individually. Here, the up-weighting factor g_w for the data set 2 is computed as,

$$g_w = \sqrt{\frac{\|\nabla\phi_1\|}{\|\nabla\phi_2\|}}. \tag{6.2.6}$$

The gradient are computed using the initial guess model and provides an estimate of the intrinsic data sensitivities. This scheme incorporates both the quantity (in terms of number of data points) and resolution capacity in a joint data set.

6.2.3 Error Floor

If the data errors contained in the matrices $\mathbf{C_d}^{MT}$ and $\mathbf{C_d}^{DCR}$ are close to the true ones, there is a certain danger of obtaining either a badly resolved model or an unstable model when iterating to an nRMS of one for the re-weighted system. Hence, the desired nRMS should be adopted according to the chosen weighting factors [67].

Let total number of data be $N_d = N_{MT} + N_{DCR}$. Assume the weights of DCR data are re-weighted by a constant factor g_w while MT data weights are unchanged. The nRMS takes the following form,

$$nRMS = \sqrt{\frac{1}{N_{MT}+N_{DCR}}\left\{\sum_{i=1}^{N_{MT}}\left(\frac{\mathbf{d}_i^{obs}-F_i(\mathbf{m})}{\sigma_i}\right)^2+\sum_{i=1}^{N_{DCR}}\left(g_w\frac{\mathbf{d}_i^{obs}-F_i(\mathbf{m})}{\sigma_i}\right)^2\right\}},$$
$$nRMS = \sqrt{\frac{1}{N_{MT}+N_{DCR}}\{Q_{MT}^1+Q_{DCR}\}},$$
$$nRMS = \sqrt{\frac{1}{N_{MT}+N_{DCR}}\left\{Q_{MT}^1+g_w^2Q_{DCR}^1\right\}}. \tag{6.2.7}$$

If the data standard errors, σ_i, are correctly estimated, a reasonable fit to the data is obtained for $Q_{MT}^1 \geqslant N_{MT}$ and $Q_{DCR}^1 \geqslant N_{DCR}$. Hence, the desired nRMS should be chosen as,

$$nRMS^{opt} = \sqrt{\frac{1}{N_{MT}+N_{DCR}}\left\{N_{MT}+g_w^2N_{DCR}\right\}}. \tag{6.2.8}$$

6.3 Example of Joint RMT and DCR Inversion

A modified version of 3D checkerboard pattern model [39], used for 3D MT in chapter 4, is selected for demonstrating the joint inversion. The top layer is 100 m thick of resistivity 100 Ω-m while the substratum has resistivity of 10 Ω-m. The top layer, Fig. 6.1, consists of nine conductive and nine resistive blocks (10 and 1000 Ω-m respectively). The top nine blocks are at a depth of $0-20$ m while lower 9 blocks are between $40-70$ m. These two sections ($0-20$ m and $40-70$ m) are referred as L1 and L2 and are sensed by both the data (RMT and DCR). For synthetic data, the model was discretized into $61 \times 71 \times 23$ cells (excluding cells in air for RMT case), with resolution of 10 m in horizontal direction. For the DCR data, 10 profiles were placed, with inter-profile spacing $50-60$ m, covering a length of 480 m (Fig. 6.1). In each profile there were 31 electrodes with a spacing of 20 m. Apparent resistivity were computed for dipole-dipole configurations, with dipole length of 20 m and separation factor n = $1-10$. For RMT data, stations were placed 40 m apart between ± 240 m on each profile(130 total stations). Data (off diagonal impedance tensor) was generated for 10 periods, logarithmically spaced between $10^{-5}-10^{-1}$ s. The combined data set has 5200 and 2350 data points for RMT and DCR respectively. Two percent random noise was added to both data. The error floor for RMT was set at 2% of $|\mathbf{Z}_{xy}\mathbf{Z}_{yx}|^{1/2}$ for off-diagonal impedance elements and for DCR at 2% of apparent resistivity. For all the inversion runs, a homogeneous model of 100 Ω-m, was used as apriori and initial model. The initial model

was discretized into same number of cells as the synthetic model.

Three inversion were performed. First we invert the RMT data only. In 31 *NLCG* iterations the nRMS error reduced from 58.4 to 1.55. The inverted model thus obtained is shown in Fig. 6.2b. Next, we invert the DCR data only. In 34 *NLCG* iterations the nRMS error reduced from 101.5 to 1.82 resulting in inverted model shown in Fig. 6.3a. Finally, RMT and DCR data set were inverted simultaneously. Before inversion, the DCR data set was up-weighted based on the number of data points in each set (scheme 1 as discussed above). In 52 *NLCG* iterations the nRMS error reduced from 97.2 to 2.45 . The inverted model thus obtained is shown in Fig. 6.3b. The inverted model for RMT inversion, DCR inversion and RMT$-$DCR joint inversion will be referred as M1, M2 and M3 respectively in the following discussion.

For comparison of the M1, M2 and M3 we have focused on the model parameters (resistivity of cells) occupied by resistive and conductive blocks. Four such regions are the volume occupied by conductive blocks in L1, resistive blocks in L1, conductive blocks in L2 and resistive blocks in L2. For detailed comparison between three models, we have drawn 1D sections, along $z-$ and $y-$ direction (Fig. 6.4 and 6.5) similar to those drawn for **'Rubick'** model (chapter 4). Also number of model parameters, in percentage, (ratio of cells in particular resistivity range to total number of cells in a volume), in various resistivity ranges are computed. For example, consider all the cells in the volume occupied by 4 conductive blocks in L1. The percentage of these cells are computed for different resistivity ranges. Such a task is performed for all the four regions and represented in stem plot (Fig. 6.6) as well as in tabular form (Table 6.1).

In models M1, M2 and M3 all the conductive and the resistive blocks in L1 are recovered very well. The lateral dimensions of these blocks are well demarcated. Approximately 31% of the cells of M1 belonging to conductive blocks lies in the range $5-15$ Ω-m while this is approximately 50% for both M2 and M3. For M1, the resistivity of conductive blocks is more spread as compared to M2 and M3. For resistive blocks there is a difference in models M1, M2 and M3. In M1 the resistivity of the resistive blocks in L2 hardly exceeds 600 Ω-m while for M2 and M3 it even exceeds 1000 Ω-m. Thus in M1 resistivity values are under-estimated, as expected, while for M2 and M3 they are over-estimated. Almost 30% of the parameters, in M2 and M3, have resistivity more than 1000 Ω-m. In terms of imaging capability for shallow conductive features RMT and DCR gives same results while

for resistive features DCR gives better result as compared to RMT. For the shallow features there seems to be no superiority of joint inversion over individual inversion as DCR inversion alone gives result similar to RMT-DCR inversion.

There is a marked difference in the imaging of conductive and the resistive blocks in L2 region in these three models. The base of conductive blocks is smeared more in M1 and M2 as compared to M3. Also the top of these blocks is at shallow depth in M2 as compared to M1 and M3. Only 5% of the cells in the volume occupied by conductive blocks in L2 lies between $5-10$ Ω-m in M1 while 13% and 19% for M2 and M3 respectively. Also, as evident from the stem plot in terms of imaging of conductive features at intermediate depth, RMT$-$DCR joint inversion gives better result as compared to RMT or DCR inversion only. There is a striking difference in the resolution of resistive blocks in L2. In M1, they are not recovered. Almost 86% of the cells in this volume lies between $50-150$ Ω-m. This can cause a misinterpretation if this data set was from a field survey. In M2, these features are better recovered as compared to M1. These resistivity reaches upto 350 Ω-m. However, in M3 these blocks are very well recovered. Also the resistivity reaches upto 550 Ω-m. From Fig. 6.4 and 6.5, it is observed that resistive features are better resolved in M3 as compared to M2 and M1. The same is evident from Fig. 6.6d.

In M1, the substratum is recovered but the conductive structures of L2 and substratum are not resolved. This may be due to the less number of periods per decade. For M2, the substratum was not resolved at all. From this inversion result one can infer that joint inversion results in better inverse models as compared to individual inversion.

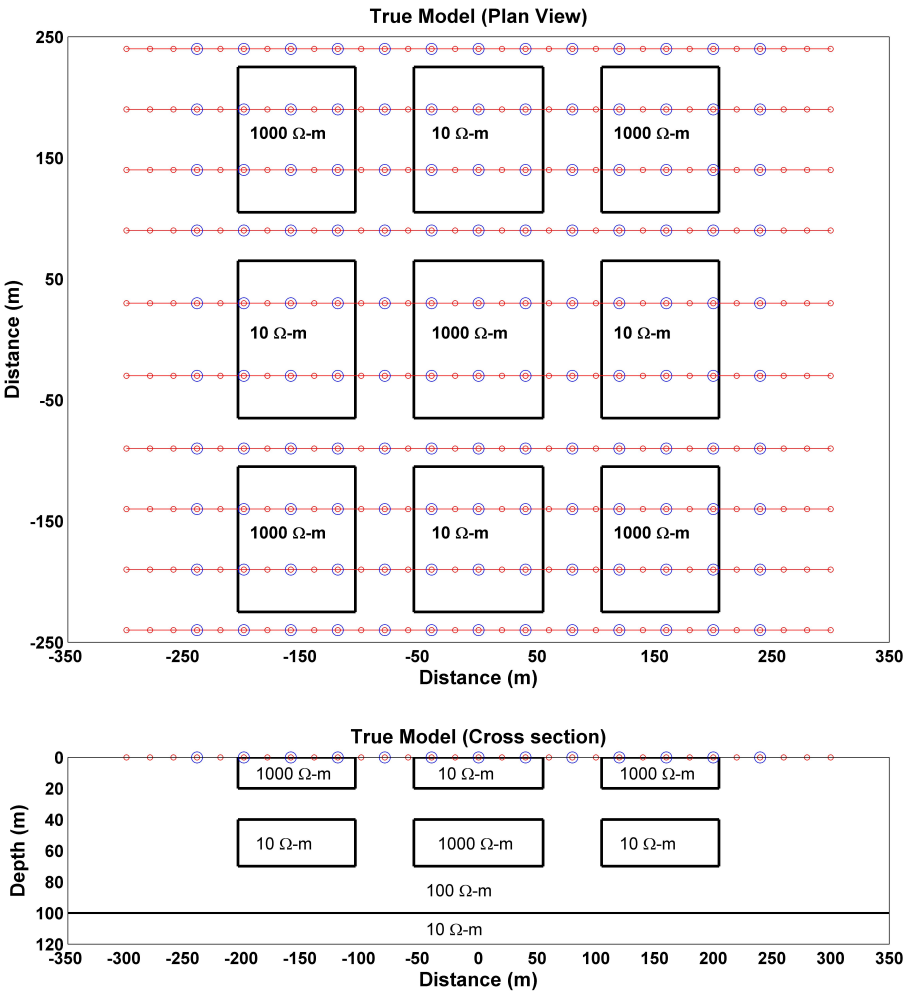

Figure 6.1: (top) Plan view of resistivity model for 3D test, with circles (red) representing electrodes and circles (blue) representing RMT station, and (bottom) Cross-section view of the model at $x = 0$ profile.

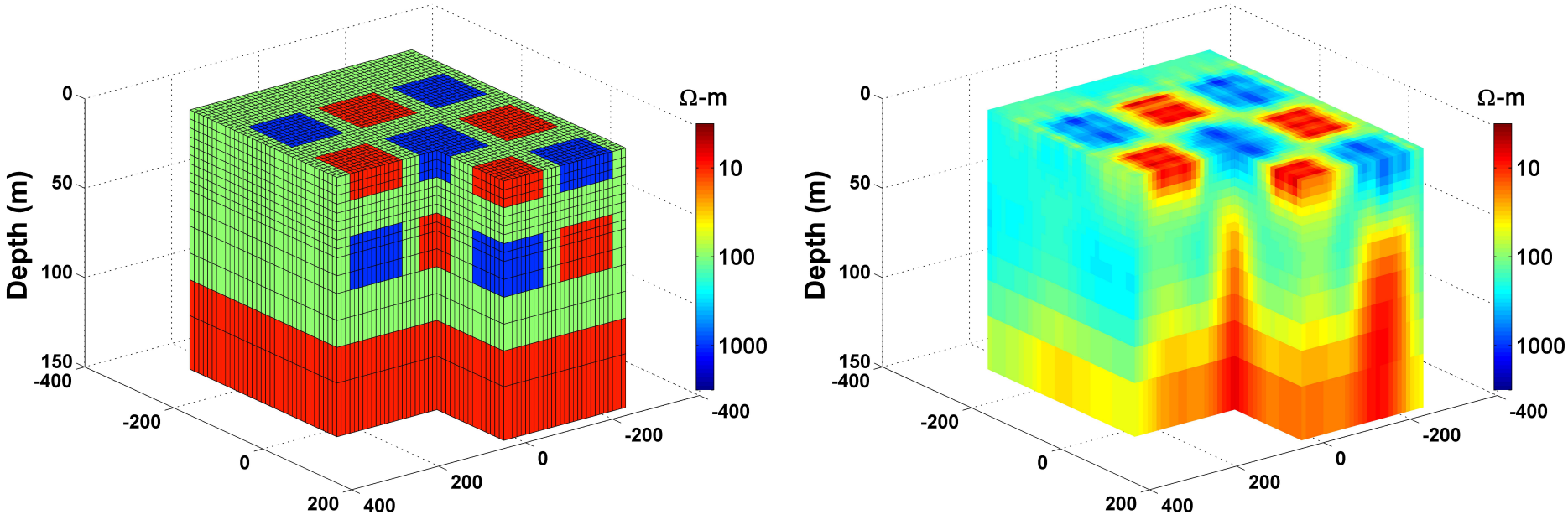

Figure 6.2: (a) Resistivity model used to generate synthetic data for 3D RMT test, and (b) Inverse model obtained after 31 NLCG iterations with RMT data only. Note that in the cut-away view the upper surface shown is at 10 m depth, but the structures shown extend to the surface.

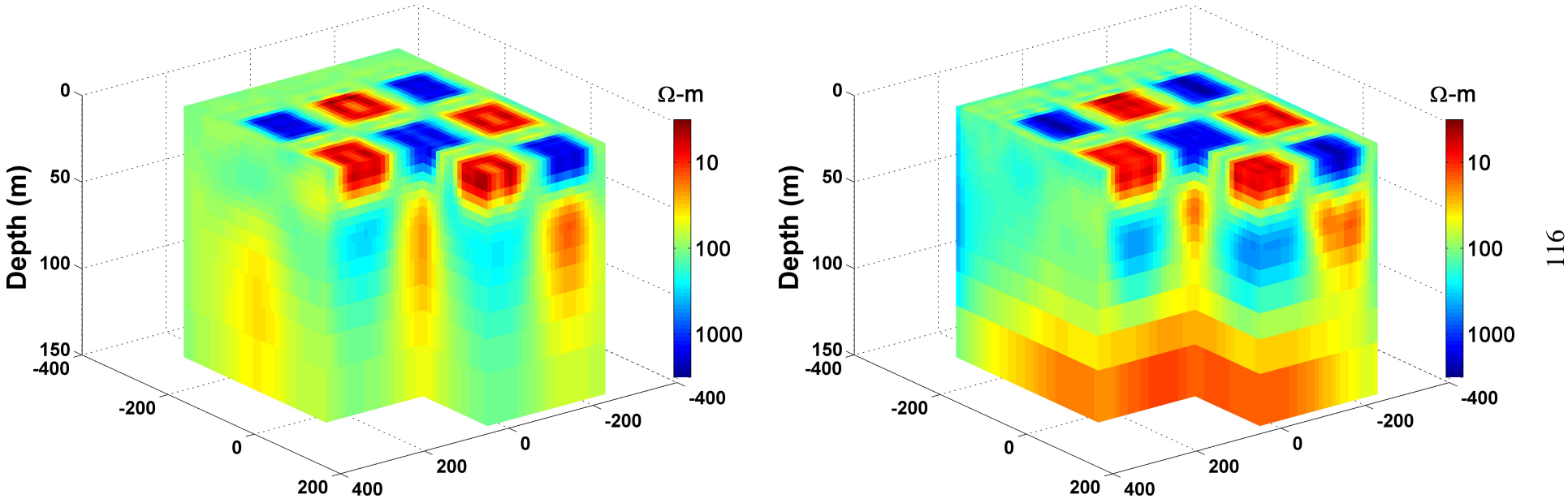

Figure 6.3: (a) Inverse model obtained after 34 NLCG iterations with DCR data only, and (b)joint inverse model obtained after 52 NLCG iterations with RMT and DCR data. Note that in the cut-away view the upper surface shown is at 10 m depth, but the structures shown extend to the surface.

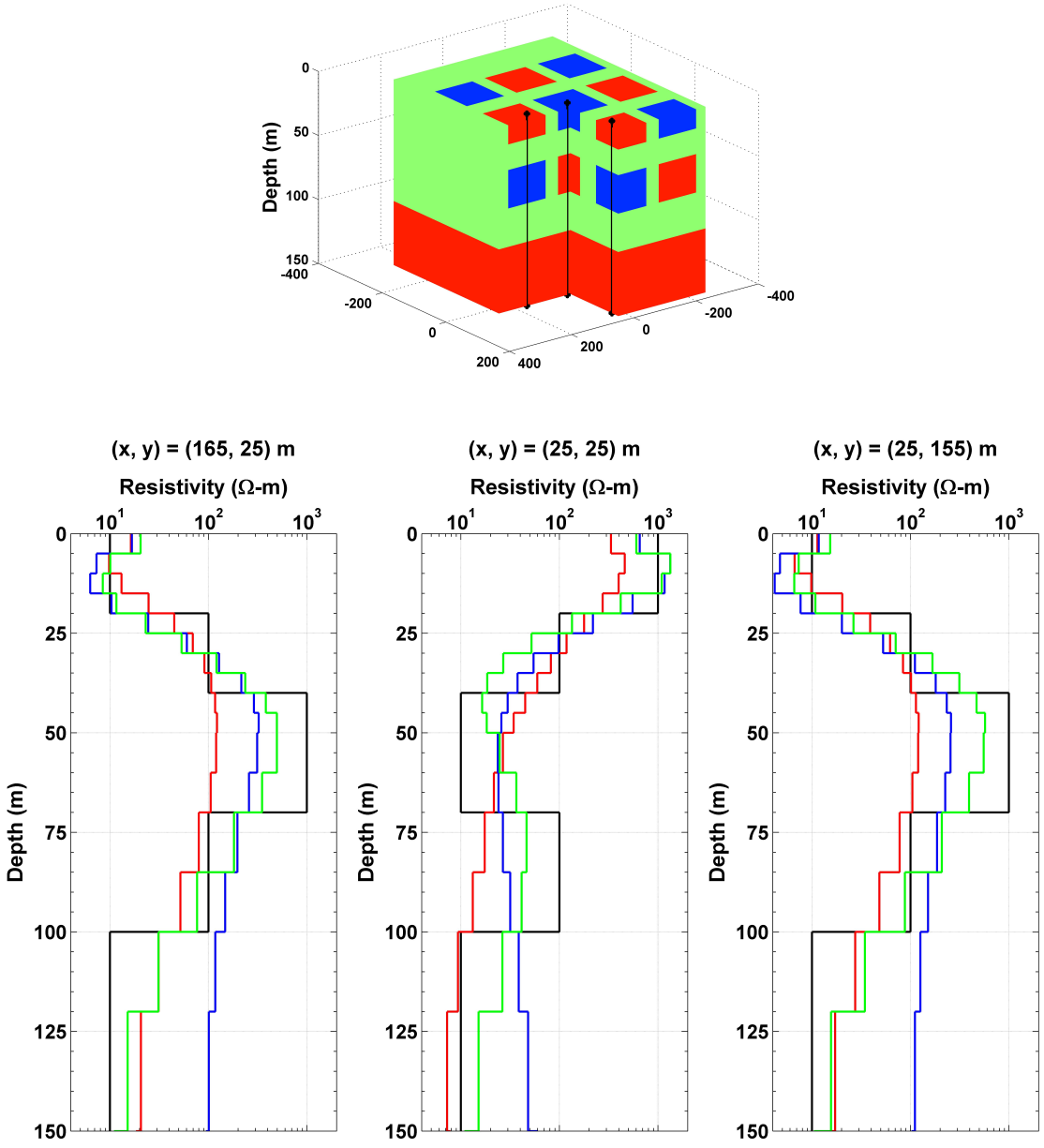

Figure 6.4: 1D section of true model (black), inverted model using RMT data only (red), inverted model using DCR data (blue), and RMT-DCR joint inversion (green) at three different $(x,\ y)$ points.

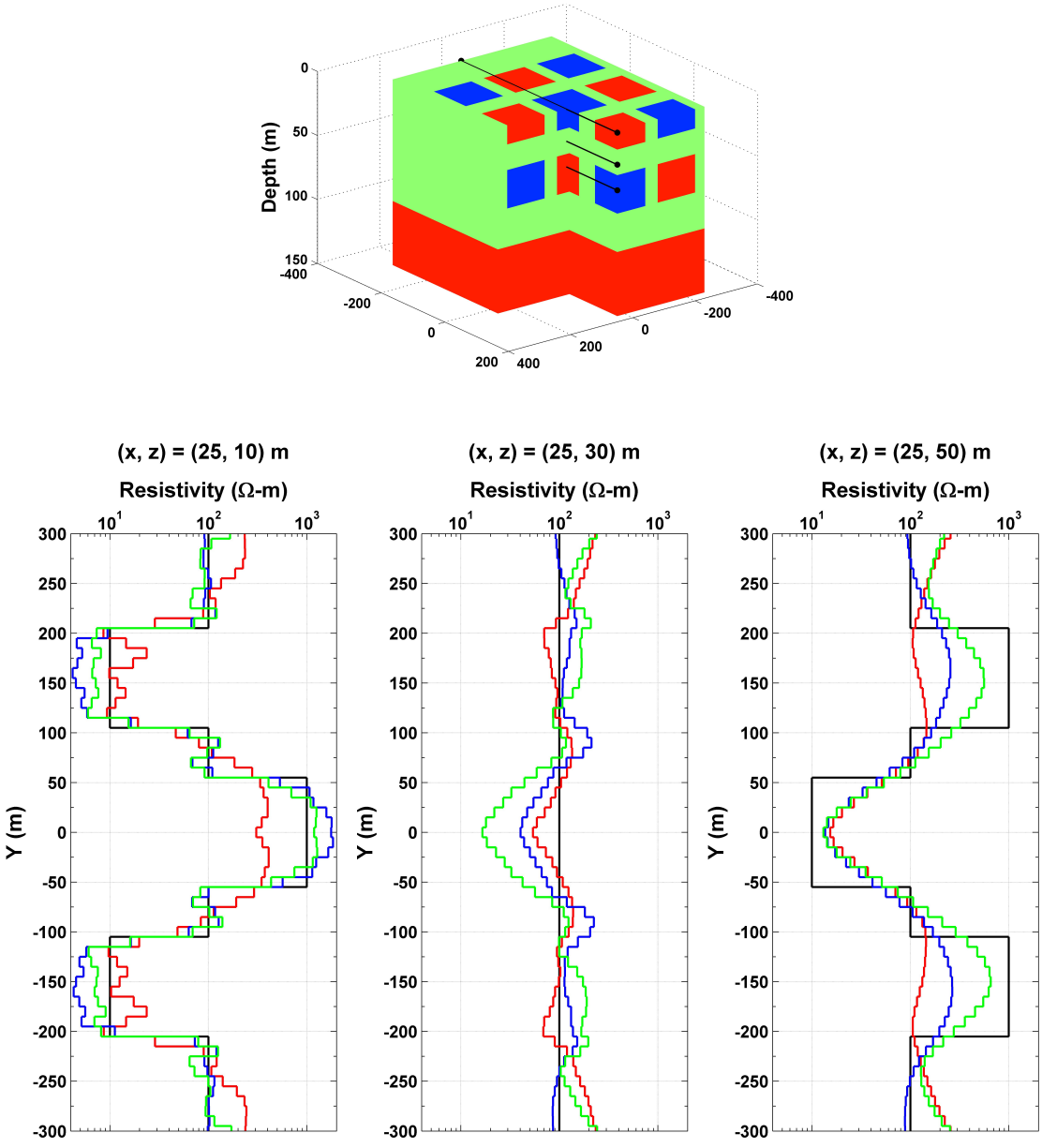

Figure 6.5: 1D section of true model (black), inverted model using RMT data only (red), inverted model using DCR data (blue), and RMT−DCR joint inversion (green) at three different $(x,\ z)$ points. This section passes through all the three different anomalies

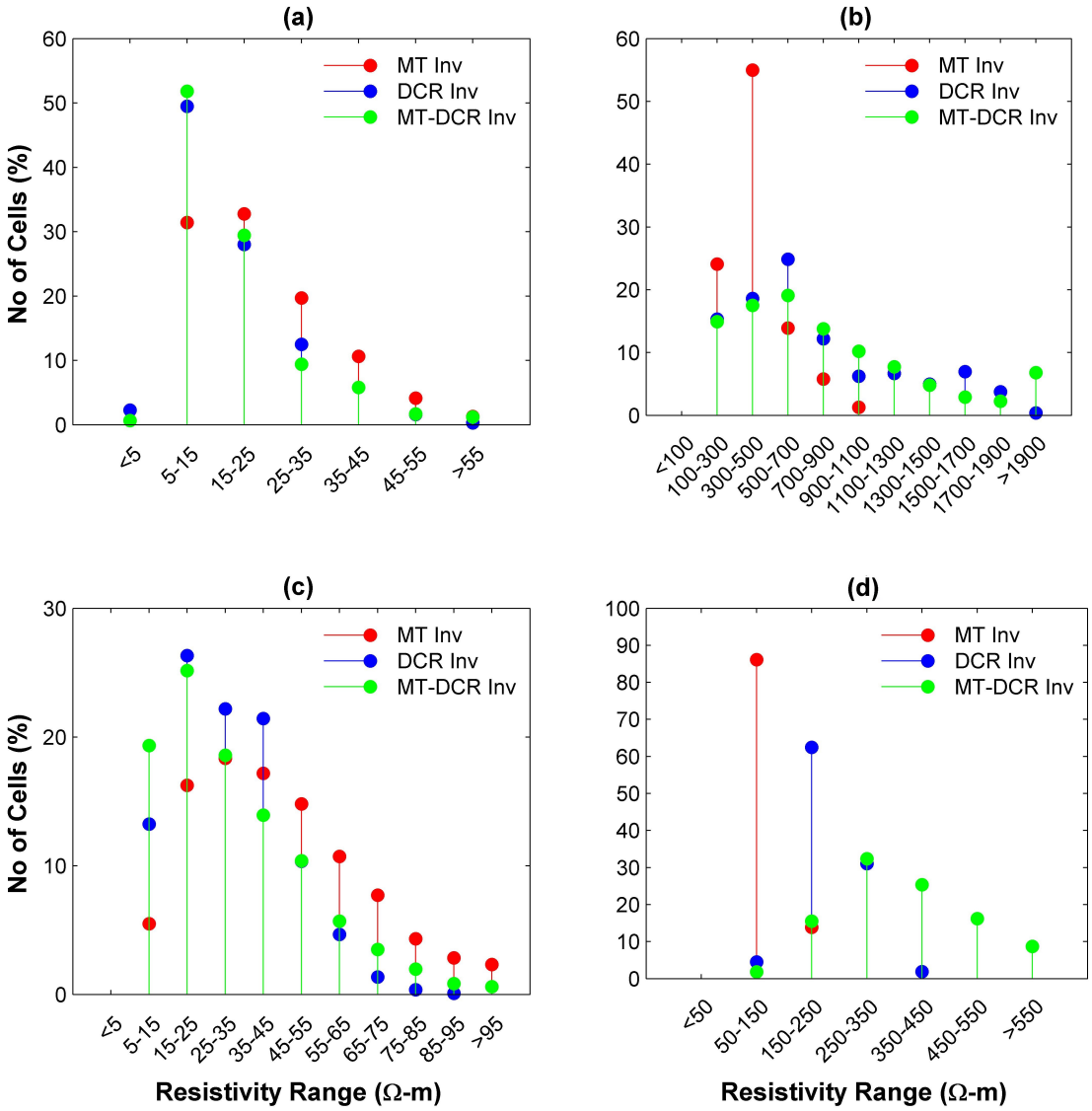

Figure 6.6: Stem plot showing percentage of cell in various resistivity ranges for (a) volume occupied by conductive blocks in L1, (b) resistive blocks in L1, (c) conductive blocks in L2, and (d) resistive blocks in L2.

Table 6.1: Table showing information about number of model parameters and resistivity range for inversion of MT, DCR and MT-DCR joint inversion. These model parameters corresponds to the cell of resistive and conductive blocks only.

Res. Range (Ω-m)	**MT Inv.** (%)	**DCR Inv.** (%)	**MT−DCR Inv.** (%)
Cells for Conductive Blocks in L1			
< 5	–	2.29	0.67
$5 - 15$	31.44	49.48	51.81
$15 - 25$	32.78	28.01	29.44
$25 - 35$	19.70	12.50	9.40
$35 - 45$	10.64	5.82	5.82
$45 - 55$	4.15	1.62	1.67
> 55	1.29	0.29	1.19
Cells for Resistive Blocks in L1			
$100 - 300$	24.08	15.33	14.93
$300 - 500$	54.98	18.62	17.54
$500 - 700$	13.88	24.88	19.10
$700 - 900$	5.78	12.20	13.76
$900 - 1100$	1.28	6.22	10.19
$1100 - 1300$	–	6.66	7.74
$1300 - 1500$	–	4.98	4.82
$1500 - 1700$	–	6.94	2.89
$1700 - 1900$	–	3.77	2.25
> 1900	–	0.40	6.78
Cells for Conductive Blocks in L2			
$5 - 15$	5.50	13.24	19.34
$15 - 25$	16.25	26.32	25.16
$25 - 35$	18.34	22.19	18.58
$35 - 45$	17.17	21.43	13.92
$45 - 55$	14.81	10.35	10.39
$55 - 65$	10.71	4.65	5.70
$65 - 75$	7.70	1.36	3.49
$75 - 85$	4.33	0.36	1.97
$85 - 95$	2.85	0.08	0.84
> 95	2.33	–	0.60
Cells for Resistive Blocks in L2			
$50 - 150$	86.12	4.53	1.81
$150 - 250$	13.88	62.45	15.55
$250 - 350$	–	31.11	32.40
$350 - 450$	–	1.91	25.38
$450 - 550$	–	–	16.17
> 550	–	–	8.68

In the present model large number of RMT sites are used. Although additions of additional sites does not add much computational burden if the new sites are within the modeling domain (as in the present case). However, acquisition of additional sites will be time consuming and costly. Keeping this in mind, RMT stations were reduced from 130 to just 15 (inter station spacing $100 - 150$ m) thus representing a more realistic field scenario. Out of these 15 sites 9 are over the blocks and rest 6 are over the halfspace. The total number of RMT data points were reduced to 600 only. For the combined data set three inversion runs were performed: *(i)* Run 1, without data re-weighting, *(ii)* Run 2, with data re-weighting based on number of data points in the individual data sets, and *(iii)* Run 3, data re-weighting based on norms of gradient of each data set computed individually.

Run 1

For this run no data scaling was used. In 72 *NLCG* iterations the nRMS error reduced from 95.3 to 2.3. In the inverted model, Fig. 6.7a, all the main features are recovered. However, visually the quality of inverted model is poor as compared to inverse result obtained with data re-weighting (cf Fig. 6.7b-c). The resistivity of substratum is less than that obtained in Run 2 & 3. This is due to the fact that DCR data was more as compared to RMT, hence, it overshadowed RMT data set.

Run 2

As DCR data was more in number, RMT data weights were up-weighted by a factor of 1.97 (see Section 6.2.2 for the definition of factor) and the desired error floor was modified to 1.09 from 1. In 62 *NLCG* iterations the nRMS error reduced from 107.9 to 2.9. In the inverted model, Fig. 6.7b, all the main features are recovered. The obtained resistivity of the substratum is more as compared to that obtained in Run 1. This is due to the fact that more weight was given to the RMT data as compared to in Run 1 hence, the overshadowing of DCR data over RMT data is reduced.

Run 3

For data scaling, norm of gradient for individual data set were computed and RMT data weights were up-weighted by a factor of 1.42 and the desired error floor was modified to 1 from 1.04. In 72 *NLCG* iterations the nRMS error reduced from 100.5 to 2.4. In the inverted

model, Fig. 6.7c, all the main features are recovered. The obtained inverse model is similar to that in Run 2. However, resistivity of the substratum is more as compared to that obtained in Run 2. This is due to the fact that weight was given to the RMT data is less than those given in Run 2.

Fig. 6.8 $-$ 6.9 (only for 12 sites, 9 sites over blocks and 3 over half space) gives the misfit plot of apparent resistivity and phase for Z_{xy} and Z_{yx} for Run 1, Run 2 and Run 3. From these figures it is evident that data scaling gives better inversion result as compared to no data scaling. For example, the sites that are over the resistive blocks (site number 002, 004, 008, 012 and 014) misfit is phases is less for Run 2 and Run 3 as compared to Run 1 for intermediate period. For the sites that are over the conductive blocks (site number 003, 007, 009, 013) misfit is phases is less for Run 2 and Run 3 as compared to Run 1 for periods less than 10^{-2} s. Fig. 6.10 shows comparison of observed and predicted data for a profile at $x = +30$ in pseudosection for the three runs. No remarkable differences are seen in the pseudosection. This maybe due to the fact that the top features are imaged similarly in the three runs. In the present case, the DCR data is not sensitive to the substratum (due to limited depth of investigations) hence, the indifference.

In the three runs, visually the quality of inverted models is inferior as compared to one with 130 RMT stations (Fig. 6.3b). This is due to the fact that inter-station spacing plays a crucial role in the reconstruction of the model. But, with only few RMT stations one can improve the inversion results and also recover deeper features.

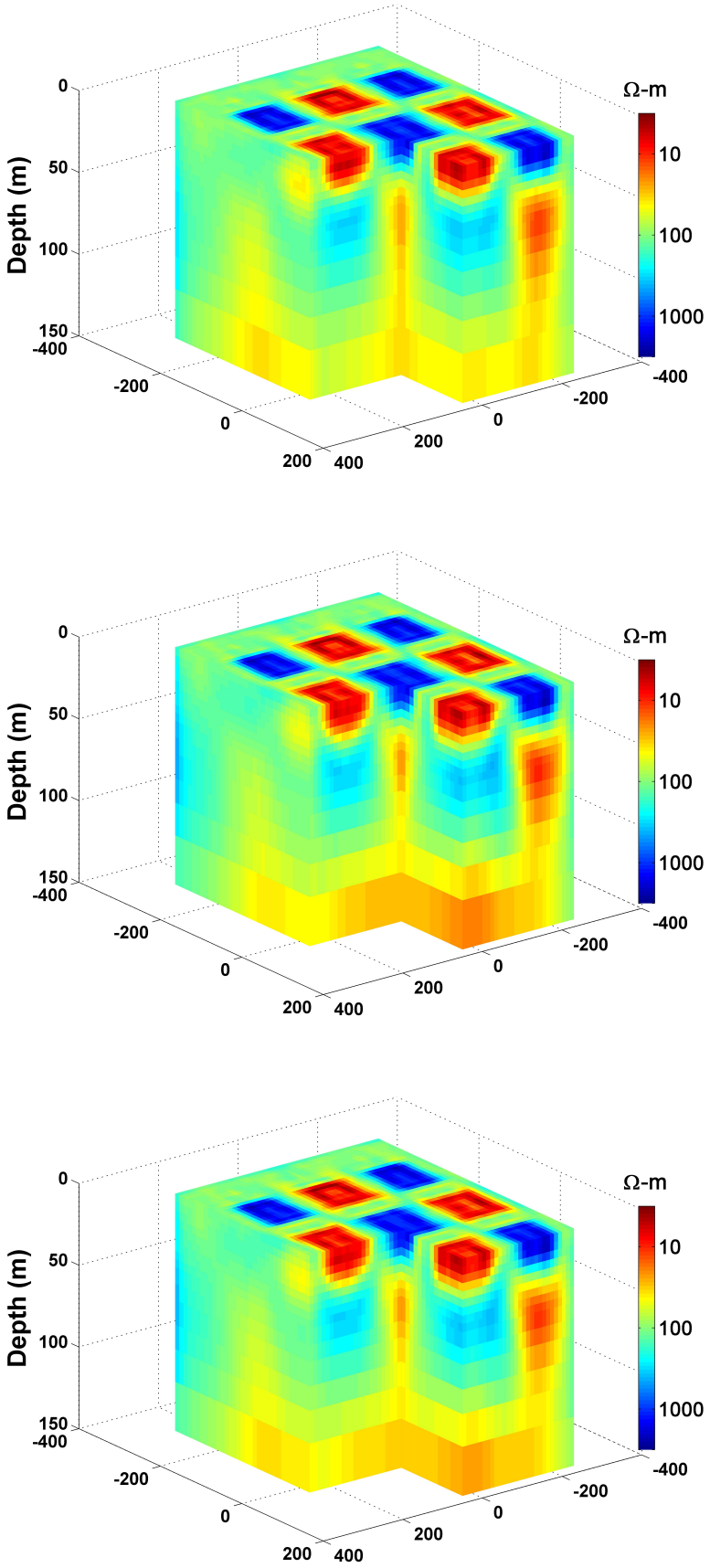

Figure 6.7: (from top) Inverted model for Run 1, 2 and 3. Note that in the cut-away view the upper surface shown is at 10 m depth, but the structures shown extend to the surface.

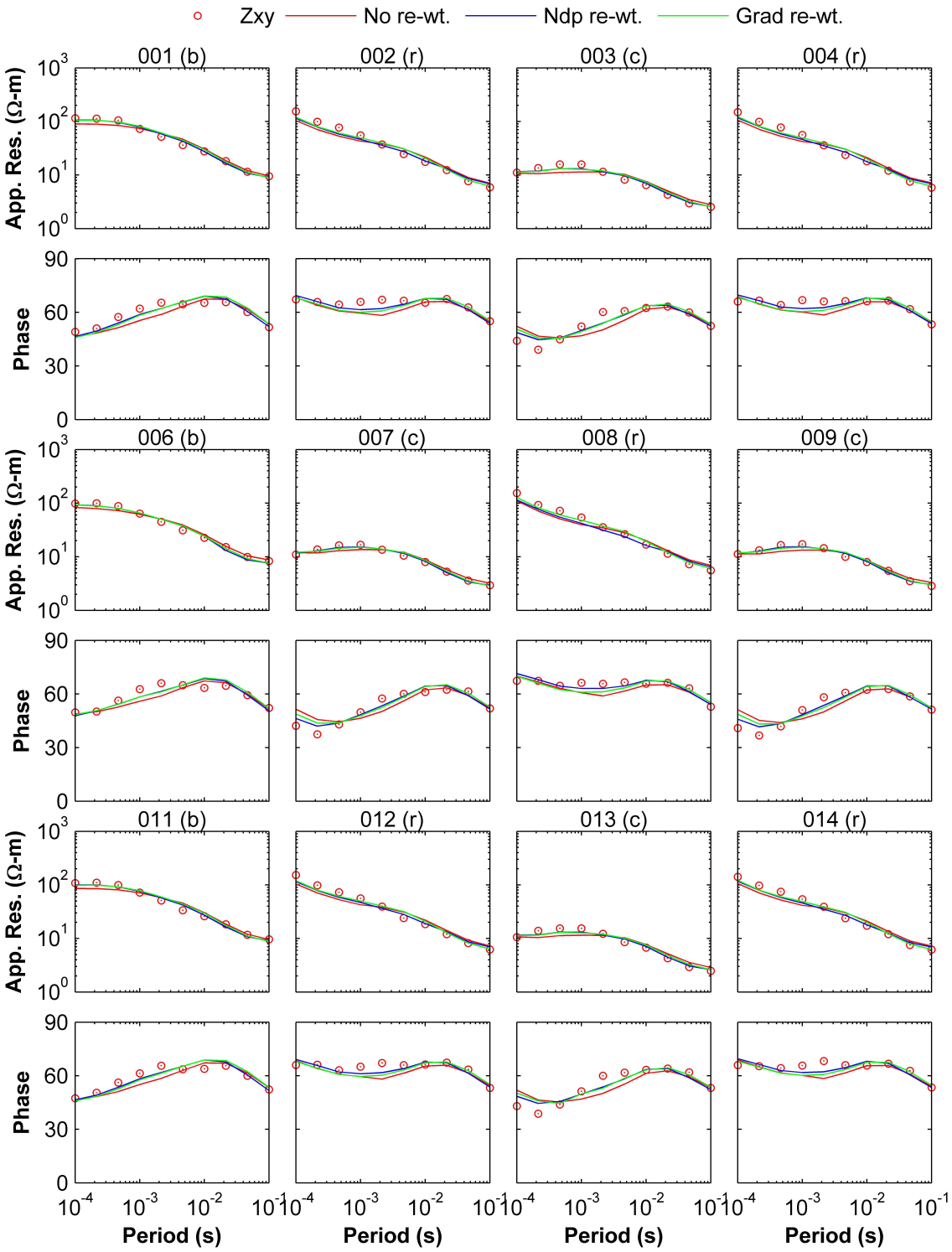

Figure 6.8: Misfit plot of apparent resistivity and phase for Run 1, Run 2 and Run 3 for Z_{xy} elements. Red circles represents the observed data whereas lines (red, blue and green) represents computed data for the inverse models (Run 1, Run 2 and Run 3). The suffix in parenthesis after the sites name indicate the location of the site over type of zone; 'b'-background, 'c'-conductive, and 'r'-resistive.

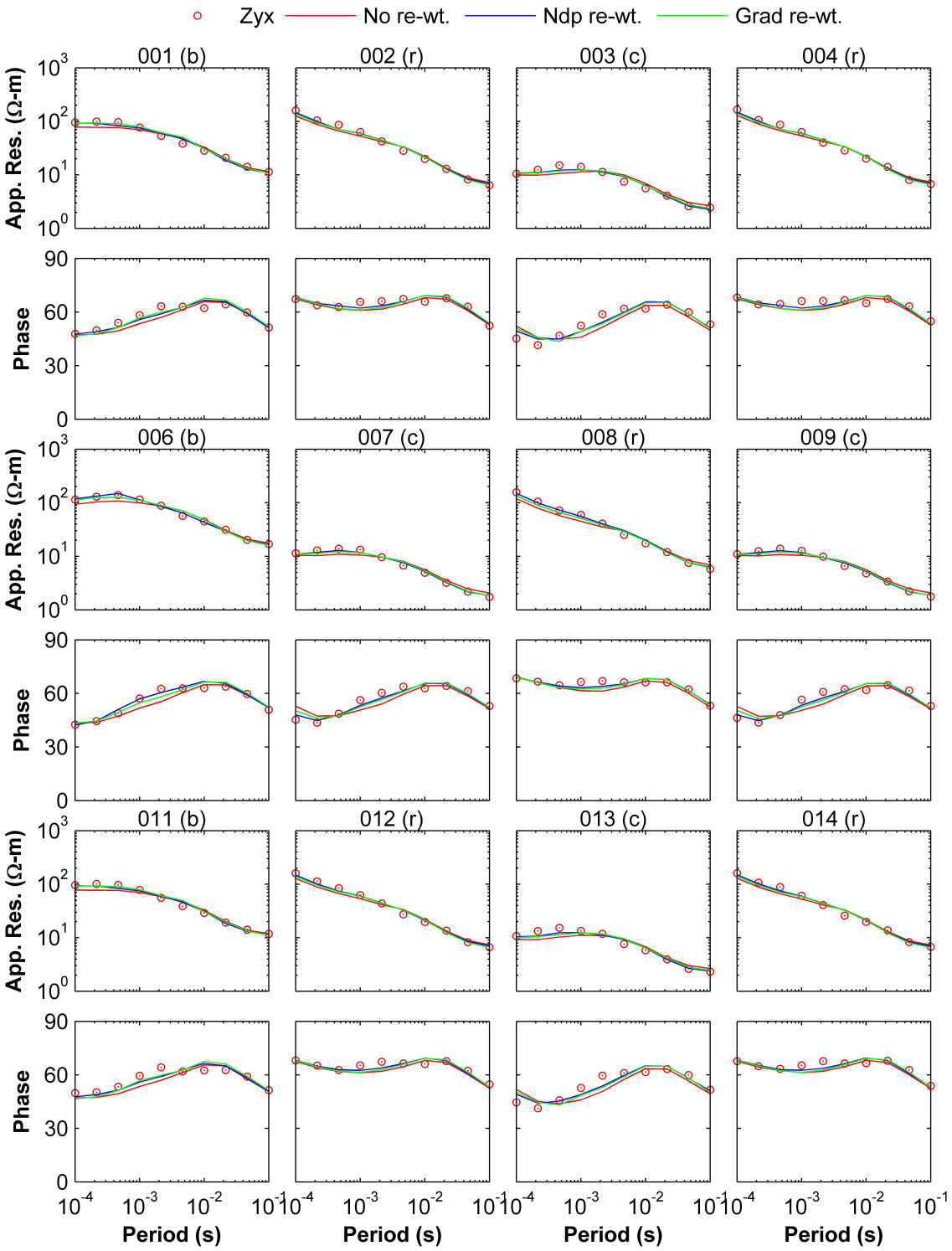

Figure 6.9: Misfit plot of apparent resistivity and phase for Run 1, Run 2 and Run 3 for Z_{yx} elements. Red circles represents the observed data whereas lines (red, blue and green) represents computed data for the inverse models (Run 1, Run 2 and Run 3). The suffix in parenthesis after the sites name indicate the location of the site over type of zone; 'b'-background, 'c'-conductive, and 'r'-resistive.

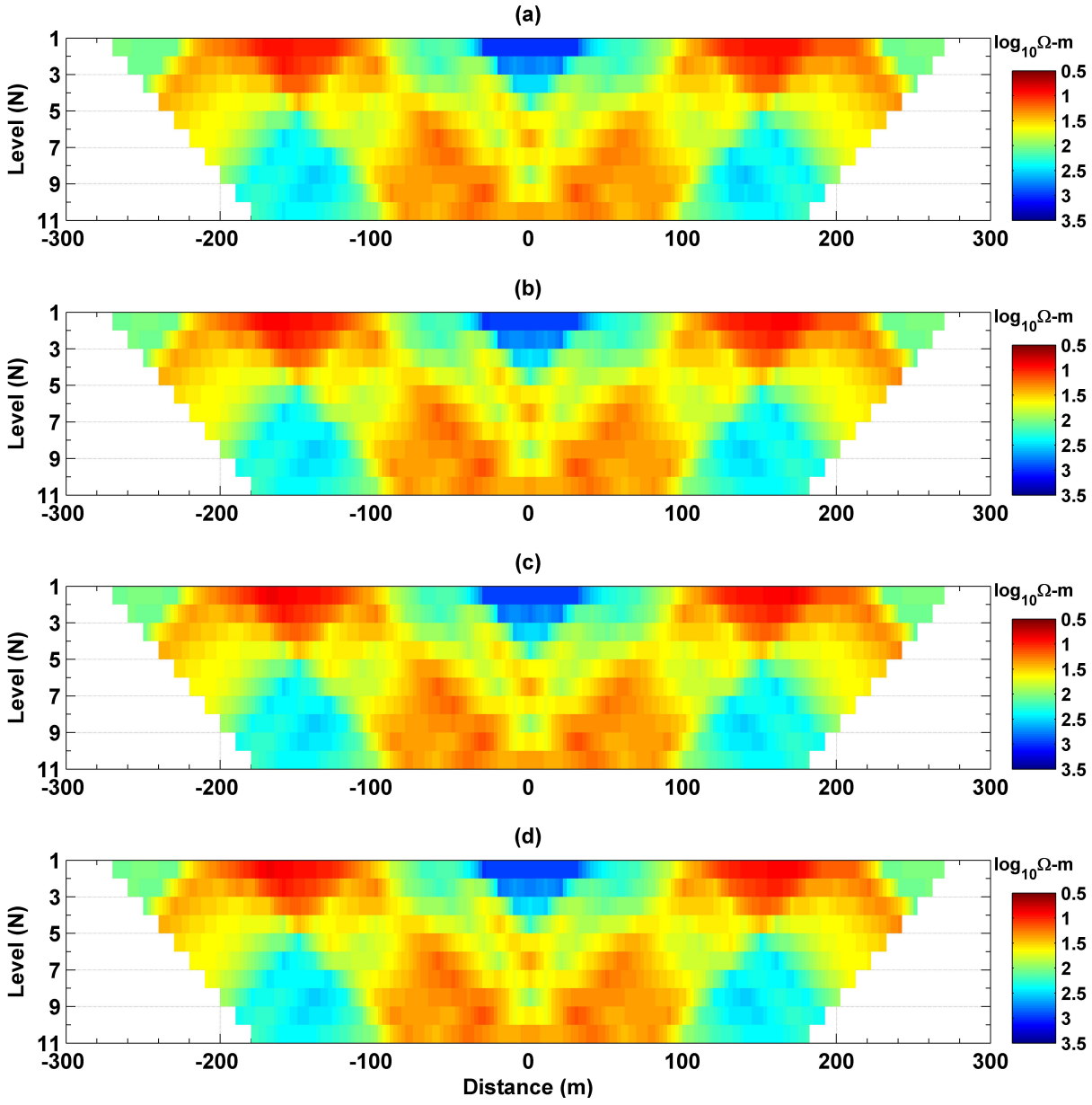

Figure 6.10: (a) Synthetic Dipole-Dipole array data shown in pseudosection, and predicted Dipole-Dipole array data of the inverse model for (b) Run 1, (c) Run 2, and (d) Run 3 for $x = +30$ m profile.

6.4 Summary

AP3DMT code is extended to *AP3DMT-DC* in which the individual or joint inversion of MT and DCR data is included. Through synthetic experiment it is demonstrated that joint inversion results in a better inverse models as compared to individual inversion. It is observed that data scaling proves to be very crucial. Data re-weighting based solely on the number of data points in each dataset will give more contribution to the data set with less data points. Data re-weighting is trival to compute but it does not give any weightage to the sensitivity of data weight. Here, we have presented sythetic examples only. Test on field data will be conducted in the future work.

Chapter 7

SUMMARY AND CONCLUSION

3D inversion of Magnetotelluric (MT) data is a very large scale and computationally intensive task. The mandate of this study was to develop an accurate, efficient and user friendly, MATLAB based 3D forward modeling and inversion algorithm for MT data. The developed code, *AP3DMT*, is in modular form where, the basic components of inversion − forward modeling, model regularization, data functionals, and sensitivity computation are standalone thus making the code easy to modify as per the requirement. A brief summary of developed algorithms is as follows.

A 3D forward modeling algorithm for MT based on finite difference method is developed. The accuracy of the developed forward modeling algorithm is rigorously tested and verified. In the first phase of development of inversion algorithm, a 3D inversion algorithms based on Non-Linear Conjugate Gradient and Gauss-Newton optimization technique (model space and data space) were developed for 3D MT data. The algorithms were rigorously tested on several synthetic and a real field data set.

During the development of divergence correction sub-program, we identified that it can be developed into a full fledged 3D DCR inversion code. Taking the advantage of the modular form of *AP3DMT*, we modified only the forward modeling and Jacobian part to accommodate the DCR inversion in the original code. The DCR forward modeling is based on finite difference method. For removal of singularity due to a point source primary/secondary approach is used. For solving the system matrix, pre-conditioned iterative solver or direct solver can be used, depending on the size of the matrix and number of sources. The inversion scheme is similar to MT, where sensitivity forward calculation are used for the sensitivity matrix. Both forward and inverse modeling is tested rigorously on

different models.

Once we have developed inversion codes for MT and DCR inversion, we implemented joint inversion of MT and DCR data to improve the interpretation. Through a synthetic example it is demonstrated that the joint inversion results in a better model than inversion of individual dataset. Two different data weighting schemes for joint inversion are tested.

The conclusions drawn from this study are given in next section.

7.1 Conclusion

The new developed code, *AP3DMT* on MATLAB platform is versatile, robust, flexible and user friendly. It can be used to invert just the MT data or the DCR data, or invert both data sets simultaneously. The MATLAB platform provides a powerful computation environment along with extensive numerical libraries; data visualization capabilities.

The versatility of the grid generator is demonstrated for a two different type of models (target bodies represented using cuboids and polyhedrons) and construction of resistivity matrix is presented for a spiral body embedded in a half space.

The forward modeling responses for **3d2** model computed using *AP3DMT* are in good agreement with the responses obtained using *ModEM*. The absolute relative error in amplitude of off-diagonal impedance is less than 2% while for phases it is less than 1.5%.

Through synthetic experiment on a two block model it is demonstrated that all three inversion (*NLCG*, *CG-MS* and *CG-DS*) results in the similar inverse model. The difference is the convergence rate and computation time. For the test model, it was found that convergence of *CG-DS* (9 iterations) is slow as compared to *CG-MS* (6 iterations) for same parameters. Further this convergence rate depends on the number of iterations used in the *CG* routine. If the matrix equations (inversion) is not solved to an appropriate accuracy, the inversion does not converge to a desired level (nRMS value = 1). If the matrix equation is solved at higher tolerance (by calling more *CG* iterations), it will increase the computation time although the desired misfit may reach in less number of inversion iterations. To strike a balance, we found after initial 3-4 inversion iterations, *CG* iterations should be increased in order to reach desired misfit level. As compared to *GN*, *NLCG* requires more number of iterations (47 in total) but less computation time.

Further, the inversion of two different synthetic data sets (computed using *ModEM*)

demonstrates the accuracy of the developed code *AP3DMT*. For the **Rubick** model, main features were recovered. For the **DTM2** model, the circular shape of the hemisphere was recovered although the correct representation of the circular body is limited due to the use of rectangular meshing. However, the deeper part of the structure is not delineated. In this model, the nRMS value reaches the desired value, but a fit of responses for different sites and different periods reveals the fact that for sites that lie away from the edge of the hemisphere, there is an excellent agreement between the observed and the computed responses as compared to sites that are close to the edge of the hemisphere. Further, fit of longer periods is better as compared to shorter periods. This indicates that one should not rely on a single nRMS value but more carefully examine the misfit over all data space. It was observed that the inverted model, using Roorkee-Gangotri field data, obtained using *AP3DMT* closely matches the model obtained using *ModEM* for the same data set.

For DCR inversion, *GN-MS* proves to be robust. For the test model, similar to a mine dump, we were able to recover the shape and size of the anomaly.

The joint inversion results in better inverse model as compared to individual inversion. For the test model, there is marked improvement in the resolution of the resistive features in the second layer. It is observed that data scaling proves to be very crucial else the influence of more numerous data set of one type will become significant and it will overshadow data set of second type. Data re-weighting, based solely on the number of data points in each dataset or norms of the gradients of the data misfit will give more contribution to the data set with less data points thus reducing biasness of data set with more number of data point.

7.2 Scope for Further Research

This study has produced efficient, reliable and robust algorithms for analysis and interpretation of MT and DCR data. There are several possibilities of further extension of the work done in this study on the following lines:

Present algorithms work for isotropic medium and these can be modified for anisotropic medium.

The algorithm can compute responses using partitioning of fields, hence forward modeling of controlled source EM methods can be added simply by adding primary field response computation subprogram for the given EM source.

Present algorithms for MT [151] and CSEM [29] are standalone and these can be combined together for joint inversion of CSEM-MT data.

Appendix A

Finite difference approximation of Governing equation For MT

Governing equation of EM modeling is given as,

$$\bigtriangledown \times \bigtriangledown \times \mathbf{E} + i\omega\mu_0\sigma\mathbf{E} = 0, \tag{A-1}$$

This vector PDE can be decomposed into three scaler PDEs and these equations are written as,

$$-\frac{\partial^2 \mathbf{E}_x}{\partial y^2} - \frac{\partial^2 \mathbf{E}_x}{\partial z^2} + \frac{\partial^2 \mathbf{E}_y}{\partial x \partial y} + \frac{\partial^2 \mathbf{E}_z}{\partial x \partial z} + k^2 \mathbf{E}_x = 0, \tag{A-2}$$

$$-\frac{\partial^2 \mathbf{E}_y}{\partial x^2} - \frac{\partial^2 \mathbf{E}_y}{\partial z^2} + \frac{\partial^2 \mathbf{E}_x}{\partial x \partial y} + \frac{\partial^2 \mathbf{E}_z}{\partial y \partial z} + k^2 \mathbf{E}_y = 0, \tag{A-3}$$

$$-\frac{\partial^2 \mathbf{E}_z}{\partial x^2} - \frac{\partial^2 \mathbf{E}_z}{\partial y^2} + \frac{\partial^2 \mathbf{E}_x}{\partial x \partial z} + \frac{\partial^2 \mathbf{E}_y}{\partial y \partial z} + k^2 \mathbf{E}_z = 0, \tag{A-4}$$

where $k^2 (= i\omega\mu_0\sigma)$ denotes wavenumber, subscripts x, y, z denote the components of electric field in x, y, z directions respectively. Using the central difference formula on staggered grid, eq A-2, A-3 and A-4 for i, j, k node can be approximated as:

$$
\begin{aligned}
&\left[\frac{2}{\triangle y_{j-1}\triangle y_j}+\frac{2}{\triangle z_{k-1}\triangle z_k}+k^2(i+\frac{1}{2},j,k)\right]\mathbf{E}_x(i+\frac{1}{2},j,k)\\
&-\left[\frac{1}{\triangle y_{j-1}\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_x(i+\frac{1}{2},j-1,k)-\left[\frac{1}{\triangle y_j\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_x(i+\frac{1}{2},j+1,k)\\
&-\left[\frac{1}{\triangle z_{k-1}\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_x(i+\frac{1}{2},j,k-1)-\left[\frac{1}{\triangle z_k\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_x(i+\frac{1}{2},j,k+1)\\
&+\left[\frac{1}{\triangle x_i\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_y(i,j-\frac{1}{2},k)-\left[\frac{1}{\triangle x_i\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_y(i,j+\frac{1}{2},k)\\
&-\left[\frac{1}{\triangle x_i\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_y(i+1,j-\frac{1}{2},k)+\left[\frac{1}{\triangle x_i\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_y(i+1,j+\frac{1}{2},k)\\
&+\left[\frac{1}{\triangle x_i\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_z(i,j,k-\frac{1}{2})-\left[\frac{1}{\triangle x_i\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_z(i,j,k+\frac{1}{2})\\
&-\left[\frac{1}{\triangle x_i\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_z(i+1,j,k-\frac{1}{2})+\left[\frac{1}{\triangle x_i\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_z(i+1,j,k+\frac{1}{2}) \qquad =0,
\end{aligned}
$$

$$
\begin{aligned}
&\left[\frac{2}{\triangle x_{i-1}\triangle x_i}+\frac{2}{\triangle z_{k-1}\triangle z_k}+k^2(i,j+\frac{1}{2},k)\right]\mathbf{E}_y(i,j+\frac{1}{2},k)\\
&-\left[\frac{1}{\triangle x_{i-1}\triangle x_{i+\frac{1}{2}}}\right]\mathbf{E}_y(i-1,j+\frac{1}{2},k)-\left[\frac{1}{\triangle x_i\triangle x_{i+\frac{1}{2}}}\right]\mathbf{E}_y(i-1,j+\frac{1}{2},k)\\
&-\left[\frac{1}{\triangle z_{k-1}\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_y(i,j+\frac{1}{2},k-1)-\left[\frac{1}{\triangle z_k\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_y(i,j+\frac{1}{2},k+1)\\
&+\left[\frac{1}{\triangle y_j\triangle x_{1+\frac{1}{2}}}\right]\mathbf{E}_x(i-\frac{1}{2},j,k)-\left[\frac{1}{\triangle y_j\triangle x_{1+\frac{1}{2}}}\right]\mathbf{E}_x(i+\frac{1}{2},j,k)\\
&-\left[\frac{1}{\triangle y_j\triangle x_{1+\frac{1}{2}}}\right]\mathbf{E}_x(i-\frac{1}{2},j+1,k)+\left[\frac{1}{\triangle y_j\triangle x_{1+\frac{1}{2}}}\right]\mathbf{E}_x(i+\frac{1}{2},j+1,k)\\
&+\left[\frac{1}{\triangle y_j\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_z(i,j,k-\frac{1}{2})-\left[\frac{1}{\triangle y_j\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_z(i,j,k+\frac{1}{2})\\
&-\left[\frac{1}{\triangle y_j\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_z(i,j+1,k-\frac{1}{2})+\left[\frac{1}{\triangle y_j\triangle z_{k+\frac{1}{2}}}\right]\mathbf{E}_z(i,j+1,k+\frac{1}{2}) \qquad =0,
\end{aligned}
$$

$$
\begin{aligned}
&\left[\frac{2}{\triangle x_{i-1}\triangle x_i}+\frac{2}{\triangle y_{j-1}\triangle y_j}+k^2(i,j,k+\frac{1}{2})\right]\mathbf{E}_z(i,j,k+\frac{1}{2})\\
&-\left[\frac{1}{\triangle x_{i-1}\triangle x_{i+\frac{1}{2}}}\right]\mathbf{E}_z(i-1,j,k+\frac{1}{2})-\left[\frac{1}{\triangle x_i\triangle x_{i+\frac{1}{2}}}\right]\mathbf{E}_z(i+1,j,k+\frac{1}{2})\\
&-\left[\frac{1}{\triangle y_{j-1}\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_z(i,j-1,k+\frac{1}{2})-\left[\frac{1}{\triangle y_j\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_z(i,j+1,k+\frac{1}{2})\\
&+\left[\frac{1}{\triangle z_k\triangle x_{i+\frac{1}{2}}}\right]\mathbf{E}_x(i-\frac{1}{2},j,k)-\left[\frac{1}{\triangle z_k\triangle x_{i+\frac{1}{2}}}\right]\mathbf{E}_x(i+\frac{1}{2},j,k)\\
&-\left[\frac{1}{\triangle z_k\triangle x_{i+\frac{1}{2}}}\right]\mathbf{E}_x(i-\frac{1}{2},j,k+1)+\left[\frac{1}{\triangle z_k\triangle x_{i+\frac{1}{2}}}\right]\mathbf{E}_x(i+\frac{1}{2},j,k+1)\\
&+\left[\frac{1}{\triangle z_k\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_y(i,j-\frac{1}{2},k)-\left[\frac{1}{\triangle z_k\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_y(i,j+\frac{1}{2},k)\\
&-\left[\frac{1}{\triangle z_k\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_y(i,j-\frac{1}{2},k+1)+\left[\frac{1}{\triangle z_k\triangle y_{j+\frac{1}{2}}}\right]\mathbf{E}_y(i,j+\frac{1}{2},k+1) \qquad =0.
\end{aligned}
$$

Appendix B

Expression of L for MT

In this section first we will discuss **L** matrix for Impedance, apparent resistivity and phase and phase tensor.

L for Impedance

Let **Z** be a 2×2 impedance tensor with its elements defined as $Z_{xx},\ Z_{xy},\ Z_{yx}$ and Z_{yy}. For a single frequency, the data **d** can be written as,

$$\mathbf{d} = \begin{bmatrix} \mathbf{Z}_{xx} \\ \mathbf{Z}_{xy} \\ \mathbf{Z}_{yx} \\ \mathbf{Z}_{yy} \end{bmatrix} \tag{B-5}$$

where, each block ($i = 1,2\ ;\ j = 1,2$) is a complex column vector (of length equal to number of sites) with elements representing the corresponding components of impedance values at the observation sites.

The matrix **L** (complex) takes the form as,

$$\mathbf{L} = \begin{bmatrix} \frac{\partial \mathbf{Z}_{xx}}{\partial \mathbf{e}} \\ \frac{\partial \mathbf{Z}_{xy}}{\partial \mathbf{e}} \\ \frac{\partial \mathbf{Z}_{yx}}{\partial \mathbf{e}} \\ \frac{\partial \mathbf{Z}_{yy}}{\partial \mathbf{e}} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_x^a(\lambda_{ex} - \bar{\mathbf{Z}}_{xx}\lambda_{bx} - \bar{\mathbf{Z}}_{xy}\lambda_{by}) & \mathbf{H}_y^a(\lambda_{ex} - \bar{\mathbf{Z}}_{xx}\lambda_{bx} - \bar{\mathbf{Z}}_{xy}\lambda_{by}) \\ \mathbf{H}_x^b(\lambda_{ex} - \bar{\mathbf{Z}}_{xx}\lambda_{bx} - \bar{\mathbf{Z}}_{xy}\lambda_{by}) & \mathbf{H}_y^b(\lambda_{ex} - \bar{\mathbf{Z}}_{xx}\lambda_{bx} - \bar{\mathbf{Z}}_{xy}\lambda_{by}) \\ \mathbf{H}_x^a(\lambda_{ey} - \bar{\mathbf{Z}}_{yx}\lambda_{bx} - \bar{\mathbf{Z}}_{yy}\lambda_{by}) & \mathbf{H}_y^a(\lambda_{ey} - \bar{\mathbf{Z}}_{yx}\lambda_{bx} - \bar{\mathbf{Z}}_{yy}\lambda_{by}) \\ \mathbf{H}_x^b(\lambda_{ey} - \bar{\mathbf{Z}}_{yx}\lambda_{bx} - \bar{\mathbf{Z}}_{yy}\lambda_{by}) & \mathbf{H}_y^b(\lambda_{ey} - \bar{\mathbf{Z}}_{yx}\lambda_{bx} - \bar{\mathbf{Z}}_{yy}\lambda_{by}) \end{bmatrix} \tag{B-6}$$

with each block of size $nobs \times N_e$. In eq B-6, $\mathbf{H}_x^k$ and $\mathbf{H}_y^k$ ($k = a,b$) are diagonal matrices with elements as inverse of magnetic fields at local sites, $\bar{\mathbf{Z}}_{ik}$ ($i = x,y; j = x,y$) represents the diagonal matrices with elements as impedance values at local sites and $\lambda_{ex}, \lambda_{ey}$ and $\lambda_{bx}, \lambda_{by}$ are interpolation and transformation matrices respectively. Note that the transformation matrices $\lambda_{bx}, \lambda_{by}$ are real and frequency independent, hence care must be taken while performing any operation involving these matrices. For example $\bar{\mathbf{Z}}_{xx}\lambda_{bx}$ will become $\alpha\bar{\mathbf{Z}}_{xx}\lambda_{bx}$ where α is $(-i\omega\mu)^{-1}$.

L for VTFs

For VTFs only first 2 rows in eq B-6 exists with λ_{ex} replaced by λ_{bz} and impedance matrices with VTFs matrices by hence, the expression becomes,

$$\mathbf{L} = \begin{bmatrix} \frac{\partial \mathbf{T}_x}{\partial \mathbf{e}} \\ \frac{\partial \mathbf{T}_y}{\partial \mathbf{e}} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_x^a(\lambda_{bz} - \bar{\mathbf{T}}_x\lambda_{bx} - \bar{\mathbf{T}}_y\lambda_{by}) & \mathbf{H}_y^a(\lambda_{bz} - \bar{\mathbf{T}}_x\lambda_{bx} - \bar{\mathbf{T}}_y\lambda_{by}) \\ \mathbf{H}_x^b(\lambda_{bz} - \bar{\mathbf{T}}_x\lambda_{bx} - \bar{\mathbf{T}}_y\lambda_{by}) & \mathbf{H}_y^b(\lambda_{bz} - \bar{\mathbf{T}}_x\lambda_{bx} - \bar{\mathbf{T}}_y\lambda_{by}) \end{bmatrix}. \tag{B-7}$$

Expression of L for Apparent resistivity and Phase

Apparent resistivity and phase provide examples of observations that are intrinsically real. In terms of the impedance, the apparent resistivity is defined as,

$$\rho_a = (\omega\mu)^{-1}|Z|^2 = (\omega\mu)^{-1}\left[Z_r^2 + Z_i^2\right], \tag{B-8}$$

where Z_r and Z_i are real and imaginary parts of the impedance Z and ω is angular frequency. Applying the chain rule,

$$\frac{\partial \rho_a}{\partial \mathbf{m}} = \frac{\partial \rho_a}{\partial Z_r}\frac{\partial Z_r}{\partial \mathbf{m}} + \frac{\partial \rho_a}{\partial Z_i}\frac{\partial Z_i}{\partial \mathbf{m}} = \frac{2}{\omega\mu}\left[Z_r\frac{\partial Z_r}{\partial \mathbf{m}} + Z_i\frac{\partial Z_i}{\partial \mathbf{m}}\right] \tag{B-9}$$

$$= \frac{2}{\omega\mu}\left[Z_r\Re\frac{\partial Z}{\partial \mathbf{m}} + Z_i\Im\frac{\partial Z}{\partial \mathbf{m}}\right] = \Re\left[\frac{2Z^*}{\omega\mu}\frac{\partial Z}{\partial \mathbf{m}}\right] = \Re\left[\frac{2Z^*\mathbf{I}_Z^T}{\omega\mu}\frac{\partial Z}{\partial \mathbf{m}}\right]. \tag{B-10}$$

Thus, $\mathbf{I}_\rho = 2Z^*\mathbf{I}_Z^T/\omega\mu$ gives the (complex) row of $\mathbf{L}$ for an apparent resistivity, again with the convention that the real part of the product in equation is taken taken for the corresponding row of the real Jacobian. Similarly for the phase $\phi = \tan^{-1}(Z_r/Z_i)$, we find that the row of $\mathbf{L}$ takes the form $\mathbf{I}_\phi = iZ^*\mathbf{I}_Z^T/|Z|^2$.

Expression of L for Phase Tensor

Caldwell et al. [17] introduced the concept of a 'phase tensor' (PT) and demonstrated that regional phase information can be recovered directly from the observed impedance tensor, where in, both the near-surface inhomogeneity and the regional conductivity structures can be 3-D. Hence, elements of PT can be inverted directly. Following Caldwell et al. [17], PT

is defined as,

$$\Phi = \mathbf{X}^{-1}\mathbf{Y} = \frac{1}{det\,\mathbf{X}} \begin{pmatrix} X_{22}Y_{11} - X_{12}Y_{21} & X_{22}Y_{12} - X_{12}Y_{22} \\ X_{11}Y_{21} - X_{21}Y_{11} & X_{11}Y_{22} - X_{21}Y_{12} \end{pmatrix}, \tag{B-11}$$

where $det\,\mathbf{X} = X_{11}X_{22} - X_{21}X_{12}$ and X and Y are the real and imaginary part of the MT impedance $\mathbf{Z} = X + iY$. We invert the phase tensor in term of its 4 real-valued components Φ_{ij}. Following the chain rule the sensitivity of the phase tensor Φ with respect to the solution of the electric fields $\mathbf{e}$ is,

$$\frac{\partial \Phi}{\partial \mathbf{e}} = \frac{\partial \Phi}{\partial \mathbf{Z}}\frac{\partial \mathbf{Z}}{\partial \mathbf{e}} = \frac{\partial \Phi}{\partial \mathbf{X}}\frac{\partial \mathbf{X}}{\partial \mathbf{e}} + \frac{\partial \Phi}{\partial \mathbf{Y}}\frac{\partial \mathbf{Y}}{\partial \mathbf{e}}. \tag{B-12}$$

eq B-12 means that sensitivities for phase tensor components can be obtained by a linear combination of the impedance sensitivities weighted by the derivatives of the phase tensor elements with respect to the (real and imaginary parts of the) impedance tensor elements [77, 121].

$$\frac{\partial \Phi_{ij}}{\partial \mathbf{e}} = \frac{\partial \Phi_{ij}}{\partial \mathbf{Z}}\frac{\partial \mathbf{Z}}{\partial \mathbf{e}} = \begin{pmatrix} \frac{\partial \Phi_{ij}}{\partial X_{11}} \\ \frac{\partial \Phi_{ij}}{\partial X_{12}} \\ \vdots \\ \frac{\partial \Phi_{ij}}{\partial Y_{22}} \end{pmatrix} \cdot \begin{pmatrix} \frac{\partial X_{11}}{\partial \mathbf{e}} \\ \frac{\partial X_{12}}{\partial \mathbf{e}} \\ \vdots \\ \frac{\partial Y_{22}}{\partial \mathbf{e}} \end{pmatrix}. \tag{B-13}$$

The derivatives of Φ_{11} with respect to the eight components of X and Y are,

$$\Phi_{11} = \frac{X_{22}Y_{11} - X_{12}Y_{21}}{det\,\mathbf{X}} = \frac{X_{22}Y_{11} - X_{12}Y_{21}}{X_{11}X_{22} - X_{21}X_{12}}, \tag{B-14}$$

$$\frac{\partial \Phi_{11}}{\partial X_{11}} = \frac{1}{(det\,\mathbf{X})^2}\Big[X_{22}(X_{22}Y_{11} - X_{12}Y_{21})\Big] = \frac{1}{det\,\mathbf{X}}X_{22}\Phi_{11}, \tag{B-15}$$

$$\frac{\partial \Phi_{11}}{\partial X_{12}} = \frac{1}{(det\,\mathbf{X})^2}\Big[-Y_{21}det\,\mathbf{X} - X_{21}(X_{22}Y_{11} - X_{12}Y_{21})\Big] = -\frac{1}{det\,\mathbf{X}}(Y_{21} + X_{21}\Phi_{11}), \tag{B-16}$$

$$\frac{\partial \Phi_{11}}{\partial X_{21}} = \frac{1}{(det\,\mathbf{X})^2}\Big[-X_{12}(X_{22}Y_{11} - X_{12}Y_{21})\Big] = -\frac{1}{det\,\mathbf{X}}X_{12}\Phi_{11}, \tag{B-17}$$

$$\frac{\partial \Phi_{11}}{\partial X_{22}} = \frac{1}{(det\,\mathbf{X})^2}\Big[Y_{11}det\,\mathbf{X} + X_{11}(X_{22}Y_{11} - X_{12}Y_{21})\Big] = \frac{1}{det\,\mathbf{X}}(Y_{11} + X_{11}\Phi_{11}), \tag{B-18}$$

$$\frac{\partial \Phi_{11}}{\partial Y_{11}} = \frac{1}{(det\,\mathbf{X})^2}(X_{22}det\,\mathbf{X}) = \frac{1}{det\,\mathbf{X}}X_{22}, \tag{B-19}$$

$$\frac{\partial \Phi_{11}}{\partial Y_{12}} = 0, \tag{B-20}$$

$$\frac{\partial \Phi_{11}}{\partial Y_{21}} = \frac{1}{(det\,\mathbf{X})^2}(-X_{12} det\,\mathbf{X}) = \frac{1}{det\,\mathbf{X}} X_{12}, \tag{B-21}$$

$$\frac{\partial \Phi_{11}}{\partial Y_{22}} = 0. \tag{B-22}$$

Similarly, derivatives for the other elements of the phase tensor can be derived.

Appendix C

Finite difference approximation of Governing equation For DC

Governing equation of DCR modelling is given as,

$$\nabla \cdot [\sigma(x,y,z)\nabla v(x,y,z)] = I\delta(x-x_q)\delta(y-y_q)\delta(z-z_q). \tag{C-1}$$

This vector PDE can be decomposed into primary and secondary and the equation for secondary potentials is written as,

$$\nabla \cdot [\sigma(x,y,z)\nabla v_a(x,y,z)] = -\nabla \cdot [\sigma_a(x,y,z)\nabla v_n(x,y,z)], \tag{C-2}$$

where σ_a denotes conductivity, subscript a represents anomalous media and subscripts n normal respectively. Using the central difference formula on normal grid, eq C-2 for i,j,k node can be approximated as,

$$\begin{aligned}
&\left[\frac{\sigma(i,j-\frac{1}{2},k)}{\triangle y_{j-1}\triangle y_{j+\frac{1}{2}}}+\frac{\sigma(i,j+\frac{1}{2},k)}{\triangle y_j\triangle y_{j+\frac{1}{2}}}+\frac{\sigma(i-\frac{1}{2},j,k)}{\triangle x_{i-1}\triangle x_{i+\frac{1}{2}}}+\frac{\sigma(i+\frac{1}{2},j,k)}{\triangle x_i\triangle x_{i+\frac{1}{2}}}+\frac{\sigma(i,j,k-\frac{1}{2})}{\triangle z_i\triangle z_{k+\frac{1}{2}}}+\frac{\sigma(i,j,k-\frac{1}{2})}{\triangle z_k\triangle z_{k+\frac{1}{2}}}\right]v^s(i,j,k)\\
&-\left[\frac{\sigma(i,j-\frac{1}{2},k)}{\triangle y_{j-1}\triangle y_{j+\frac{1}{2}}}\right]v^s(i,j-1,k)-\left[\frac{\sigma(i,j+\frac{1}{2},k)}{\triangle y_j\triangle y_{j+\frac{1}{2}}}\right]v^s(i,j+1,k)\\
&-\left[\frac{\sigma(i-\frac{1}{2},j,k)}{\triangle x_{i-1}\triangle x_{i+\frac{1}{2}}}\right]v^s(i-1,j,k)-\left[\frac{\sigma(i+\frac{1}{2},j,k)}{\triangle x_i\triangle x_{i+\frac{1}{2}}}\right]v^s(i+1,j,k)\\
&+\left[\frac{\sigma(i,j,k-\frac{1}{2})}{\triangle z_i\triangle z_{k+\frac{1}{2}}}\right]v^s(i,j,k+1)-\left[\frac{\sigma(i,j,k-\frac{1}{2})}{\triangle z_k\triangle z_{k+\frac{1}{2}}}\right]v^s(i,j,k+1)=RHS,
\end{aligned} \tag{C-3}$$

where, RHS is computed using the central difference formula as given above.

BIBLIOGRAPHY

[1] Alumbaugh, D. L., Newman, G. A., Prevost, L., and Shadid, J. N. (1996). Three-dimensional Wide Band Electromagnetic Modeling on Massively Parallel Computers. *Radio Science*, 31:1–23.

[2] Amatyakul, P., Vachiratienchai, C., and Siripunvaraporn, W. (2017). WSJointInv2D-MT-DCR: an efficient joint two-dimensional magnetotelluric and direct current resistivity inversion. *Computers & Geosciences*, 102:100–108.

[3] Argote-Espino, D. L., López-García, P. A., and Tejero-Andrade, A. (2016). 3D-ERT geophysical prospecting for the investigation of two terraces of an archaeological site northeast of Tlaxcala state, Mexico. *Journal of Archaeological Science: Reports*, 8:406–415.

[4] Avdeev, D. (2005). Three-dimensional electromagnetic modeling and inversion from theory to application. *Geothermics*, 26:767–799.

[5] Avdeev, D. and Avdeeva, A. (2009). 3D Magnetotelluric inversion using a limited-memory quasi-Newton optimization. *Geophysics*, 74:F45–F57.

[6] Bahr, K. (1991). Geological noise in Magnetotelluric data: a classification of distortion types. *Physics of the Earth and Planetary Interiors*, 66:24–38.

[7] Bai, D., Unsworth, M. J., Meju, M. A., Ma, X., Teng, J., Kong, X., Sun, Y., Sun, J., Wang, L., Jiang, C., et al. (2010). Crustal deformation of the eastern Tibetan plateau revealed by magnetotelluric imaging. *Nature geoscience*, 3(5):358.

[8] Bastani, M. F., Hubert, J., Kalscheuer, T., Pedersen, L., Godio, A., and Bernard, J. (2012). 2D joint inversion of RMT and ERT data versus individual 3D inversion of full tensor RMT data: an example from Trecate site in Italy. *Geophysics*, 77:WB233–WB243.

[9] Berdichevsky, M. N. and Dmitriev, V. I. (2010). *Models and methods of magnetotellurics*. Springer Science & Business Media.

[10] Bhattacharya, B. and Shalivahan, S. (2016). *Geoelectric Methods: Theory and Application*. New Delhi, India: McGraw Hill Education (India) Pvt. Ltd.

[11] Bhattacharya, B. B. and Shalivahan (2002). The electric moho underneath Eastern Indian Craton. *Geophysical Research Letters*, 29(10):14–1.

[12] Bing, Z. and Greenhalgh, S. A. (2001). Finite element three dimensional direct current resistivity modelling: accuracy and efficiency considerations. *Geophysical Journal International*, 145(3):679–688.

[13] Boonchaisuk, S., Vachiratienchai, C., and Siripunvaraporn, W. (2008). Two-dimensional direct current (DC) resistivity inversion: data space Occam's approach. *Physics of the Earth and Planetary Interiors*, 168:204–211.

[14] Brasse, H., Schäfer, A., Díaz, D., Alvarado, G. E., Muñoz, A., and Mütschard, L. (2015). Deep-crustal magma reservoirs beneath the Nicaraguan volcanic arc, revealed by 2-D and semi 3-D inversion of magnetotelluric data. *Physics of the Earth and Planetary Interiors*, 248:55–62.

[15] Brasse, H. and Soyer, W. (2001). A magnetotelluric study in the Southern Chilean Andes. *Geophysical Research Letters*, 28(19):3757–3760.

[16] Cagniard, L. (1953). Basic theory of the magnetotelluric method of geophysical prospecting. *Geophysics*, 18.

[17] Caldwell, T. G., Bibby, H., and Brown, C. (2004). The magnetotelluric phase tensor. *Geophysical Journal International*, 158.

[18] Caldwell, W. B., Klemperer, S. L., Lawrence, J. F., Rai, S. S., and others (2013). Characterizing the Main Himalayan Thrust in the Garhwal Himalaya, India with receiver function CCP stacking. *Earth and Planetary Science Letters*, 367:15–27.

[19] Caldwell, W. B., Klemperer, S. L., Rai, S. S., and Lawrence, J. F. (2009). Partial melt in the upper-middle crust of the northwest Himalaya revealed by Rayleigh wave dispersion. *Tectonophysics*, 477(1):58–65.

[20] Candansayar, M. E. and Tezkan, B. (2008). Two-dimensional joint inversion of radiomagnetotelluric and direct current resistivity data. *Geophysical Prospecting*, 56(5):737–749.

[21] Chave, A. D. and Jones, A. G. (2012). *The magnetotelluric method: Theory and practice*. Cambridge University Press.

[22] Chave, A. D. and Smith, J. T. (1994). On electric and magnetic galvanic distortion tensor decompositions. *Journal of Geophysical Research*, 99:4669–4682.

[23] Coggon, J. H. (1971). Electromagnetic and electrical modeling by the finite element method. *Geophysics*, 36(1):132–155.

[24] Commer, M. and Newman, G. A. (2009). Three-dimensional controlled-source electromagnetic and Magnetotelluric joint inversion. *Geophysical Journal International*.

[25] Constable, C. S., Parker, R. L., and Constable, C. G. (1987). Occam's inversion: a practical algorithm for generating smooth models from electromagnetic sounding data. *Geophysics*, 52:289–300.

[26] Coscia, I., Greenhalgh, S. A., Linde, N., Doetsch, J., Marescot, L., Günther, T., Vogt, T., and Green, A. G. (2011). 3D crosshole ERT for aquifer characterization and monitoring of infiltrating river water. *Geophysics*, 76(2):G49–G59.

[27] Cumming, W. and Mackie, R. (2010). Resistivity imaging of geothermal resources using 1D, 2D and 3D MT inversion and TDEM static shift correction illustrated by a Glass Mountain case history. In *Proceedings world geothermal congress*, Bali, Indonesia.

[28] Degroot-Hedlin, C. and Constable, S. (1990). Occam's inversion to generate smooth, two-dimensional models from magnetotelluric data. *Geophysics*, 55(12):1613–1624.

[29] Dehiya, R., Singh, A., Gupta, P. K., and Israil, M. (2017). Optimization of computations for adjoint field and jacobian needed in 3D CSEM inversion. *Journal of Applied Geophysics*, 136:444–454.

[30] D'Errico, J. (2005). Surface fitting using gridfit. *MATLAB Central File Exchange, Retrieved May 18, 2014*.

[31] D'Errico, J. (2012). inhull. *MATLAB Central File Exchange, Retrieved October 26, 2014*.

[32] Devi, A., Israil, M., Mamoriya, P., Gupta, P. K., and Varshney, S. K. (2018). Oblique and transverse tectonics structures in Garhwal Himalaya Corridor inferred from 3D inversion of magnetotelluric profile data. *(manuscript under preperation)*.

[33] Dey, A. and Morrison, H. F. (1979). Resistivity modeling for arbitrarily shaped three-dimensional structures. *Geophysics*, 44(4):753–780.

[34] Diaferia, I., Loddo, M., Schiavone, D., and Siniscalchi, A. (2008). Shallow to intermediate resistivity features of the Colfiorito Fault System inferred by DC and MT survey. *Annals of Geophysics*, 51(2-3).

[35] Dieter, K., Paterson, N. R., and Grant, F. S. (1969). IP and resistivity type curves for three-dimensional bodies. *Geophysics*, 34(4):615–632.

[36] Egbert, G. D. (1994). A new stochastic process on the sphere: application to characterization of long-period global scale external sources. In *14th Workshop on Electromagnetic Induction in the Earth and Moon*, Brest, France.

[37] Egbert, G. D. (2006). Efficient Inversion of Multi-frequency and Multi-Source Electromagnetic Data. In *Final Project Report. College of Oceanic and Atmospheric Sciences,*, Oregon State University, College of Oceanic and Atmospheric Sciences, Oregon State University.

[38] Egbert, G. D. (2012). Hybrid conjugate gradient-occam algorithms for inversion of multifrequency and multitransmitter EM data. *Geophysical Journal International*, 190(1):255–266.

[39] Egbert, G. D. and Kelbert, A. (2012). Computational recipes for electromagnetic inverse problems. *Geophysical Journal International*, 188:251–267.

[40] Ellis, R. G. and Oldenburg, D. W. (1994). The pole-pole 3-D DC-resistivity inverse problem: a conjugate gradient approach. *Geophysical Journal International*, 119(1):187–194.

[41] Ernst, T., Brasse, H., Cerv, V., Hoffmann, N., Jankowski, J., Jozwiak, W., Kreutzmann, A., Neska, A., Palshin, N., Pedersen, L. B., et al. (2008). Electromagnetic images of the deep structure of the Trans-European Suture Zone beneath Polish Pomerania. *Geophysical Research Letters*, 35(15).

[42] Farquharson, C. G. and Miensopust, M. P. (2011). Three-dimensional finite-element modelling of magnetotelluric data with a divergence correction. *Journal of Applied Geophysics*, 75(4):699–710.

[43] Fomenko, E. Y. and Mogi, T. (2002). A new computation method for a staggered grid of 3D EM field conservative modeling. *Earth Planets Space*, 54:499–509.

[44] Freund, R. W. and Nachtigal, N. (1991). QMR: a quasi-minimal residual method for non-Hermitian linear systems. *Numerische Mathematik*, 60(1):315–339.

[45] Friedrichs, B. (2003). MAPROS: magnetotelluric data processing software. *Metronix GmbH; Braunschweig, Germany*.

[46] Godin, L. and Harris, L. B. (2014). Tracking basement cross-strike discontinuities in the indian crust beneath the Himalayan orogen using gravity data–relationship to upper crustal faults. *Geophysical Journal International*, 198(1):198–215.

[47] Gokarn, S. G., Rao, C. K., and Gupta, G. (2002). Crustal structure in the Siwalik Himalayas using magnetotelluric studies. *Earth, planets and space*, 54(1):19–13.

[48] Gomez-Trevino, E. and Edwards, R. (1983). Electromagnetic soundings in the sedimentary basin of southern Ontario-a case history. *Geophysics*, 48(3):311–330.

[49] Gribenko, A., Green, M., Cuma, M., and Zhdanov, M. S. (2010). Efficient 3D inversion of MT data using integral equations method and the receiver footprint approach: application to the large-scale inversion of the EarthScope MT data. In *Expanded Abstracts of the SEG meeting*, page 644–649, Denver, Colorado.

[50] Groom, R. W. and Bailey, R. (1989). Decomposition of magnetotelluric impedance tensors in the presence of local three-dimensional galvanic distortion. *Journal of Geophysical Research*, 94:1913–1925.

[51] Groom, R. W. and Bailey, R. C. (1991). Analytic investigations of the effects of near-surface three-dimensional glavanic scatterers on MT tensor decompositions. *Geophysics*, 56(4):496–518.

[52] Günther, T., Rücker, C., and Spitzer, K. (2006). Three-dimensional modelling and inversion of DC resistivity data incorporating topography-II. Inversion. *Geophysical Journal International*, 166(2):506–517.

[53] Haber, E. (2005). Quasi-Newton methods for large scale electromagnetic inverse problem. *Inverse Problem*, 21:305–317.

[54] Haber, E., Ascher, U. M., Aruliah, D. A., and Oldenburg, D. W. (2000). Fast Simulation of 3D Electromagnetic Problems Using Potentials. *Journal of Computational Physics.*, 163:150–171.

[55] Haber, E., Ascher, U. M., and Oldenburg, D. W. (2004). Inversion of 3D electromagnetic data in frequency and time domain using an inexact all-at-once approach. *Geophysics*, 69:1216–1228.

[56] Haber, E., Oldenburg, D. W., and Shekhtman, R. (2007). Inversion of time domain three-dimensional electromagnetic data. *Geophysical Journal International*, 171:550–564.

[57] Habibian, B. D., Brasse, H., Oskooi, B., Ernst, T., Sokolova, E., Varentsov, I., EMTESZ Working Group, et al. (2010). The conductivity structure across the Trans-European Suture Zone from magnetotelluric and magnetovariational data modeling. *Physics of the Earth and Planetary Interiors*, 183(3):377–386.

[58] Heise, W., Caldwell, T. G., Bibby, H. M., and Bannister, S. C. (2008). Three-dimensional modelling of magnetotelluric data from the Rotokawa geothermal field, Taupo Volcanic Zone, New Zealand. *Geophysical Research Letters*, 173:740–750.

[59] Heise, W., Caldwell, T. G., Bibby, H. M., and Bennie, S. L. (2010). Three-dimensional electrical resistivity image of magma beneath an active continental rift, Taupo Volcanic Zone, New Zealand. *Geophysical Research Letters*, 37 (10):art. no. L10301.

[60] Hohmann, G. W. (1975). Three dimensional induced polarization and EM modeling. *Geophysics*, 40:309–324.

[61] Israil, M., Mamoriya, P., Gupta, P. K., and Varshney, S. K. (2016). Transverse tectonics feature delineated by modelling of magnetotelluric data from Garhwal Himalaya corridor. *Current Science*, 111(5).

[62] Israil, M., Tyagi, D., Gupta, P. K., and Niwas, S. (2008). Investigations for imaging electrical structure of Garhwal Himalaya corridor, Uttarakhand, India. *Journal of Earth System Sciences*, 117:189–200.

[63] Jones, F. and Pascoe, L. (1972). The perturbation of alternating geomagnetic fields by three-dimensional conductivity inhomogeneities. *Geophysical Journal International*, 27(5):479–485.

[64] Jones, K. A., Ingham, M. R., and Bibby, H. M. (2008). The hydrothermal vent system of Mount Ruapehu, New Zealand– a high frequency MT survey of the summit plateau. *Journal of Volcanology and Geothermal Research*, 176:591–600.

[65] Julia, J., Ammon, C., Herrmann, R., and Correig, A. M. (2000). Joint inversion of receiver function and surface wave dispersion observations. *Geophysical Journal International*, 143(1):99–112.

[66] Kalscheuer, T., Bastani, M., Donohue, S., Persson, L., Pfaffhuber, A. A., Reiser, F., and Ren, Z. (2013). Delineation of a quick clay zone at Smørgrav, Norway, with electromagnetic methods under geotechnical constraints. *Journal of Applied Geophysics*, 92:121–136.

[67] Kalscheuer, T., De los Ángeles García Juanatey, M., Meqbel, N., and Pedersen, L. B. (2010). Non-linear model error and resolution properties from two-dimensional single and joint inversions of direct current resistivity and radiomagnetotelluric data. *Geophysical Journal International*, 182(3):1174–1188.

[68] Kaufman, A. A. and Keller, G. V. (1981). *The Magnetotelluric Sounding Method*. Elsevier.

[69] Kaufman, A. A. and Keller, G. V. (1983). *Frequency and transient soundings*. Springer.

[70] Kelbert, A., Egbert, G. D., and Schultz, A. (2008). Non-linear conjugate gradient inversion for global EM induction: resolution studies. *Geophysical Journal International*, 173:365–381.

[71] Kelbert, A., Meqbel, N., Egbert, G. D., and Tandon, K. (2014). ModEM: a modular system for inversion of electromagnetic geophysical data. *Computers and Geosciences*, 66:40–53.

[72] Kesh, I. (2015). inputsdlg: Enhanced input dialog box. *MATLAB Central File Exchange, Retrieved November 28, 2015*.

[73] Khattri, K. M. and Tyagi, A. K. (1983a). Seismicity patterns in the Himalayan plate boundary and identification of the areas of high seismic potential. *Tectonophysics*, 96(3-4):281–297.

[74] Khattri, K. M. and Tyagi, A. K. (1983b). The transverse tectonic features in the Himalaya. *Tectonophysics*, 96(1-2):19–29.

[75] Komori, S., Kagiyama, T., Takakura, S., Ohsawa, S., Mimura, M., and Mogi, T. (2013). Effect of the hydrothermal alteration on the surface conductivity of rock matrix: comparative study between relatively-high and low temperature hydrothermal systems. *Journal of Volcanology and Geothermal Research*, 264:164–171.

[76] Korja, T., Smirnov, M., Pedersen, L., and Gharibi, M. (2008). Structure of the Central Scandinavian Caledonides and the underlying Precambrian basement, new constraints from magnetotellurics. *Geophysical Journal International*, 175(1):55–69.

[77] Kristina, T., Oliver, R., and Egbert, G. D. (2015). 3-D joint inversion of the magnetotelluric phase tensor and vertical magnetic transfer functions. *Geophysical Journal International*, 203:1128–1148.

[78] Kumar, G. P. and Manglik, A. (2011). Effect of Himalayan topography on two-dimensional interpretation of magnetotelluric data. *Current Science*, 100(3).

[79] Le Fort, P. (1975). Himalayas: the collided range. present knowledge of the continental arc. *American Journal of Science*, 275(1):1–44.

[80] Ledo, J. (2006). 2-D versus 3-D magnetotelluric data interpretation. *Surveys in Geophysics*, 27:111–148.

[81] Lee, S. K., Kim, H. J., Song, Y., and Lee, C. K. (2009). MT2DInvMatlab—A program in MATLAB and FORTRAN for two-dimensional magnetotelluric inversion. *Computers & Geosciences*, 35(8):1722–1734.

[82] Lee, T. (1975). An integral equation and its solution for some two-and three-dimensional problems in resistivity and induced polarization. *Geophysical Journal International*, 42(1):81–95.

[83] Lemonnier, C., Marquis, G., Perrier, F., Avouac, J. P., Chitrakar, G., Kafle, B., Sapkota, S., Gautam, U., Tiwari, D., and Bano, M. (199). Electrical structure of Himalaya of Central Nepal: high conductivity around mid-crustal ramp along the MHT. *Geophysical Research Letters*, 26(21):3261–3264.

[84] Li, Y. and Oldenburg, D. W. (1992). Approximate inverse mappings in DC resistivity problems. *Geophysical Journal International*, 109(2):343–362.

[85] Lilley, F. E. M. and Weaver, J. T. (2010). Phases greater than 90^o in MT data: analysis using dimensionality tools. *Journal of Applied Geophysics*, 70:9–16.

[86] Linde, N. and Pedersen, L. B. (2004). Characterization of a fractured granite using radio magnetotelluric (RMT) data. *Geophysics*, 69(5):1155–1165.

[87] Loke, M. H. and Barker, R. D. (1996). Rapid least-squares inversion of apparent resistivity pseudosections by a quasi-Newton method. *Geophysical prospecting*, 44(1):131–152.

[88] Lowry, T., Allen, M. B., and Shive, P. N. (1989). Singularity removal: A refinement of resistivity modeling techniques. *Geophysics*, 54(6):766–774.

[89] Mackie, R. L. and Madden, T. R. (1993). Three-dimensional magnetotelluric inversion using conjugate gradients. *Geophysical Journal International*, 115:215–229.

[90] Mackie, R. L., Madden, T. R., and Wannamaker, P. E. (1993). Three-dimensional magnetotelluric imodeling using difference equations: Theory and comparison to integral equation solutions. *Geophysics*, 58:215–226.

[91] Mackie, R. L., Smith, J. T., and Madden, T. R. (1994). Three-dimensional electromagnetic modeling using finite difference equations: The magnetotelluric example. *Radio Science*, 29:923–935.

[92] Mackie, R. L. and Watts, M. D. (2007). Joint 3D inversion of marine CSEM and MT data. In *Expanded abstract of SEG meeting*, San Antonio.

[93] Mahesh, P., Rai, S., Sivaram, K., Paul, A., Gupta, S., Sarma, R., and Gaur, V. (2013). One-dimensional reference velocity model and precise locations of earthquake

hypocenters in the Kumaon-Garhwal Himalaya. *Bulletin of the Seismological Society of America*, 103(1):328–339.

[94] Manglik, A. and Verma, S. K. (1998). Delineation of sediments below flood basalts by joint inversion of seismic and magnetotelluric data. *Geophysical Research Letters*, 25(21):4015–4018.

[95] Manglik, A., Verma, S. K., Sain, K., Arayana, T., and Rao, V. V. (2011). Joint Inversion of Seismic and MT Data–An Example from Southern Granulite Terrain, India. In *The Earth's Magnetic Interior*, pages 83–90.

[96] Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11:431–441.

[97] Maurer, H., Holliger, K., and Boerner, D. E. (1998). Stochastic regularization: Smoothness or similarity? *Geophysical Research Letters*, 25(15):2889–2892.

[98] McNeice, G. and Jones, A. G. (2001). Multisite, multifrequency tensor decomposition of magnetotelluric data. *Geophysics*, 66:158–173.

[99] Meju, M. A. (2002). Geoelectromagnetic exploration for natural resources: models, case studies and challenges. *Surveys in Geophysics*, 23:133–205.

[100] Menke, W. (2012). *Geophysical data analysis: Discrete inverse theory*, volume 45. Academic press.

[101] Meqbel, N. (2009). *The Electrical Conductivity Structure of the Dead Sea Basin Derived from 2D and 3D Inversion of Magnetotelluric Data*. PhD thesis, Free University of Berlin, Berlin, Germany.

[102] Miensopust, M. P., Queralt, P., Jones, A. G., and the 3D MT-modellers (2013). Magnetotelluric 3-D inversion-a review of two successful workshops on forward and inversion code testing and comparison. *Geophysical Journal International*, 193:1216–1238.

[103] Miglani, R., Shahrukh, M., Israil, M., Gupta, P. K., Varshney, S. K., and Sokolova, E. (2014). Geoelectric structure estimated from magnetotelluric data from the Uttarakhand Himalaya, India. *Journal of Earth System Science*, 123(8):1907–1918.

[104] Mitsuhata, Y. and Uchida, T. (2004). 3D magnetotelluric modeling using the T-ω finite-element method. *Geophysics*, 69(1):108–119.

[105] Molnar, P. (1990). A review of the seismicity and the rate of active underthrusting and deformation at the Himalaya. *Journal of Himalayan Geology*, 1(2):131–154.

[106] Moorkamp, M., Heincke, B., Jegen, M., Roberts, A. W., and Hobbs, R. W. (2011). A framework for 3-D joint inversion of MT, gravity and seismic refraction data. *Geophysical Journal International*, 184(1):477–493.

[107] Nabighian, M. N., editor (1988). *Electromagnetic methods in applied geophysics: Theory*, volume 1. SEG Books.

[108] Nam, M. J., Kim, H. J., Song, Y., Lee, T. J., Son, J.-S., and Suh, J. H. (2007). 3D magnetotelluric modelling including surface topography. *Geophysical Prospecting*, 55(2):277–287.

[109] Newman, G. A. and Alumbaugh, D. L. (1997). Three-dimensional massively parallel electromagnetic inversion-I Theory. *Geophysical Journal International*, 128:345–354.

[110] Newman, G. A. and Alumbaugh, D. L. (2000). Three-dimensional magnetotelluric inversion using non-linear conjugate gradients. *Geophysical Journal International*, 140:410–424.

[111] Newman, G. A. and Boggs, P. T. (2004). Solution accelerators for large-scale three-dimensional electromagnetic inverse problems. *Inverse Problem*, 20:S151–S170.

[112] Newman, G. A., Gasperikova, E., Hoversten, G. M., and Wannamaker, P. E. (2008). Three-dimensional magnetotelluric characterization of the Coso geothermal field. *Geothermics*, 37(4):369–399.

[113] Newman, G. A., Recher, S., Tezkan, B., and Neubauer, F. M. (2003). Case History 3D inversion of a scalar radio Magnetotelluric field data set. *Geophysics*, 68:791–802.

[114] Nocedal, J. and Wright, S. J. (2000). *Numerical Optimization*. Springe, New York, USA.

[115] Ogawa, Y. (2002). On two-dimensional modeling of magnetotelluric field data. *Surveys in Geophysics*, 23:251–272.

[116] Okabe, M. (1981). Boundary element method for the arbitrary inhomogeneities problem in electrical prospecting. *Geophysical Prospecting*, 29(1):39–59.

[117] Oskooi, B., Pedersen, L. B., Smirnov, M., Árnason, K., Eysteinsson, H., Manzella, A., Group, D. W., et al. (2005). The deep geothermal structure of the Mid-Atlantic Ridge deduced from MT data in SW Iceland. *Physics of the Earth and Planetary Interiors*, 150(1):183–195.

[118] Park, S. K. and Van, G. P. (1991). Inversion of pole-pole data for 3-D resistivity structure beneath arrays of electrodes. *Geophysics*, 56(7):951–960.

[119] Parker, R. L. (1983). The magnetotelluric inverse problem. *Geophysical surveys*, 6(1-2):5–25.

[120] Parker, R. L. (1994). *Geophysical inverse theory*. Princeton University Press, Princeton.

[121] Patro, K. P., Uyeshima, M., and Siripunvaraporn, W. (2013). Three-dimensional inversion of magnetotelluric phase tensor data. *Geophysical Journal International*, 192(2):58–66.

[122] Patro, P. K., Brasse, H., Sarma, S. V. S., and Harinarayana, T. (2005). Electrical structure of the crust below the Deccan Flood Basalts (India), inferred from magnetotelluric soundings. *Geophysical Journal International*, 163(3):931–943.

[123] Patro, P. K. and Egbert, G. D. (2008). Regional conductivity structure of Cascadia: preliminary results from 3D inversion of USA array transportable array Magnetotelluric data. *Geophysical Research Letters*, 35(20).

[124] Patro, P. K. and Harinarayana, T. (2009). Deep geoelectric structure of the Sikkim Himalayas (NE India) using magnetotelluric studies. *Physics of the Earth and Planetary Interiors*, 173(1):171–176.

[125] Petrick Jr, W. R., Sill, W. R., and Wards, S. (1981). Three-dimensional resistivity inversion using alpha centers. *Geophysics*, 46(8):1148–1162.

[126] Pidlisecky, A., Haber, E., and Knight, R. (2007). RESINVM3D: A 3D resistivity inversion package. *Geophysics*, 72(2):H1–H10.

[127] Pidlisecky, A., Moran, T., Hansen, B., and Knight, R. (2016). Electrical resistivity imaging of seawater intrusion into the monterey bay aquifer system. *Groundwater*, 54(2):255–261.

[128] Pridmore, D. F., Hohmann, G. W., Ward, S. H., and Sill, W. R. (1981). An investigation of finite-element modeling for electrical and electromagnetic data in three dimensions. *Geophysics*, 46(7):1009–1024.

[129] Raiche, A. P., Jupp, D., Rutter, H., and Vozoff, K. (1985). The joint use of coincident loop transient electromagnetic and schlumberger sounding to resolve layered structures. *Geophysics*, 50(10):1618–1627.

[130] Ramirez, A., Daily, W., Binley, A., LaBrecque, D., and Roelant, D. (1996). Detection of leaks in underground storage tanks using electrical resistance methods. *Journal of Environmental and Engineering Geophysics*, 1(3):189–203.

[131] Rawat, G., Arora, B. R., and Gupta, P. K. (2014). Electrical resistivity cross-section across the Garhwal Himalaya: Proxy to fluid-seismicity linkage. *Tectonophysics*, 637:68–79.

[132] Reddy, I. K., Phillips, R. J., and Rankin, D. (1977). Three-dimensional modelling in magnetotelluric and magnetic variational sounding. *Geophysical Journal*, 51:313–325.

[133] Rijo, L. (1984). Inversion of three-dimensional resistivity and induced-polarization data. In *SEG Technical Program Expanded Abstracts 1984*, pages 113–117. Society of Exploration Geophysicists.

[134] Rodi, W. L. (1976). A technique for improving the accuracy of finite element solutions for magnetotelluric data. *Geophysical Journal International*, 44:483–506.

[135] Rodi, W. L. and Mackie, R. L. (2001). Nonlinear conjugate gradients algorithm for 2-D magnetotelluric inversion. *Geophysics*, 66:174–187.

[136] Saad, Y. (2003). *Iterative methods for sparse linear systems*. Siam.

[137] Saad, Y. and Schultz, M. H. (1986). Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869.

[138] Sasaki, Y. (1989). Two-dimensional joint inversion of magnetotelluric and dipole-dipole resistivity data. *Geophysics*, 54:254–262.

[139] Sasaki, Y. (1994). 3-D resistivity inversion using the finite-element method. *Geophysics*, 59(12):1839–1848.

[140] Sasaki, Y. (1999). Three-dimensional frequency-domain modeling using the finite-difference method. *Butsuri-Tansa*, 52:421–431.

[141] Sasaki, Y. (2001). Full 3D inversion of electromagnetic data on PC. *Journal of Applied Geophysics*, 46:45–54.

[142] Sasaki, Y. (2004). Three-dimensional inversion of static-shifted magnetotelluric data. *Earth Planets Space*, 56:239–248.

[143] Seher, T. and Tezkan, B. (2007). Radiomagnetotelluric and direct current resistivity measurements for the characterization of conducting soils. *Journal of Applied Geophysics*, 63(1):35–45.

[144] Shalivahan and Bhattacharya, B. B. (2002). Implications of novel results about Moho from magnetotelluric studies. *Current Science*, 83(10):1259–1264.

[145] Shalivahan and Bhattacharya, B. B. (2005). Electrical anisotropy of asthenosphere in a region of window to mantle underneath Eastern Indian Craton. *Physics of the Earth and Planetary Interiors*, 152(1):43–61.

[146] Shan, C., Bastani, M., Malehmir, A., Persson, L., and Engdahl, M. (2014). Integrated 2D modeling and interpretation of geophysical and geotechnical data to delineate quick clays at a landslide site in southwest Sweden. *Geophysics*.

[147] Sharma, S., Prácser, E., and Roy, K. (2005). Joint inversion of seismic refraction and magnetotelluric data for resolving deeper subsurface structure. *Acta Geodaetica et Geophysica Hungarica*, 40(2):241–258.

[148] Shima, H. (1992). 2-D and 3-D resistivity image reconstruction using crosshole data. *Geophysics*, 57(10):1270–1281.

[149] Simpson, F. and Bahr, K. (2005). *Practical magnetotellurics*. Cambridge University Press.

[150] Singh, A., Dehiya, R., Gupta, P. K., and Israil, M. (2014). Development of block inversion algorithm and its comparison with cell inversion scheme. In *22th Workshop on Electromagnetic Induction, Weimar, Germany*.

[151] Singh, A., Dehiya, R., Gupta, P. K., and Israil, M. (2017). A MATLAB based 3D modeling and inversion code for MT data. *Computers & Geosciences*, 104:1–11.

[152] Sinharay, R. K. and Bhattacharya, B. B. (2001). An analysis of magnetotelluric (MT) data over geothermal region of Bakreshwar, West Bengal. *Journal of Geophysics*, 22(1):31–39.

[153] Sinharay, R. K., Srivastava, S., and Bhattacharya, B. B. (2010). Audiomagnetotelluric studies to trace the hydrological system of thermal fluid flow of Bakreswar Hot Spring, Eastern India: A case history. *Geophysics*, 75(5):B187–B195.

[154] Siripunvaraporn, W. (2012). Three-Dimensional Magnetotelluric Inversion: An Introductory Guide for Developers and Users. *Surveys in Geophysics*, 33:5–27.

[155] Siripunvaraporn, W. and Egbert, G. (2000). An effcient data-subspace inversion method for 2D magnetotelluric data. *Geophysics*, 65(3):791–803.

[156] Siripunvaraporn, W. and Egbert, G. (2007). Data space conjugate gradient inversion for 2-D magnetotelluric data. *Geophysical Journal International*, 170:986–994.

[157] Siripunvaraporn, W. and Egbert, G. (2009). WSINV3DMT: vertical magnetic field transfer function inversion and parallel implementation. *Physics of the Earth and Planetary Interiors*, 173:317–329.

[158] Siripunvaraporn, W., Egbert, G., and Lenbury, Y. (2002). Numerical accuracy of magnetotelluric modeling: a comparison of finite difference approximations. *Earth Planets Space*, 54(6):721–725.

[159] Siripunvaraporn, W., Egbert, G., Lenbury, Y., and Uyeshima, M. (2005a). Three-dimensional magnetotelluric inversion: data-space method. *Physics of the Earth and Planetary Interiors*, 150:3–14.

[160] Siripunvaraporn, W., Egbert, G., and Uyeshima, M. (2005b). Interpretation of two-dimensional magnetotelluric profile data with three-dimensional inversion: synthetic examples. *Geophysical Journal International*, 160:804–814.

[161] Siripunvaraporn, W., Uyeshima, M., and Egbert, G. (2004). Three-dimensional inversion for Network-Magnetotelluric data. *Earth Planets Space*, 56:893–902.

[162] Smirnov, M. (2003). Magnetotelluric data processing with a robust statistical procedure having a high breakdown point. *Geophysical Journal International*, 152(1):1–7.

[163] Smirnov, M. and Pedersen, L. B. (2009). Magnetotelluric measurements across the Sorgenfrei-Tornquist Zone in southern Sweden and Denmark. *Geophysical Journal International*, 176(2):443–456.

[164] Smith, J. T. (1996a). Conservative modeling of 3-D electromagnetic fields, Part I: properties and error analysis. *Geophysics*, 61:1308–1318.

[165] Smith, J. T. (1996b). Conservative modeling of 3-D electromagnetic fields, Part II: biconjugate gradient solution and an accelerator. *Geophysics*, 61:1319–1324.

[166] Spitzer, K. (1995). A 3D finite-difference algorithm for DC resistivity modeling using conjugate gradient methods. *Geophysical Journal International*, 123:903–914.

[167] Spitzer, K. (1998). The three-dimensional DC sensitivity for surface and subsurface sources. *Geophysical Journal International*, 134(3):736–746.

[168] Spitzer, K., Wurmstich, B., Oristaglio, M., and Spies, B. (1999). Speed and accuracy in 3D resistivity modeling. *Three-Dimensional Electromagnetics*, 7:161–176.

[169] Streich, R. (2009). 3D finite-difference frequency-domain modeling of controlled-source electromagnetic data: Direct solution and optimization for high accuracy. *Geophysics*, 74(5):F95–F105.

[170] Swift, C. M. (1967). *A magnetotelluric investigation of electrical conductivity anomaly in the southwestern United States*. PhD thesis, MIT, Cambridge, M.A.

[171] Szalai, S., Szokoli, K., Metwaly, M., Gribovszki, Z., and Prácser, E. (2017). Prediction of the location of future rupture surfaces of a slowly moving loess landslide by electrical resistivity tomography. *Geophysical Prospecting*, 65(2):596–616.

[172] Szokoli, K., Szarka, L., Metwaly, M., Kalmár, J., Prácser, E., and Szalai, S. (2017). Characterisation of a landslide by its fracture system using electric resistivity tomography and pressure probe methods. *Acta Geodaetica et Geophysica*.

[173] Takasugi, S., Tanaka, K., Kawakami, N., and Muramatsu, S. (1992). High spatial resolution of the resistivity structure revealed by a dense network MT measurement. *Journal of geomagnetism and geoelectricity*, 44(4):289–308.

[174] Tarantola, A. (2005). *Inverse problem theory and methods for model parameter estimation*. Siam.

[175] Tezkan, B., Georgescu, P., and Fauzi, U. (2005). A radiomagnetotelluric survey on an oil-contaminated area near the Brazi Refinery, Romania. *Geophysical Prospecting*, 53(3):311–323.

[176] Tezkan, B., Goldman, M., Greinwald, S., Hördt, A., Müller, I., Neubauer, F. M., and Zacher, G. (1996). A joint application of radiomagnetotellurics and transient electromagnetics to the investigation of a waste deposit in Cologne (Germany). *Journal of Applied Geophysics*, 34(3):199–212.

[177] Tezkan, B., Hördt, A., and Gobashy, M. (2000). Two dimensional inversion of radiomagnetotelluric data: selected case histories for waste site exploration. *Journal of Applied Geophysics*, 44:237–256.

[178] Tezkan, B. and Saraev, A. (2008). A new broadband radiomagnetotelluric instrument: applications to near surface investigations. *Near Surface Geophysics*, 6(4):245–252.

[179] Tikhonov, A. N. (1950). On determining electric characteristics of teh deep layers of the earth's crust. *Dokl Acad Nauk SSSR*, 73:295–297.

[180] Ting, S. C. and Hohmann, G. W. (1981). Integral equation modeling of three-dimensional magnetotelluric response. *Geophysics,*, 46:182–197.

[181] Tsokas, G. N., Giannopoulos, A., Tsourlos, P., Vargemezis, G., Tealby, J., Sarris, A., Papazachos, C. B., and Savopoulou, T. (1994). A large scale geophysical survey in the archaeological site of Europos (northern Greece). *Journal of Applied Geophysics*, 32(1):85–98.

[182] Unsworth, M., Bedrosian, P., Eisel, M., Egbert, G., and Siripunvaraporn, W. (2000). Along strike variations in the electrical structure of the San Andreas Fault at Parkfield, California. *Geophysical Research Letters*, 27:3021–3024.

[183] Unsworth, M. J., Jones, A. G., Wei, W., Marquis, G., Gokarn, S. G., Spratt, J. E., Bedrosian, P., Booker, J., Leshou, C., Clarke, G., et al. (2005). Crustal rheology of the Himalaya and Southern Tibet inferred from magnetotelluric data. *Nature*, 438(7064):78.

[184] Valdiya, K. S. (1976). Himalayan transverse faults and folds and their parallelism with subsurface structures of north Indian plains. *Tectonophysics*, 32(3-4):353–386.

[185] Valdiya, K. S. (2003). Reactivation of Himalayan frontal fault: implications. *Current Science*, pages 1031–1040.

[186] Van der Vorst, H. A. (2003). *Iterative Krylov methods for large linear systems*, volume 13. Cambridge University Press.

[187] Vozoff, K. (1972). The magnetotelluric method in the exploration of sedimentary basins. *Computers & Geosciences*, 37:98–141.

[188] Vozoff, K. and Jupp, D. (1975). Joint inversion of geophysical data. *Geophysical Journal International*, 42(3):977–991.

[189] Wait, J. R. (1962). Electromagnetic waves in stratified media, vol. 3 of international series of monographs on electromagnetic waves.

[190] Wannamaker, P. E. (1991). Advances in three dimensional magnetotelluric modeling using integral equations. *Geophysics*, 56:1716–1728.

[191] Ward, S. H. and Hohmann, G. W. (1987). Electromagnetic theory for geophysical applications. M. N. Nabighian (ed). *Electromagnetic methods in applied geophysics. 2, Application, Parts A and B, SEG, Tulsa*, pages 131–311.

[192] Weaver, J. T. (1994). *Mathematical methods for geo-electromagnetic induction.* Research Studies Press Ltd. John Wiley & Sons.

[193] Weaver, J. T., Agarwal, A. K., and Lilley, F. E. M. (2000). Characterization of the magnetotelluric impedance tensor. *Geophysical Journal International*, 129:133–142.

[194] Wei, W., Unsworth, M., Jones, A., Booker, J., Tan, H., Nelson, D., Chen, L., Li, S., Solon, K., Bedrosian, P., and others (2001). Detection of widespread fluids in the Tibetan crust by magnetotelluric studies. *Science*, 292(5517):716–719.

[195] Wittke, J. and Tezkan, B. (2014). Meshfree magnetotelluric modelling. *Geophysical Journal International*, 198(2):1255–1268.

[196] Xiao, Q., Cai, X., Xu, X., Liang, G., and Zhang, B. (2010). Application of the 3D magnetotelluric inversion code in a geologically complex area. *Geophysical Prospecting*, 58(6):1177–1192.

[197] Xiong, Z. and Tripp, A. C. (1995). Electromagnetic scattering of large structures in layered earths using integral equations. *Radio Science*, 30:921–921.

[198] Xu, S.-z., Gao, Z., and Zhao, S.-k. (1988). An integral formulation for three-dimensional terrain modeling for resistivity surveys. *Geophysics*, 53(4):546–552.

[199] Yamaya, Y., Mogi, T., Hashimoto, T., and Ichihara, H. (2009). Hydrothermal system beneath the crater of Tarumai volcano, Japan: 3-D resistivity structure revealed using audio-magnetotellurics and induction vector. *Journal of Volcanology and Geothermal Research*, 187(3):193–202.

[200] Yan, L. I., Xiang-Yun, H. U., Wen-Cai, Y. A. N. G., Wen-Bo, W. E. I., Hui, F. A. N. G., Bo, H. A. N., and Rong-Hua, P. E. N. G. (2013). A Study on Parallel Computation for 3D Magneto-Telluric Modeling Using the Staggered-Grid Finite Difference Method. *Chinese Journal of Geophysics*, 56(3):287–295.

[201] Yee, K. S. (1966). Numerical Solution of Initial Boundary Value Problems Involving Maxwell's Equations in Isotropic Media. *IEEE Transactions on Antennas and Propagation*, AP-14:302–307.

[202] Yogeshwar, P., Tezkan, B., Israil, M., and Candansayar, M. (2012). Groundwater contamination in the Roorkee area, India: 2D joint inversion of radiomagnetotelluric and direct current resistivity data. *Journal of Applied Geophysics*, 76:127–135.

[203] Yuguo, L. and Spitzer, K. (2002). Three-dimensional DC resistivity forward modelling using finite elements in comparison with finite-difference solutions. *Geophysical Journal International*, 151:924–934.

[204] Zhang, J., Mackie, R. L., and Madden, T. R. (1995). 3D resistivity forward modeling and inversion using conjugate gradients,. *Geophysics*, 60(5):1313–1325.

[205] Zhao, S. and Yedlin, M. J. (1996). Some refinements on the finite-difference method for 3-D dc resistivity modeling. *Geophysics*, 61(5):1301–1307.

[206] Zhdanov, M. and Hursan, G. (2000). 3D electromagnetic inversion based on quasianalytical approximation. *Inverse Problem*, 16:1297–1322.

[207] Zhdanov, M. S. (2009). Geophysical electromagnetic theory and methods. *Elsevier*.

[208] Zhdanov, M. S. and Keller, G. V. (1994). *The geoelectrical methods in geophysical exploration*, volume 31. Elsevier Science Limited.

[209] Zhdanov, M. S., Lee, S. K., and Yoshioka, K. (2006). Integral equation method for 3D modeling of electromagnetic fields in complex structures with inhomogeneous background conductivity. *Geophysics*, 71(6):G333–G345.

[210] Zhdanov, M. S., Varentsov, I. M., Weaver, J. T., Golubev, N. G., and Krylov, V. A. (1997). Methods for modelling electromagnetic fields results from COMMEMI—the international project on the comparison of modelling methods for electromagnetic induction. *Journal of Applied Geophysics*, 37:133–271.