On Optimization Techniques for Solving Nonlinear Inverse Problems

Eldad Haber[*] Uri M. Ascher[†] Doug Oldenburg[‡]

March 1, 2000

Abstract

This paper considers optimization techniques for the solution of nonlinear inverse problems where the forward problems, like those encountered in electromagnetics, are modelled by differential equations. Typically, such problems are solved utilising a Gauss-Newton method in which the forward model constraints are implicitly incorporated. Variants of Newton's method which use second derivative information are rarely employed because their perceived disadvantage in computational cost per step offsets their potential benefits of faster convergence. In this paper we show that by formulating the inversion as a constrained or unconstrained optimization problem, and by employing sparse matrix techniques, we can carry out variants of sequential quadratic programming and the full Newton iteration with only a modest additional cost. By working with the differential equation explicitly we are able to relate the constrained and the unconstrained formulations and discuss advantages of each. To make the comparisons meaningful we adopt the same global optimization strategy for all inversions. As an illustration, we focus upon a 1D electromagnetic example simulating a magnetotelluric survey. This problem is sufficiently rich that it illuminates most of the computational complexities that are prevalent in multi-source inverse problems and we therefore describe its solution process in detail. The numerical results illustrate that variants of Newton's method which utilize second derivative information can produce a solution in fewer iterations and, in some cases where the data contain significant noise, requiring fewer floating point operations than Gauss-Newton techniques. Although further research is

[*] Department of Computer Science and Institute of Applied Mathematics, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada. (haber@cs.ubc.ca). The work of this author was partially supported under NSERC Canada Grants OGP0004306 and a PIms Postdoctoral Fellowship.

[†] Department of Computer Science, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada. (ascher@cs.ubc.ca). The work of this author was partially supported under NSERC Canada Grant OGP0004306.

[‡] Department of Earth and Ocean Science, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada. (doug@geop.ubc.ca).

required, we believe that the variants proposed here will have a significant impact on developing practical solutions to large-scale 3D electromagnetic inverse problems.

Keywords: Constrained optimization, Gauss-Newton, Lagrange multipliers, Electromagnetic, Magnetotellurics, Newton's method, Sequential quadratic programming, Noise, Nonlinear inverse problems, Regularization.

1 Introduction

The inversion of electromagnetic data to recover 3D distributions of physical properties from observed electric and magnetic fields represents a major challenge in computational geosciences. Traditional strategies employing Gauss-Newton methods are computationally slow because they require the formulation and inversion of a large dense sensitivity matrix and also because many iterations are often required to achieve convergence. This has been the motivation for us to re-examine strategies for solving the EM inverse problem. Major research questions are focused upon whether taking full Newton steps can significantly reduce the number of iterations (and flops) required for convergence, whether or not it is necessary for the forward modelling constraint to be strictly enforced at each iteration, and whether an algorithm can be developed whereby the formation and storage of the sensitivity matrix can be obviated. In order to address these questions we need to work directly with the differential equations.

Although motivated by EM problems, our work is general and applicable to solving any nonlinear inverse problem whose goal is to recover the coefficient functions in a system of differential equations. Accordingly, we shall use a generic notation and begin with a general formulation of an inverse problem.

Typically, one seeks to recover a model m based on observations of a field u, where u is related to m by a forward model – a system of differential equations which we write in discretized form as

$$f(m,u) = 0. \tag{1}$$

Denoting the data vector by b and the location of the observations by Q, the problem is to find m such that (1) holds and

$$\|Qu - b\| \leq Tol, \tag{2}$$

where Tol depends on the noise level. For instance, by the discrepancy principle, if the data vector has length N_b and contains noise with known standard deviation σ then set

$$Tol = N_b \sigma. \tag{3}$$

However, since the data are noisy, and the inverse problem of recovering m from (2,1) is often ill-posed even without noise, there is no unique model which generates the data. Therefore, a process of regularization is used to recover stably a relatively smooth (or piecewise smooth) solution to a nearby problem which is unique, at least locally. A model often utilized in practice minimizes a least squares residual vector where a regularization term is added [22],

$$\begin{aligned} \min_{m,u} \quad & \frac{1}{2}\|Qu-b\|^2 + \frac{\beta}{2}\|W(m-m_{ref})\|^2 \\ \text{s.t.} \quad & f(m,u)=0, \end{aligned} \tag{4}$$

where W is typically a weighting matrix involving discretized derivatives which does not depend on m, m_{ref} is a reference model, and $\beta \geq 0$ is the regularization parameter.

The problem (4) is a nonlinear constrained optimization problem. Often in practice, though, the forward model can be written as

$$A(m)u = q \tag{5}$$

where A is a square, nonsingular matrix which is typically large and sparse. Thus, the forward model is linear in u and allows an explicit elimination, $u = A^{-1}q$. The constrained optimization problem

$$\begin{aligned} \min_{m,u} \quad & \frac{1}{2}\|Qu-b\|^2 + \frac{\beta}{2}\|W(m-m_{ref})\|^2 \\ \text{s.t.} \quad & A(m)u=q, \end{aligned} \tag{6}$$

can then be written as an unconstrained, nonlinear least squares problem,

$$\min_{m} \frac{1}{2}\|QA(m)^{-1}q-b\|^2 + \frac{\beta}{2}\|W(m-m_{ref})\|^2. \tag{7}$$

The optimization problems obtained in this way are typically very large – often too large for standard software to handle – but they have significant sparsity structure which can be exploited. The common approach in the literature hitherto has been to solve the unconstrained formulation (7) using the Gauss-Newton method. This method generates a sequence of iterates where for a given current iterate m, an update of the form $m \leftarrow m + \delta m$ is subsequently carried out, and the process is repeated to convergence. The correction direction δm is obtained by linearizing the expression under the norm in (7) and solving a linear least squares problem,

$$(J^TJ + \beta W^TW)\delta m = J^T(b - QA^{-1}q) - \beta W^TW(m - m_{ref}),$$

where $J = J(m) = \frac{\partial(QA^{-1}q)}{\partial m}$ is the sensitivity matrix, cf. [3, 17].

The Gauss-Newton method is a simplification of Newton's method for solving the system of nonlinear equations which form the necessary conditions for the unconstrained minimum,

$$J^T(QA^{-1}q - b) + \beta W^TW(m - m_{ref}) = 0, \tag{8}$$

in which second-derivative information is discarded [5, 18]. It has well-known advantages and disadvantages in general. However, its almost exclusive use in the geophysical data inversion literature stems from the fact that the calculation of the second derivatives involved in the Newton step has commonly been regarded as prohibitively expensive. As a consequence, serious evaluations comparing the relative merit of taking full Newton or Gauss-Newton steps have not been carried out in the geophysical EM literature.

In this paper we show that by formulating the inverse problem directly in the differential equation domain we may make these comparisons. Forming the full Newton step does not in fact cost much more than the Gauss-Newton approximation. Moreover, regarding the model and the predicted data as two *independent* quantities which are connected through the forward modeling, i.e. considering the constrained problem (4) or (6) (as, e.g., in [15]), affords a unified view of such methods, as well as additional generality, both in terms of the modeling (including (4)) and in terms of applying sparse linear algebra techniques for the execution of each iteration (cf. [7]).

In addition, we show that in order to obtain a product of the sensitivity matrix and a given vector we need not calculate the sensitivity matrix explicitly. This is very important for large scale problems, because J is typically a dense matrix which can be very large. Our computational expediency is realized because J, although full, can have an implicit sparse representation in the cases considered here.

Various Newton and Gauss-Newton variants are developed in Section 3, where we make explicit use of the special form of the constraints (5). To compare their overall efficacy, we adopt a single global inversion methodology based upon a continuation procedure in the regularization parameter, β. This is described in Section 4.

For the numerical experiments we consider the 1D magnetotelluric (MT) problem. We present this in detail so that it can be used as a template for working with other data sets that have multiple sources.

2 Details of the Problem

In this paper we consider exclusively the forward model (5), with $A = A(m)$ a nonsingular $N_u \times N_u$ matrix. This matrix represents a finite difference or finite element discretization of a linear differential system plus boundary conditions, and it depends on a vector m of length N_m containing the model parameters. For the magnetotelluric (MT) example described later, A is a large, sparse matrix which depends on the (discretized) log conductivity m; u represents the electric field through the earth on the grid; and q contains the boundary values.

We especially consider cases where the inverse problem arises from multi-experiments, for example the MT experiment with more than one frequency or a controlled source experiment with sources at different locations. In these cases the matrix A is composed of blocks

$$A = diag\{A_1, \dots , A_n\},$$

where the matrix $A_k(m)$ represents the k^{th} experiment. The vector u in this case is written as

$$u = [u_1^T, \dots, u_n^T]^T$$

where the sub-vector u_k is of length N_u^k and represents the field for the k^{th} experiment.

Note that the number N_m of model parameters is usually different from the number $N_u = \sum_{k=1}^{n} N_u^k$ of parameters used to discretize u. In fact, $N_m \ll N_u$ in most problems that arise from multi-experiments, since many fields are associated with a single model.

Returning to the data fitting problem, the predicted data are assumed to be the field u measured at locations which form a subset of the discretization grid. Hence, the matrix Q appearing in (6) and (7) is $N_b \times N_u$ and consists of selected unit rows. It projects the field u to the measurement locations. In most geophysical instances Q projects the field onto the surface; hence its typical dimensions satisfy $N_b \ll N_u$.

Given the forward model (5) described above, we formulate the *unconstrained* formulation (7) and the *constrained* formulation (6). In the constrained formulation u is not an explicit function of m and, unless enforced specifically, $Au \neq q$ during the minimization process. However, the constraint (5) holds upon convergence of the minimization process.

The constrained formulation has a number of advantages. As mentioned before, it is more general than the unconstrained approach. Furthermore, it gives additional degrees of freedom which may be of use both in developing sparse matrix techniques and if the constraints are highly nonlinear. This approach may produce a minimization problem that is closer to a linear one, since A is not inverted explicitly [18].

There are also disadvantages to the constrained formulation. The size of the problem increases and we need to calculate Lagrange multipliers (although we show some ways around that below). More importantly, the problem is to find a saddle point of the Lagrangian, as compared to finding a strict minimum in (7). The choice of merit function subsequently becomes more complex for the constrained formulation (see, e.g., [18] and (21) below).

In the next section we show how we can use a well established technique, Sequential Quadratic Programming (SQP), to solve the constrained optimization problem for a given parameter β. We then show that by small variations in the SQP framework it is possible to obtain the Newton or Gauss-Newton iterations for the unconstrained problem for essentially a similar cost per iteration.

3 Variants of Newton's Method and the SQP Framework

Consider the constrained minimization problem (6). Introducing the Lagrangian

$$\mathcal{L}(u,m,\lambda) = \frac{1}{2}\|Qu-b\|^2 + \frac{\beta}{2}\|W(m-m_{ref})\|^2 + \lambda^T[A(m)u-q] \tag{9}$$

where λ is a vector of length N_u of Lagrange multipliers, a necessary condition for a solution of our problem is that the first derivatives of the Lagrangian vanish,

$$\begin{aligned}
\mathcal{L}_u &= Q^T(Qu-b) + A^T\lambda = 0 && \text{(10a)}\\
\mathcal{L}_m &= \beta W^T W(m-m_{ref}) + G^T\lambda = 0 && \text{(10b)}\\
\mathcal{L}_\lambda &= Au - q = 0 && \text{(10c)}
\end{aligned}$$

where

$$G = G(u,m) = \frac{\partial(A(m)u)}{\partial m}.$$

The $N_u \times N_m$ matrix G is typically sparse in the usual case that A is sparse.

Next, consider Newton's method for solving the nonlinear equations (10). At a given iterate u, m, λ, the Newton correction direction is given by the solution of the linear system

$$\begin{pmatrix} Q^TQ & K & A^T \\ K^T & \beta W^T W + R & G^T \\ A & G & 0 \end{pmatrix} \begin{pmatrix} \delta u \\ \delta m \\ \delta\lambda \end{pmatrix} = - \begin{pmatrix} \mathcal{L}_u \\ \mathcal{L}_m \\ \mathcal{L}_\lambda \end{pmatrix} \tag{11}$$

where

$$K = K(m,\lambda) = \frac{\partial(A^T\lambda)}{\partial m}, \quad R = R(u,m,\lambda) = \frac{\partial(G^T\lambda)}{\partial m} \tag{12}$$

are two new matrices introduced as part of the second-derivative information. These matrices, like G, are large but very sparse, and they can be evaluated without need of numerical or automatic differentiation. We show this for the 1D multi-experiment MT example in Section 5 and for the 3D electromagnetic problem in [11].

The next Newton iterate is obtained by updating $m \leftarrow m + \alpha\,\delta m$, $u \leftarrow u + \alpha\,\delta u$, $\lambda \leftarrow \lambda + \alpha\,\delta\lambda$, where $0 < \alpha \leq 1$ is a step size. This step size is determined by a (weak) line search procedure, resulting in a sufficient decrease of an appropriate merit function at each iteration [18, 8, 5]. Under sufficient regularity, which we assume, $\alpha = 1$ when the current iterate is sufficiently close to the solution.

It is well-known that this Newton method is equivalent to an SQP basic method, where at each iterate the following quadratic problem is solved,

$$\min_{\delta m, \delta u} \quad \begin{pmatrix} \delta u & \delta m \end{pmatrix} \begin{pmatrix} \mathcal{L}_{uu} & \mathcal{L}_{um} \\ \mathcal{L}_{um}^T & \mathcal{L}_{mm} \end{pmatrix} \begin{pmatrix} \delta u \\ \delta m \end{pmatrix} + \begin{pmatrix} \mathcal{L}_u & \mathcal{L}_m \end{pmatrix} \begin{pmatrix} \delta u \\ \delta m \end{pmatrix} \tag{13a}$$

$$\text{s.t.} \quad \begin{pmatrix} A & G \end{pmatrix} \begin{pmatrix} \delta u \\ \delta m \end{pmatrix} = q - Au. \tag{13b}$$

SQP methods have been studied by many, e.g., [18, 16, 13]. The equivalence with the basic Newton method allows construction of practical variants of the latter. In particular, methods have been proposed for designing an appropriate merit function to evaluate the merit of a given iterate and carry out a line search, for constructing positive definite approximations of the matrix of second derivatives in (13a) on the tangent null-space of the constraints, for directly updating λ and for using reduced Hessians. Many of these techniques are adapted below in our special setting.

3.1 Solving the Linear System

Consider the basic KKT system (11) that must be solved at each step[1]. It is natural to apply a block elimination for δu and $\delta\lambda$; this is a simple instance of reduced Hessian methods, described in [16, 18].

From the last block of rows of (11) we write

$$\delta u = -A^{-1}[\mathcal{L}_\lambda + G\delta m]. \tag{14}$$

Next, substituting δu in the first block rows gives

$$\delta\lambda \;=\; [A^{-T}Q^TQA^{-1}G - A^{-T}K]\delta m + A^{-T}Q^TQA^{-1}\mathcal{L}_\lambda - A^{-T}\mathcal{L}_u. \tag{15}$$

Finally, from the second block rows we obtain a linear system for δm alone:

$$C\,\delta m \;=\; -p \tag{16a}$$

where

$$\begin{aligned} C &= C(u,m,\lambda) = J^TJ + \beta W^TW + R - S - S^T, && (16b)\\ J &= J(u,m) = -QA^{-1}G, && (16c)\\ S &= S(u,m,\lambda) = K^TA^{-1}G, && (16d) \end{aligned}$$

and

$$p = p(u,m,\lambda) = \beta W^TW(m - m_{ref}) + J^T(QA^{-1}q - b) - K^T(u - A^{-1}q). \tag{16e}$$

[1] In this paper we do not treat the question of solving this system *approximately.*

The $N_b \times N_m$ matrix J is the sensitivity matrix of (8). To see this, consider differentiating the discretized differential equation $Au = q$ with respect to m (assuming that $u = u(m)$, as in the unconstrained formulation). This gives

$$\frac{\partial(A(m)u(m))}{\partial m} = G + A\frac{\partial u(m)}{\partial m} = 0.$$

Thus, the derivative of the predicted data with respect to the model parameters is

$$\frac{\partial(Qu)}{\partial m} = Q\frac{\partial u(m)}{\partial m} = -QA^{-1}G = J.$$

Observe that in order to calculate the product of the sensitivity matrix times a vector we need not calculate J at all! Similarly for the matrix S. This is important for large scale problems where the linear problem obtained in each nonlinear iteration is solved using iterative methods [19, 10]. Given a vector v, the product Jv can be obtained in three stages:

1. calculate the product $w = Gv$;
2. solve the system $As = w$;
3. multiply s by $-Q$.

For calculating Sv we proceed similarly, replacing $-Q$ by K^T in the last step. The point is that the matrices G, A, Q and K are all very sparse, whereas J and S are not. Because solving $As = w$ can often be achieved rapidly and economically, the calculation of Jv or Sv can also be achieved very efficiently and does not demand J or S explicitly. This procedure has been used in [11] for the 3D electromagnetic problem.

If the explicit calculation of J or S is desired (e.g. it is useful for the 1D MT example) then the above procedure can be applied, with v standing for each column, in turn, of the identity matrix.

3.2 Method Variants

The special form of the constraints (5) yields not only u in terms of m,

$$u = A^{-1}q, \tag{17}$$

but also λ in terms of m and u:

$$\lambda = A^{-T}Q^T(b - Qu). \tag{18}$$

These formulas satisfy (10c) and (10a), respectively, and yield alternatives to the use of (14) and (15). We obtain four Newton method variants, all of which calculate δm by (16) and use it to update m.

1. **N1**: The basic Newton step already introduced calculates δm, δu and $\delta\lambda$ from (16), (14) and (15) respectively, and updates m, u and λ simultaneously.

2. **N2**: Calculate δm and δu from (16) and (14) and update m and u. Then use the adjoint equations (18) to update λ. It is assumed that (18) holds also for the initial iterate.

   This variant was proposed in [14, 13, 4]. Note that it is different from the least squares multiplier often used in SQP software [18, 16], because (10b) is ignored in the update (18). However, the least squares multiplier is unrealistically expensive to calculate in our context.

3. **N3**: Calculate δm and $\delta\lambda$ from (16) and (15) and update m and λ. Then update u by (17) to satisfy the constraints. It is assumed that (17) holds also for the initial iterate.

   This method maintains feasibility throughout the iteration process, that is, the fields u are those that arise from the model m.

4. **N4**: Calculate δm from (16) and update m; then update u by (17) and λ by (18). It is assumed that (17) and (18) hold also for the initial iterate.

   It can be easily verified that this variant is, in fact, Newton's method for the unconstrained problem formulation (7). Thus, we see that the costs of carrying out a basic iteration for the constrained and the unconstrained formulations are comparable.

One major potential difficulty with the method variants above is that the matrix of second derivatives of the Lagrangian in the SQP formulation (13a) may not be positive definite on the constraint null-space (i.e. for vectors $(\delta u, \delta m)$ which satisfy $A\delta u + G\delta m = 0$) when the iterate is not close to the solution. This translates to the reduced Hessian C of (16b) not being positive definite, and requires special care, e.g. by means of trust region methods [18, 14, 4, 16] that ensure the positive definiteness of the reduced Hessian.

Positive definiteness is immediately obtained by the Gauss-Newton approximation, where second derivative information is dropped. Thus, setting $K = 0$, $R = 0$, we obtain also $S = 0$ in (16d) and

$$C = C_{GN} = J^T J + \beta W^T W \tag{19a}$$

is positive definite. The direction vector $\delta m = \delta m_{GN}$ is now the solution of the linear system

$$\begin{aligned} C_{GN}\delta m_{GN} &= -p_{GN}, \quad \text{where} && (19b) \\ p_{GN} &= J^T(Q^{-1}q - b) + \beta W^T W(m - m_{ref}). && (19c) \end{aligned}$$

In fact, these are the normal equations for the over-determined linear least squares problem for δm,

$$\begin{pmatrix} J \\ \sqrt{\beta} W \end{pmatrix} \delta m_{GN} = - \begin{pmatrix} Q A^{-1} q - b \\ \sqrt{\beta} W (m - m_{ref}) \end{pmatrix}. \tag{20}$$

The explicit calculation of λ is now unnecessary as well, except possibly for the merit function evaluation (21) in the constrained case, because otherwise it appears only in the discarded K and R. We obtain two method variants, both using (20) to obtain δm.

1. **GN1**: Calculate δu using (14) and update m and u simultaneously.

   This is a Gauss-Newton method for the constrained formulation (6).

2. **GN2**: Update m, then use (17) to update u. It is assumed that (17) holds also for the initial iterate.

   This is the popular Gauss-Newton method for the unconstrained formulation (7), which has already been briefly described in the Introduction.

The advantage of the Gauss-Newton formulations is not only in positive definiteness of C. An additional advantage arises from the robust methods available for solving linear least squares problems. If J is small enough then the system (20) is best solved using the generalized SVD of J and W [9]. When the problem is large then one may use iterative methods such as CGLS or LSQR [12, 20]. In this case, however, we note that the cost of the Newton steps is only a fraction more than that of the Gauss-Newton variants; the choice of Newton vs Gauss-Newton alternatives does not depend (as commonly believed) on the relative efficiency of calculating one step, but rather, on the overall number of iterations required.

Also, in case that the sensitivity matrix, J, is required explicitly, the Gauss-Newton method may have some advantage over the Newton method. Note that it is possible to calculate J by rows, which are the columns of J^T, by solving first the adjoint problem:

$$A^T v_j = -Q_j^T \quad j = 1, \dots, N_b.$$

Then, a row of J is given by

$$J_j^T = G^T v_j.$$

Note that, since the matrix Q contains the locations of the data, this approach of calculating J is equivalent to the adjoint equation formulation [3, 17]. If the number of data N_b is smaller than the number of model parameters N_m (as in most remote sensing applications) then this formulation is better than the one which calculates J by columns. In case of the Newton iteration we have the matrix S to consider too, so the column approach for calculating the sensitivity matrix J is more natural.

It is well-known (e.g. [5, 18]) that for large residual problems or highly nonlinear problems the Gauss-Newton variants slow down and their employment may not be advisable. This is demonstrated in Section 5.

In the discussion above we have distinuished between method variants mainly according to the criterion of using second derivative information. The other criterion of importance is whether the method addresses the constrained or the unconstrained formulation. This turns out to be less important in our setting, but differences exist.

One simplification which the formulations that satisfy (5) at the end of each step offer is the ready availability of a merit function, namely the objective function of (7). Thus, line search is conducted for the method variants $N3$, $N4$ and $GN2$ using the unconstrained least squares objective function to evaluate the quality of the iterate. The line search in the other variants is more involved due to the min-max nature of the constrained optimization problem. It requires the choice of a merit function and there are various natural candidates [18, 16, 4]. In this paper we chose the commonly used l_1 penalty function

$$\phi_F = \|Qu - b\|^2 + \beta\|W(m - m_{ref}\|^2 + \mu\|Au - q\|_1 \tag{21}$$

where μ is a penalty parameter. The choice of μ, which typically depends on λ at each iteration, is discussed in [18, 13, 4].

Whereas the Newton variants presented in this section all have approximately the same cost per basic iteration, the cost of repeated line search is lowest for $N1$ and highest for the unconstrained formulation $N4$.

4 The Global Nonlinear Procedure

The methods developed in the previous section are for solving the nonlinear problem for a specific regularization parameter β. However, the complete inverse problem is more complicated since we do not know the value of the regularization parameter *a-priori.*

Recall the original ill-posed problem (2), (5). A way to consider the Tikhonov regularization [22] is through the constrained formulation

$$\begin{aligned} \min \quad & \frac{1}{2}\|W(m - m_{ref})\|^2 && \text{(22a)} \\ \text{s.t.} \quad & \|Qu(m) - b\|^2 \le Tol^2, && \text{(22b)} \end{aligned}$$

where $u(m)$ is obtained from (7) or (6). Forming the Lagrangian of (22) and differentiating to obtain the necessary conditions for an optimum, it becomes clear that β should in fact approximate the inverse of the Lagrange multiplier of the data fitting constraint. Since this multiplier is typically increased from 0 until equality is reached in (22b), it is natural to consider a practical procedure in which β is started at a large positive value and is then subsequently decreased until (2) is satisfied. The value of β

is thus related to the noise level in the data (recall (3)). It should not be "too small", to avoid fitting the noise rather than the data.

Moreover, for small values of β the nonlinear problem of minimizing (7) or (6) can be difficult to solve, because J^TJ is typically singular and/or ill-conditioned, while for large values of β this problem becomes easier, because the simple quadratic term $\frac{\beta}{2}\|W(m-m_{ref})\|^2$ dominates. Note that for an ill-conditioned problem the direction of the correction vector δm can be severely polluted in the computation. This direction is not altered by the line search procedure. Increasing β, on the other hand, does change the direction of the correction vector δm (cf. [5, 2]).

We therefore apply simple *continuation*, or *cooling* [1, 2, 10], in order to find the regularization parameter. We start with a relatively large value of β, for which we solve an almost quadratic problem, and we then gradually reduce β and solve each new problem with the solution for the previous β as a first iterate. The minimization process with a specific β is referred to as an *outer iteration* and the Newton or Gauss-Newton iteration within each outer iteration is referred to as an *inner iteration*. The algorithm is terminated when (2) is satisfied.

If we solve each nonlinear problem in this continuation sequence with a good starting point then each solution process converges quickly to its solution without need for line search (i.e. step sizes $\alpha = 1$ are used in the inner iterations). The local quadratic convergence property of Newton's method may then be recovered as well. Thus, we control the continuation in β by requiring that a sufficient decrease is obtained in the merit function at each inner iteration using no line search.

In this work β was reduced to be 0.1 of its previous value upon convergence of the inner iteration. If that value was deemed unacceptable, β was increased by the formula $\beta \leftarrow \beta + 0.5(\beta_{old} - \beta)$. For the inner iterations we set an iteration limit of 10.

5 Numerical Comparisons

In this section we compare the method variants presented earlier in their ability to solve a nonlinear inverse problem. We have selected a geophysical magnetotelluric (MT) example because (i) the MT equations are characterized by a Helmholtz operator which arises in a broad class of models, (ii) in 1D this is a multi source problem, which serves to demonstrate how various matrices and vectors are formed when using the differential equation approach, and (iii) the MT model is considered to be an archetypal problem for diffusive electromagnetic studies and correspondingly it has been extensively investigated [6, 19, 21, 23].

The MT forward model can be written in the frequency domain as a second order boundary value problem [21]

$$\begin{aligned} &E_{zz} - \imath\omega\mu_0\sigma(z)E = 0 \\ &E_z(0) = 1, \qquad E(\infty) = 0 \end{aligned} \tag{23}$$

where depth z is the independent variable, $\sigma(z)$ is the conductivity, E is the electric field and μ_0 is a given constant. In a typical multi-experiment, values of $E = E^k$ are observed for different frequencies $\omega = \omega_k,\ k = 1, \dots, n$, and these are used to reconstruct $\sigma(z)$.

Details of the problem reformulation and discretization can be found in the Appendix. In the end of this process we obtain a forward model of the form (5), with A a large, sparse matrix. This system can be solved rapidly for u, given m [2].

Next we generate synthetic data in the following way. We choose an idealized model $m = \ln(\sigma)$ given by

$$m(t) = -1 + 3e^{\frac{-(t-0.18\eta)^2}{0.01}} + 1.5e^{\frac{-(t-0.6\eta)^2}{0.02}} \tag{24}$$

where $\eta = 10^4$ and $t = \ln(z)$, $0 \le z \le 3 \times 10^5$. This is called the "true model" in Figure 4. Then we assume that the electric field is measured only on the surface of the earth, and we use (24) and (5) to generate value of $E(0)$ for ten logarithmically spaced frequencies from $0.01Hz$ to $100Hz$. This gives $N_b = 2n = 20$ data values, and they have been further contaminated with different levels of noise. The data are plotted in Figure 1.

The goal of the inversion, then, is to recover the conductivity model given the values of the electric field on the surface for these different frequencies. The model domain was discretized into $N_m = 90$ elements, uniform in t (thus exponentially increasing in depth z), and the length of u in this case is $N_u = 91{\times}4{\times}10 = 3640$.

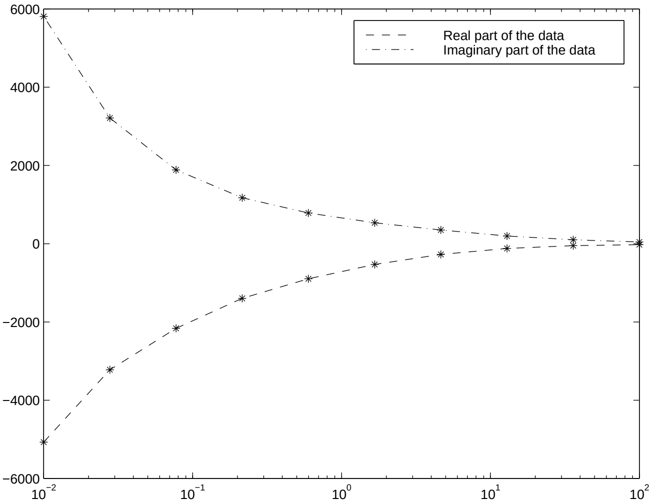

Figure 1: The real and imaginary parts of the data set for our experiment. Data are taken on the surface with different frequencies.

5.1 Numerical Experiment and Discussion

Next, we apply the methods developed in Sections 3 and 4 to the synthetic data described above. For an initial guess we choose $m_0 = m_{ref} = -0.5$, with u_0 and λ_0 subsequently calculated by (17) and (18). Thus, the initial point satisfies the forward model. In this experiment we choose W to be a combination of finite difference matrices

$$W = \begin{pmatrix} 1.5I \\ 3W_1 \\ 10W_2 \end{pmatrix}$$

where

$$W_1 = \frac{1}{\Delta t}\begin{pmatrix} 1 & -1 & & & 0 \\ & 1 & -1 & & \\ & & \cdot & \cdot & \\ 0 & & & 1 & -1 \end{pmatrix} \qquad W_2 = \frac{1}{\Delta t^2}\begin{pmatrix} -1 & 2 & -1 & & & 0 \\ & -1 & 2 & -1 & & \\ & & \cdot & \cdot & \cdot & \\ 0 & & & -1 & 2 & -1 \end{pmatrix}$$

where $\Delta t = \frac{\ln(z_f)}{N_m} = 0.14$. In order to evaluate an initial guess for the regularization parameter β, we estimate the largest generalized singular value of the sensitivity matrix $J(m_0)$ and the weighting matrix W (using one iteration of the power method). We then set β such that at the initial value, the leading term which corresponds to the misfit function in the Hessian, $(J(m_0)^T J(m_0))$, is very small (ratio of 10^4) compared with the regularization term, $\beta W^T W$. This condition translated to $\beta_0 = 2.5\times 10^9$. In order to assess the results of the different methods, we compare the following factors:

- The total number of inner iterations which were needed in order to converge. This number is given by the sum of all iterations for different β-values.

- The number of β-values (outer iterations) which were needed to achieve convergence within the tolerance.

The convergence criterion for all the algorithms was that the gradient has reduced to below 10^{-6} of its initial value.

In order to demonstrate the different properties of the techniques we calculate the results for a 2% noise (which implies a low residual problem) and for a 30% noise (which implies large residuals). The results of this experiment are presented in Table 1.

In Figure 2 we plot the convergence curve for one of the nonlinear problems (the inner iterations) which corresponds to the last outer iteration (the smallest β). For this iteration $\beta = 10^3$. Note that while the Newton step converges in 3-4 iterations and displays a quadratic rate of convergence, the Gauss-Newton process takes 6-7 iterations and displays linear convergence. This was typical to the whole process. In Figure 3 we plot the misfit and the model norm for each of the outer iterations. Finally, in Figure 4, we plot the model recovered in the process.

Table 1: Numerical Experiment 1: Comparison between methods.

Noise = 2% Method	GN1	GN2	N1	N2	N3	N4
Outer Iterations	8	8	7	7	7	7
Total Iterations	38	31	20	30	21	22
Noise = 30% Method	GN1	GN2	N1	N2	N3	N4
Outer Iterations	6	6	6	6	6	6
Total Iterations	178	157	19	22	19	20

The results in Table 1 indicate that when the noise is low, both Gauss-Newton type methods and the Newton type methods do well. However, when the noise is large Gauss-Newton methods do poorly while Newton type methods remain robust. This is expected because for large residual problems the second order terms are important for achieving rapid convergence [5, 8].

In the above experiments we did not see a significant difference between the constrained and the unconstrained approaches. This was true for a large amount of test problems we have experimented with. In order to see a difference we try to start the algorithm from an initial iterate very far from the minimum. We use the same weighting matrix but change both the starting iterate and the reference model to be $m_{ref} = -5 + \eta$, where η is a Gaussian random number with mean 0 and standard deviation 1. This model is not physical and probably would not be used for any inversion in practice, but we use it in order to demonstrate the different properties of the iterative techniques. The results for this experiment are presented in Table 2.

Table 2: Numerical Experiment 2: Comparison between methods.

Noise = 2% Method	GN1	GN2	N1	N2	N3	N4
Outer Iterations	15	10	14	9	9	10
Total Iterations	95	64	54	36	35	37
Noise = 30% Method	GN1	GN2	N1	N2	N3	N4
Outer Iterations	11	8	11	8	8	9
Total Iterations	233	188	30	27	27	30

From these numerical experiments we see again a major difference in performance for large vs low residual problems. Also, we observe a difference between the con-

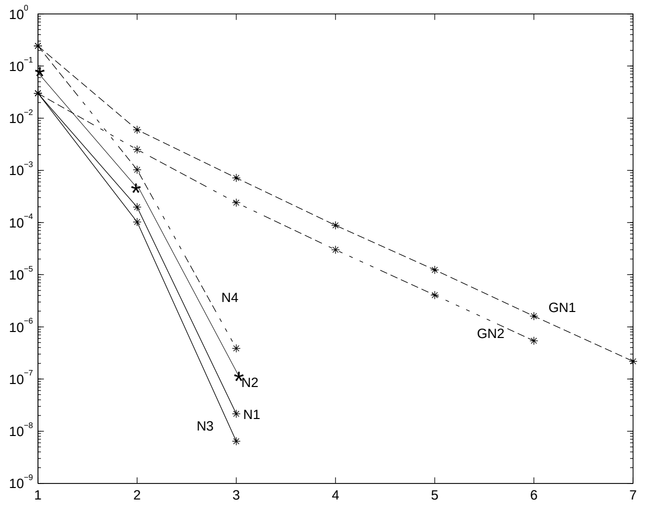

Figure 2: The norm of the gradient for each of the methods for the last β in the first numerical experiment.

strained and the unconstrained methods. In general, the unconstrained methods require more iterations than the constrained methods, which is expected for nonlinear models. However, it is important to note that for our 1D MT problem this difference has not been very pronounced.

6 Conclusions

We have considered various optimization techniques for solving nonlinear inverse problems where the associated forward problems are modelled by differential equations such as those arising in electromagnetic modelling. Whereas traditionally a Gauss-Newton method has been applied to the unconstrained formulation (7), here we have considered the more general constrained formulation (6). A number of Newton- (including SQP-) and Gauss-Newton type variants naturally arise, and we have evaluated their relative merit.

Our most important findings are:

- Understanding the discretization of the forward problem allows us to obtain first and second derivatives with a very low cost.

- Using sparse linear algebra techniques we need not calculate the sensitivity matrix (16c) explicitly. Matrix-vector products involving J can be rapidly obtained using sparse matrix operations, and potentially crucial storage difficulties are thus avoided as well.

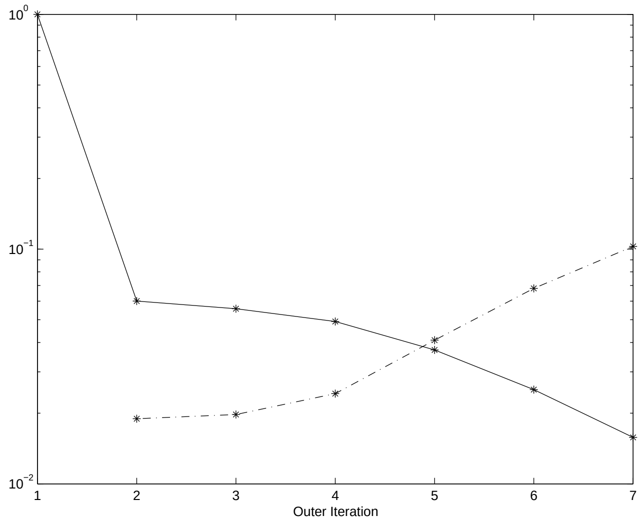

Figure 3: The misfit and model norms as a function of the outer iteration (Newton variants).

- In the present setting a full Newton step is only marginally more expensive to calculate than a Gauss-Newton step. Thus, in situations where Gauss-Newton methods slow down there are interesting, viable alternatives to consider.

Perhaps the most important advantages of the widened focus which this paper lays out are gained for very large problems, such as those arising in 3D EM. Here one must deal with hundreds of thousands of variables (unknowns). Not only storing J in full is out of the question, but also iterative linear algebra methods cannot be carried out to high accuracy if efficient overall execution is contemplated. Thus, inexact Newton-type methods must be considered. This raises a host of questions and alternatives which must be investigated and which are well beyond any one article. Here we make some general observations.

The Gauss-Newton method (19) does have important advantages, discussed in Section 3.2. The matrix C, in particular, is positive definite and has a special form which can be exploited in large calculations (e.g. [20]). But the explicit elimination of u, which implies an accurate solution of the forward model (5) (espcially when constructing the right hand side p_{GN}) can be a serious detriment. The constrained formulation (6) is more "loose" about this: no matrix inverse is present in the system of nonlinear equations (10), and the right hand side can thus be calculated cheaply and accurately. This suggests solving the KKT system directly by a preconditioned iterative method (e.g. [7]). We are currently engaged in evaluating the various questions involved, particularly as applicable in the 3D EM setting.

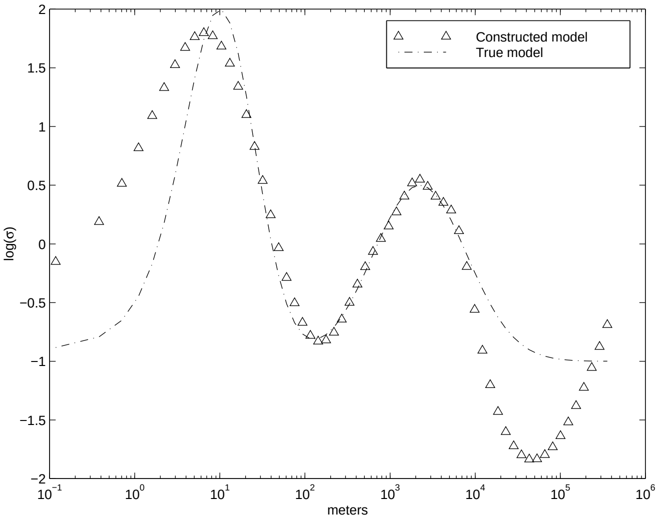

Figure 4: The reconstructed models.

7 Appendix

Here we specify the details of the discretization and setup of the 1D magnetotelluric example considered in §5.

7.1 Reformulating the equations

Consider the two-point boundary value ODE (23). When working with the constrained formulation it is easier to work with real quantities (since we need to differentiate with respect to E). Therefore, we set $E = u_1 + \imath u_2$. We also prefer to work with a first order system reformulation, so we set $u_3 = (u_1)_z$ and $u_4 = (u_2)_z$. Moreover, since we seek a positive conductivity $\sigma(z)$ which is usually spanned over a few orders of magnitude, we apply the common transformation $\sigma(z) = e^{m(z)}$. Finally, we define $r = \mu_0\omega$. This yields the real first order ODE system

$$\begin{aligned}(u_1)_z &= u_3 \qquad (25a)\\ (u_2)_z &= u_4\\ (u_3)_z &= -re^{m(z)}u_2\\ (u_4)_z &= re^{m(z)}u_1.\end{aligned}$$

In preparation for solving the problem numerically we first define the problem on a finite interval. The final depth z_f is chosen deep enough so that $u_i(z_f)\approx 0$, $i = 1, 2$. We set $z_f = 5d$, where d is the skin depth [21], $d = \sqrt{\frac{2}{\mu_0\sigma_0\omega}}$, with σ_0 an estimate of the lowest conductivity in any depth. For this choice the field at z_f is smaller than $e^{-5}{\approx}0.0067$ times its value at the surface. We used $\sigma_0 = 10^{-2}$, resulting in $z_f \approx 3{\times}10^5$. The equations in the infinite domain are now approximated by a system in the finite domain $[0, z_f]$, and we obtain the two-point boundary value problem consisting of (25a) and

$$u_3(0) = 1,\ u_4(0) = 0,\ \ u_1(z_f) = u_2(z_f) = 0. \tag{25b}$$

When more than one frequency $\omega = \omega_k$ is considered, and $r = r_k$ in (25), then the interval size $z_f = z_f^k$ depends on k as well. If the same interval is required for all $k = 1, \dots, n$ then set

$$z_f = 5\sqrt{\frac{2}{\mu_0 \sigma_0\ min_k(\omega_k)}}. \tag{26}$$

7.2 Discretization Using Finite Differences

For the inverse problem we approximate $m(z)$ as a piecewise constant function on a given grid

$$0 = z_0 < z_1 < \cdots < z_{N_m} = z_f$$

i.e., $m(z) = m_{i-1/2}$ for $z_{i-1} \le z < z_i$. This function is then represented as a vector

$$m = (m_{1/2}, \dots, m_{N_m - 1/2})^T.$$

We proceed by using the midpoint scheme for discretizing (25). If the same grid as of the model m is used then we get

$$\begin{aligned}
h_i^{-1}(u_{1,i} - u_{1,i-1}) &= (u_{3,i} + u_{3,i-1})/2 \\
h_i^{-1}(u_{2,i} - u_{2,i-1}) &= (u_{4,i} + u_{4,i-1})/2 \qquad 1 \le i \le N_m \\
h_i^{-1}(u_{3,i} - u_{3,i-1}) &= -re^{m_{i-1/2}}(u_{2,i} + u_{2,i-1})/2 \\
h_i^{-1}(v_{2,i} - v_{2,i-1}) &= re^{m_{i-1/2}}(u_{1,i} + u_{1,i-1})/2
\end{aligned} \tag{27a}$$

$$u_{3,0} = 1,\ u_{4,0} = 0,\ \ u_{1,N_m} = u_{2,N_m} = 0. \tag{27b}$$

There are numerous practical issues pertaining to the design of meshes for a multi-frequency problem where the penetration depth changes with frequency. Here we have chosen to use a single fixed grid for all frequencies. We construct the grid according to [2] for the highest frequency (finest grid and shortest z_f^k) and continue it further through the whole interval until z_f is reached. Using the same grid for all the experiments and for the model parameters makes the programming somewhat easier, but this is an issue which would have to be re-evaluated when working with large 2D or 3D problems.

Let u_k be the vector of unknowns of length $4(N_m + 1)$. The system (27) can be written as

$$A_k(m)u_k = q_k$$

where A_k is banded. In fact, A_k can be arranged in block form,

$$A_k = \begin{pmatrix} B_0 & & & & & \\ S_1 & R_1 & & & & \\ & S_2 & R_2 & & & \\ & & \ldots & \ldots & & \\ & & & S_{N_m-1} & R_{N_m-1} & \\ & & & & S_{N_m} & R_{N_m} \\ & & & & & B_1 \end{pmatrix}$$

where B_0 and B_1 are 2×4 and S_i and R_i are 4×4 blocks. In detail,

$$B_0 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad B_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix},$$

$$S_i = -h_i^{-1}I - T_i, \quad R_i = h_i^{-1}I - T_i, \quad T_i = \frac{1}{2}\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -r_k e^{m_i} & 0 & 0 \\ r_k e^{m_i} & 0 & 0 & 0 \end{pmatrix},$$

$$q_k = (1, 0, \ldots, 0)^T.$$

See Chapter 7 of [2], where techniques for solving such a system are presented and evaluated.

For n different frequencies, the different algebraic systems $A_k u_k = q_k$ can be bunched into one forward problem of size $N_u = 4n(N_m + 1)$. Clearly, $N_u \gg N_m$. The system now reads

$$A(m)u = q \tag{28}$$

where

$$A(m) = diag\{A_1(m), \ldots, A_n(m)\}.$$

7.3 Obtaining the Gradients and the Hessian

In order to calculate a Newton-type step we need to calculate the matrices G, K and R. In this subsection we describe these matrices in detail.

The matrix G is $N_u \times N_m$ and it is very sparse. It consists of blocks,

$$G = \begin{pmatrix} G_1 \\ \vdots \\ G_n \end{pmatrix}, \quad \text{where} \quad G_k = \frac{\partial(A_k u_k)}{\partial m} = \begin{pmatrix} 0_{2\times N_m} \\ G_{k,1} \\ \vdots \\ G_{k,N_m} \\ 0_{2\times N_m} \end{pmatrix}$$

is a $4(N_m+1) \times N_m$ block matrix. The $4 \times N_m$ block $G_{k,i}$, in turn, consists of zero columns except for the i^{th} column, which is

$$\begin{pmatrix} 0 \\ 0 \\ r_k e^{m_{i-1/2}}(u^k_{2,i} + u^k_{2,i-1})/2 \\ -r_k e^{m_{i-1/2}}(u^k_{1,i} + u^k_{1,i-1})/2 \end{pmatrix}.$$

The matrix K has a similar size and structure to G. It consists of blocks,

$$K = \begin{pmatrix} K_1 \\ \vdots \\ K_n \end{pmatrix}, \quad \text{where} \quad K_k = \frac{\partial(A_k^T \lambda_k)}{\partial m} = \begin{pmatrix} K_{k,1} \\ \vdots \\ K_{k,i} \\ \vdots \\ K_{k,N_m} \end{pmatrix}$$

is a $4(N_m+1) \times N_m$ block matrix. The $4 \times N_m$ block $K_{k,i}$, in turn, consists of zero columns except for the columns i and $i-1$, which are

$$\begin{pmatrix} -r_k e^{m_{i-3/2}}\lambda^k_{4,i}/2 & -r_k e^{m_{i-1/2}}\lambda^k_{4,i+1}/2 \\ r_k e^{m_{i-3/2}}\lambda^k_{3,i}/2 & -r_k e^{m_{i-1/2}}\lambda^k_{3,i+1}/2 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

We also divide λ_k into $N_m + 2$ vectors

$$\lambda_k = [\lambda^k_1, \ldots, \lambda^k_{N_m+2}]^T$$

where

$$\lambda^k_{:,i} = (\lambda^k_{4k-1}, \ldots, \lambda^k_{4k+2})^T$$

with $\lambda^k_{:,1}$ and $\lambda^k_{:,N_m+2}$ containing only two components each.

The matrix R can be calculated as

$$R = \frac{\partial(G^T\lambda)}{\partial m}$$

obtaining a diagonal matrix whose i^{th} value is given by

$$R_{i,i} = \frac{1}{2}\sum_{k=1}^{n} r_k e^{m_{i-1/2}} \left(\ (u^k_{2,i} + u^k_{1,i})\lambda^k_{3,i} - (u^k_{1,i} + u^k_{2,i})\lambda^k_{4,i}) \ \right).$$

Substituting these matrices into (11), or into one of its simplifications as per Section 3, we can now solve the MT inverse problem.

References

[1] J. C. Alexander and J. A. Yorke. The homotopy continuation method: numerically implementable topological procedures. *Trans Am Math Soc*, 242:271–284, 1978.

[2] U. Ascher, R. Mattheij, and R. Russell. *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations.* SIAM, 1995.

[3] G. Chavent and P. Lernonnier. Identification de la non-linearite' d'une equation parabolique quasi-lineare. *Applied Math and Opt.*, 1:121–162, 1974.

[4] J. E. Dennis, M. Heinkenschloss, and L. N. Vicente. Trust-region interior-point SQP algorithms for a class of nonlinear programming problems. *SIAM J. Control and Optimization*, 36:1750–1794, 1998.

[5] J. E. Dennis and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* SIAM, Philadelphia, 1996.

[6] S. Dosso. *Solution of the 1D Magnetotelluric Problem.* PhD thesis, University of British Columbia, 1990.

[7] H. Elman. Fast stokes solvers. 1998. Manuscript.

[8] P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization.* Academic Press, New York, 1981.

[9] G.H. Golub and C.F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, 1983.

[10] E. Haber. *Numerical Strategies for the Solution of Inverse Problems.* PhD thesis, University of British Columbia, 1997.

[11] E. Haber, U. Ascher, and D. Oldenburg. Solution of the 3D electromagntic inverse problem. In *Third Int. Symp. on 3D Electromagnetics*, Salt Lake City, Oct 1999.

[12] P. C. Hansen. *Rank Deficient and Ill-Posed Problems.* SIAM, Philadelphia, 1998.

[13] M. Heinkenschloss. Mesh independence for nonlinear least squares problems with norm constraints. *SIAM Journal on Optimization*, 3:81–117, 1993.

[14] M. Heinkenschloss. Projected sequential quadratic programming methods. *SIAM Journal on Optimization*, 6:373–417, 1996.

[15] K. Ito and K. Kunisch. The augmented lagrangian method for parameter estimation in elliptic systems. *SIAM J. Control Optim.*, 28:113–136, 1990.

[16] M. Lalee, J. Nocedal, and T. Plantenga. On the implementation of an algorithm for large-scale equality constrained optimization. *SIAM Journal on Optimization*, 8:682–706, 1998.

[17] P. R. McGillivray and D. W. Oldenburg. Methods for calculating frechet derivatives and sensitivities for non-linear inverse problem: a comparative study. *Geophys. Prospecting*, 38:501–512, 1990.

[18] J. Nocedal and S. Wright. *Numerical Optimization.* New York: Springer, 1999.

[19] D. W. Oldenburg. One-dimensional inversion of natural source magnetotelluric measurements. *Geophysics*, 44:1218–1244, 1979.

[20] C. C. Paige and M. A. Saunders. Lsqr: an algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Software*, 8:195–209, 1982.

[21] R. L. Parker. *Geophysical Inverse Theory.* Princeton University Press, Princeton NJ, 1994.

[22] A.N. Tikhonov and V.Ya. Arsenin. *Methods for Solving Ill-posed Problems.* John Wiley and Sons, Inc., 1977.

[23] K. P. Whittall and D. W. Oldenburg. *Inversion of Magnetotelluric Data for a One Dimensional Conductivity*, volume 5. SEG monograph, 1992.