12

ITERATIVE METHODS FOR INVERSE PROBLEMS

A. Roger and F. Chapel

12.1 Introduction

Iterative methods are widely used to solve problems in Electromagnetics and other fields. Very often the practitioner of inverse problems does not want to consider sophisticated mathematical details; instead, he tries to choose a particular algorithm in the huge number of algorithms available in the literature. Unfortunately this is a very difficult task for several reasons: first the properties of the various algorithms are often described through cumbersome and abstract theorems (at least they appear so to the physicist). Second the numerical results presented in papers and books often show contradictory aspects. In order to illustrate this point, let us cite a well known mathematician [1]:

> *"When an algorithm is presented in the optimization literature, it has usually been tested on a set of functions. The purpose of this testing is to show that the algorithm works and, indeed, that it works better than other algorithms in the same problem area. In our opinion these claims are usually unwarranted because it is often the case that there are only a small number of test functions, and that the starting points are close to the solution. . . . [It] can easily lead the cynical observer to conclude that the algorithm was tuned to particular functions. Even aside from the cynical observer, the algorithm is just not well tested.*

Third the features of the particular optimization problem to be solved are of high importance and interfere with the features of the algorithm itself, so that very often the reason of a success or a failing of a particular implementation remains obscure.

This chapter is intended to be a guideline for practitioners. Obviously only general properties can be presented at that stage, and the reader of this chapter will not be exempted from carefully studying the particular physical and mathematical features of his own problem; as stressed below, a clever physical study is most often the key of an efficient numerical solving. Yet general mathematical properties are of the highest interest for two reasons: they exhibit indisputable features of the algorithms, as opposed to numerical experiments which are often misleading unless they are performed on a very systematic and comprehensive basis, and they fix the limits of what can be expected from the use of an algorithm.

The title of this chapter emphasizes the application of Optimiza-

tion algorithms to *inverse* problems. Indeed, the needs are rather different in direct problems on the one hand, and in inverse problems on the other. In the former case, iterative techniques are most often considered as a means to avoid direct matrix inversion or refine the precision; the starting linear problem is reformulated as a problem of minimizing a quadratic function with known matrix and therefore known gradient and Hessian. In the latter case the problem is usually non linear and non quadratic, and the model to be inverted is "solved" by a computer code which acts as a black box; the computation of local approximations such as the gradient and the Hessian is not a trivial task and the construction of efficient algorithms is very complicated from a mathematical viewpoint. This chapter is devoted to *general* algorithms suitable for all problems; particular implementations will be described for every algorithm with special emphasis on application to Inverse Problems.

In the following the main families of Optimization algorithms are described together with their convergence properties. Technical terms and definitions have been avoided: the Optimization Theory constitutes by itself a large domain of Applied mathematics and the reader is referred to references for further information. Then some important practical points are stressed. Finally a few non-exhaustive examples are given to illustrate the theoretical considerations.

12.2 General Properties of Minimization Algorithms

a. The Standard Optimization Problem

The optimization algorithms deal with the general minimization problem:

$$\textit{minimize } F(X) : \Re^N \to \Re,\ X \in \Re^N \tag{1}$$

where F is a function from $\Re^N$ to $\Re$, i.e. a function of the vector X which has N real components. The algorithms are in charge of finding the vector X (i.e. the N real parameters) which minimizes F. The case of complex parameters reduces to (1) by considering real parts and imaginary parts as independent parameters.

Throughout the paper F is assumed to be sufficiently differentiable. For the sake of simplicity this paper will not deal with constrained optimization. For the same reason, we restrict ourselves to the Optimization problem in *finite dimensional spaces.*

Equation (1) states the problem in a purely mathematical manner. In practical problems one can often take advantage of some particular features of the objective function to implement particular minimization procedures. Moreover a good choice of the objective function is often crucial for the numerical efficiency. Similarly numerical rounding errors often damage the behavior of the algorithms. These practical features are examined in section 12.3. However the Optimization theory deals with the general properties of algorithms, independently from any numerical implementation. In this section we sketch some important results of this theory.

The vector X contains the N parameters to be retrieved. In inverse problems the objective function F is chosen to be an indicator of the gap between field data and simulated data. Any user of the algorithms is requested to yield a computer code capable of computing the value of $F(X)$ given any vector X.

In order to enhance the numerical efficiency, most algorithms additionally require the computation of the *gradient* $G(X_0)$, i.e. the vector of partial derivatives of F at the point X_0 with respect to the N parameters. Of course $G(X_0)$ may be evaluated by finite differences formulas; however there is no general rule to determine the stepsize to be used in these formulas: it must be small enough to make second order terms negligible and large enough to avoid round off errors in differences. Moreover the evaluation of the N partial derivatives requires the computation of at least $N+1$ values of the objective function; this is not feasible in most industrial problems because N can be very large and the objective function very expensive to compute. Fortunately it is often possible to compute directly the gradient with the help of *adjoint states* [2–3], requiring only two objective computations whatever N. In addition the adjoint state technique provides the *exact* value of the gradient, apart from rounding errors, as opposed to finite differences which imply a systematic error. This point will not be developed here.

The objective function F can be locally approximated by a Taylor expansion; the second order term is a quadratic form, the matrix of which is called the *Hessian* and will be denoted by $H(X_0)$. Finally we have:

$$F(X) = F(X_0) + G^t(X_0) \cdot (X - X_0) + \frac{1}{2}(X - X_0)^t \cdot H(X_0) \cdot (X - X_0) + \cdots \qquad (2)$$

with

$$(H(X_0))_{ij} := \frac{\partial^2 F}{\partial X_i \partial X_j} = (H(X_0))_{ji}$$

This expansion up to order two is called "the local quadratic model." Clearly when the objective function is quadratic the local model is the model itself and $H(X_0)$ does not depend on X_0.

b. Global and Local Methods

Inverse problems often involve very complicated objective functions with partially unknown properties. The problem (1) is that of finding a *global* minimum of the objective function; but this is a very difficult mathematical problem because there is no convenient theorem to ensure that a minimum is *global,* except for particular functions such as convex functions. Some numerical algorithms have been recently described to find *global* minima: tunneling methods [4], clusters methods [5], simulated annealing [6]. All these algorithms demand very large computational facilities even for simple objective functions and thus cannot be used in many industrial problems of interest. This paper will only be concerned with *local* optimization algorithms, i.e. algorithms able to find *local* minima of the given functional.

Local minima can be characterized by local properties, i.e. properties of the Gradient and the Hessian. Let us recall some elementary theorems [7]:

Necessary condition: if X_* is a local minimum, then at that point the gradient G_* is zero and the Hessian H_* is semi-positive definite (i.e. $\langle X, H_* X \rangle \geq 0 \forall X \in \Re^N$).

Sufficient condition: if at a given point X_* the gradient G_* is zero and the Hessian H_* is positive definite (i.e. $(\langle X, H_* X \rangle > 0 \forall X \in \Re^N, X \neq 0$), then X_* is a local minimum.

If F is quadratic H is semi-positive (else F would have no minimum at all); if in addition H is positive then F has only one global minimum.

Definition: an algorithm is said to be **globally convergent** if, *for any starting point* X_0 the sequence $\{X_k\}$ generated by the algorithm (or a sequence extracted from it) converges towards a point X_* with zero gradient.

Note that this definition takes into account only the stationarity condition of the first order; few theorems involving the stationarity

condition of the second order are available in the mathematical literature.

The property of global convergence is a criterion of robustness of the algorithm. It is very important to keep in mind that it does **not** imply the convergence toward a *global* minimum but only towards a *local* minimum (under additional assumptions).

c. *Convergence Speed*

This criterion is a means for evaluating the efficiency of an algorithm. Let us consider a sequence $\{X_k\}$ converging towards X_* and let $|.|$ denote the usual euclidian norm in $\Re^N$.

Definitions: the convergence is **linear** if:

$$\limsup_{k\to\infty} \frac{|X_{k+1} - X_*|}{|X_k - X_*|} = \alpha < 1$$

The convergence is **superlinear** if:

$$\exists \gamma > 1 : \limsup_{k\to\infty} \frac{|X_{k+1} - X_*|}{|X_k - X_*|^\gamma} < +\infty$$

In particular for $\gamma = 2$ the convergence is said to be **quadratic.**

d. *The Least Squares Case*

This case, often encountered in practice, occurs when the objective function F is a sum of M squares, i.e.:

$$F(X) = \frac{1}{2}\langle V(X), V(X)\rangle, \qquad V(X) \in \Re^M$$

Then the gradient and the Hessian take a special form. Let $J(X)$ be the Jacobian of V, i.e. the $M \times N$ matrix the coefficient J_{ij} of which is the partial derivative of the i-th component of $V(X)$ with respect to the j-th component of X. The gradient is given by:

$$G(X) = J^t(X) \cdot V(X)$$

where the subscript t denotes transposition of the matrix. The Hessian is given by:

$$H(X) = J^t(X) \cdot J(X) + V(X) \cdot K(X) \tag{3}$$

where $K(X)$ is a third order tensor containing the second order derivatives of the components of $V(X)$ with respect to the components of X.

12.3 Some Classes of Minimization Algorithms

In the following we describe some techniques which are used to construct minimization algorithms. Some of these techniques are mutually exclusive, and some are not: in fact such a large number of algorithms has been desribed in the literature that they cannot be described all or even mentioned here.

a. The Line Search Strategy

Most of the algorithms are based upon the so-called Line Search Technique: at the k th step one computes a *descent direction* $D_k \in \Re^N$ and then one solves the problem:

$$\textit{minimize } f_k(q) := F(X_k + qD_k), \qquad q \in \Re \tag{4}$$

In other words at every step a one-dimensional minimization is performed ("line search"). It is worth noting that when F is quadratic and its Hessian is known, which is the case when solving matrix equations, the line search is useless since the distance q can be exactly expressed (see for instance reference [8]). Some considerations based on a physical approach such as the Born approximation [9] can also provide a value for q. When the Hessian is unknown or the objective function is not quadratic, the line search must be performed numerically and this is the key of the efficiency of the algorithm.

When the optimized parameters derive from the different physical quantities, they can be divided into subspaces; the gradient vector is then cut into parts, associated to each class, which constitute a local subspace [10]. The search at each step involves the Hessian whose smaller dimensionality facilitates its inversion and the control of its conditioning. This multilinear search has proven its efficiency in geophysical inverse problems where parameters such as velocities and reflectivities affect the seismograms in a different way. A seismic recording is approximately linearly dependent on the reflectivities whereas the influence of velocities is strongly non linear. The bad conditioning of the problem is increased by the different orders of magnitude

within the parameters. D'Aboville [11] studied multilinear search algorithms with descent directions chosen as gradients, conjugate gradients or Quasi-Newton. The conclusion of his work states the superiority of the Quasi-Newton method compared to gradient methods as long as the computation of the Hessian relative to each class of parameters is not too costly. However, proofs of stability and convergence of multilinear search algorithms are still to be established.

b. Gradients and Newton Algorithms

In the following we present various algorithms which generate a sequence $\{X_k, k = 0, +\infty\}$ of iterates; G_k and H_k denote respectively the gradient and Hessian of F at X_k.

The conceptually simplest algorithm is the steepest descent algorithm, in which one sets:

$$D_k = -G_k$$

Conjugate gradient algorithms are more sophisticated methods, featured to minimize quadratic objective functions. In these algorithms the descent direction is given by formulas such as:

$$\text{for } k \neq pN : D_k = a_{k-1} D_{k-1} + B_{k-1}, \qquad a_{k-1} \in \Re,\ B_{k-1} \in \Re^N \tag{5}$$

$$\text{for } k = pN : D_k = -G_k \tag{6}$$

where a_k and B_k are explicitely given in terms of known quantities such as previous steps and gradients. Note that the algorithm is "restarted" every N iterations according to (6). Various formulas are available in the literature ([8], [12–15]); the interested reader is referred to other chapters of the book for further details. The point is that conjugate gradients take into account not only the first order information (gradient) but also the second order information which is contained in the variation of the gradient.

In the primary Newton algorithm the next iterate is given by:

$$X_{k+1} = X_k - H_k^{-1} \cdot G_k \tag{7}$$

However this algorithm is substantially improved by the use of the Line Search technique, with a descent direction given by:

$$k \neq pN : D_k = -H_k^{-1} \cdot G_k \tag{8}$$

$$k = pN : D_k = -G_k \tag{9}$$

In other words the minimum is searched for in the direction of the Newton point $-H_k^{-1} \cdot G_k$ with periodic restarting; this technique makes sense only when the Newton point is actually a minimum of the local model, i.e. when H_k is positive definite. Note that if the objective function is quadratic and the Hessian exactly known, the exact minimum is found in only one step, but at the price of the inversion of the Hessian matrix. The generalization of this algorithm to functional spaces is called the Newton-Kantorovich algorithm [16].

However very often the Hessian is not known or far too costly to compute. Some techniques have been developed in order to calculate approximations of the Hessian, called Quasi-Hessians; the corresponding algorithms are called *Quasi-Newton* algorithms. The Quasi-Hessian H_k^q at the k^{th} iteration is calculated by additive concessions of the form:

$$H_k^q = H_{k-1}^q + a_k u_k u_k^t + b_k v_k v_k^t \qquad a_k, b_k \in \Re,\ u_k, v_k \in \Re^N$$

The most famous formulas for a_k, b_k, u_k, v_k are the rank-one formula [7], the DFP formula [17] and the BFGS formula [18]. These formulas are constructed so that when the Hessian is constant (i.e. F is quadratic) the Quasi-Hessian is exactly equal to the Hessian for $k \geq N$. It is worth noting that similar formulas also provide sequences of inverses of Quasi-Hessians, thus allowing simple implementations of Quasi-Newton algorithms.

In the least square case, the Gauss Newton approximation is very popular:

$$H_k^q = J_k^t \cdot J_k \tag{10}$$

because it provides a positive or semi-positive approximation of the Hessian just from first order information (the Jacobian). In view of (3), it is clear that this approximation is all the more better as the residuals (components of $V(X)$) are small.

The Quasi-Newton and Gauss-Newton formulas can be used with the Line Search strategy as well as with the Trust Region strategy.

Some authors have also proposed intermediate algorithms between Quasi-Newton and conjugate gradient methods [19].

c. The Trust Region Strategy

A new strategy has been developed in the last ten years. It is the so-called **Trust Region Method,** which provides an alternative

to the Line Search Strategy and is particularly well adapted when the modeling code is highly time-consuming. It involves no descent direction; instead, at every step one solves the following problem:

$$\textit{minimize } Q_k(\delta X) := G^t(X_k) \cdot \delta X + \frac{1}{2}\delta X^t \cdot H(X_k) \cdot \delta X \qquad (11)$$

with

$$\delta X \in \Re^N, \delta X \leq \Delta_k$$

In other words, the local quadratic model is minimized inside a sphere ("Trust Region") of radius Δ_k ("Trust Radius") around the current point X_k, yielding a trial step δX_k. In (11), $H(X_k)$ may be either the exact Hessian or an approximation of it obtained with a Quasi-Newton formula or the Gauss-Newton formula.

Then the value of the "quality coefficient" r_k defined by:

$$r_k := \frac{F(X_k + \delta X_k) - F(X_k)}{Q_k(\delta X_k)}$$

is computed; r_k is the ratio of the actual reduction of the objective function to the reduction expected from the quadratic model. Then the iterate is updated according to the following rule:

1: If $r_k < s$, then $X_{k+1} = X_k$

2: If $r_k \geq s$, then $X_{k+1} = X_k + \delta X_k$

where s is a fixed parameter which must be such that $0 < s < 0.25$ to ensure good convergence properties. By the same time the trust radius Δ_k is updated according to the following rule:

1: If $r_k \geq \eta$, then $\Delta_{k+1} = \gamma_2 \Delta_k$

2: If $\mu < r_k < \eta$, then $\Delta_{k+1} = \Delta_k$

3: If $r_k \leq \mu$, then $\Delta_{k+1} = \gamma_1 |\delta X_k|$

where $\eta, \mu, \gamma_1, \gamma_2$ are fixed real numbers which must satisfy $0 < \mu < \eta < 1$ and $0 < \gamma_1 < 1 < \gamma_2$ to ensure good convergence properties. Many Trust Region based algorithms are described in the literature [20–24]. Historically, Levenberg [25] first introduced a "regularization parameter" γ_k in order to treat least squares problems with badly conditioned Hessians; the iterate was given by a slight modification of (7):

$$X_{k+1} = X_k - (H_k + \lambda_k D)^{-1} \cdot G_k \qquad (12)$$

where D is a diagonal positive definite matrix. The Gauss-Newton approximation (10) was used. Later, Marquardt [26] interpreted the

parameter λ_k as a Lagrange multiplier associated with a constraint on the minimization. In most modern versions of this algorithm the equation (12) is replaced by the problem (11); this still leads to an equation similar to (12), but the parameter λ is now estimated from the trust radius Δ_k, the control of which is much easier; in addition strong convergence proofs are well established for these algorithms.

Trust Region Algorithms demand the computation of the gradient and the Hessian or an approximation of it; in addition the minimization of the local model is not that simple. On the other hand these algorithms offer very strong theoretical properties of convergence and their efficiency has been proven by numerous numerical experiments. Moreover, as opposed to the line search, the minimization of the local model makes no use of the modeling code and thus requires much less objective computations.

d. Convergence Theorems

Theorem 1: the Steepest Descent, the Conjugate Gradient, the Newton and Quasi-Newton algorithms *with Line Search* are **globally** convergent under additional hypotheses on the accuracy of the line Search.

Theorem 2: the Newton algorithm without Line Search is **not** globally convergent.

These fundamental properties follow from a general theorem of Zangwill [27]. The key of the theorem is here the Line Search, which must obey a rule ensuring a sufficient decrease of the objective function at every step. The best-known rules are those of Goldstein [28], Wolfe [29] and Powell [30]. Note also that the property of global convergence for Conjugate Gradients or Newton algorithms follows from that of the Steepest Descent algorithm, owing to the periodic restart of the descent direction ((4) and (6)).

Theorem 3: the Trust region algorithms are **globally** convergent, provided that some conditions on the Hessians or Quasi-Hessians and the accuracy of the minimization of the local model [20], [23] are satisfied.

The global convergence expresses the soundness of the working of the algorithm. But one may also question the efficiency of the algorithm: this feature is related to the convergence speed. The following properties hold:

1. The steepest descent has linear convergence [31]

2. The Conjugate Gradients have superlinear convergence in N steps, i.e.:

$$\frac{|X_{k+N} - X_*|}{|X_k - X_*|} \to 0 \text{ as } k \to +\infty$$

If in addition the Hessian satisfies a Lipschitz condition:

$$\exists k \in \Re : H(X) \cdot Y \leq k \cdot Y \quad \forall X, Y \in \Re^N$$

then the convergence is quadratic [32].

3. The Trust Regions have superlinear convergence in one step [33], i.e.:

$$\frac{|X_{k+1} - X_*|}{|X_k - X_*|} \to 0 \text{ as } k \to +\infty$$

These properties clearly show that in principle the Quasi-Newton methods (both with Line Search and Trust Regions) require $1/N$ fewer steps than the Conjugate Gradients for the same asymptotical behavior. However the Conjugate Gradients may be accelerated, especially by preconditioning techniques.

Many of the above theorems still hold when the Hessian or Quasi-Hessian is approximately calculated. Some of them also extend to the case of approximate gradients.

12.4 Practical Features

a. Definition of the Objective Function

The behavior of algorithms depends strongly on the kind of objective function to be minimized.

Since a minimum is searched for, an obvious requirement is that the objective function must be chosen such that it increases in any direction at infinity. If this condition is not satisfied, the algorithms will not converge for some starting points because the iterate point X_k will go to infinity in a direction where the objective function is decreasing.

Another difficulty which is frequently encountered is that an involved objective function often has a large number of **local** minima in which the local minimization algorithms fall (section 12.5.2). In other words, the practitioner observes the *non-uniqueness* of the solution. Obviously this difficulty does not occur when solving matrix equations

since the objective function is quadratic. Generally speaking, the best antidote—easier said than done—is to change the definition of the objective function so as to suppress the local minima; in particular, working with a smaller number of parameters, if possible, has often a good effect (example in section 12.5.d). One may also cope with this difficulty by imposing some constraints on the parameters and starting from a better first estimate. Some examples are given below.

From the numerical viewpoint, mixing parameters of different physical nature in the same vector X can be very troublesome and lead to large numerical errors. The various parameters should be first normalized in values and in variations on a physical basis. This can be achieved with a preconditioning technique, which is described in section 12.4.c.

The general theorems which are given in section 12.3 often take into account errors in the evaluation of the gradient; however they assume that the objective function is exactly calculated. Unfortunately this is not the case in many inverse problems where the objective function is corrupted by numerical errors. This phenomenon is often predominant in the neighbourhood of a minimum, so that objective functions become oscillatory. In such a case an efficient stopping criterion should be supplied in order to avoid useless computations.

Very often the inclusion of *a-priori* knowledge is a good way to improve the convergence of the algorithm. This can be done by putting simple constraints on the parameters [22], [34]: for instance a permittivity or a conductivity must obviously be positive! Sometimes adding a regularization term to the objective function [16] facilitates its minimizing and removes numerical instabilities.

b. Computational Price of the Objective Function

The choice of the minimization algorithm is mainly determined by the kind of objective function to be minimized: if it can be evaluated very cheaply, the use of a global minimization algorithm can be envisaged. But in most industrial problems the code which computes the objective function is very complicated and goes through the resolution of partial differential equations, so that the resources dedicated to the optimization algorithm itself are not significant. In such a case one must save as much as possible the computations of objective functions and gradients. From this viewpoint Trust Region algorithms are very attractive because they make full use of the local model at each step.

(See section 12.5.b.)

c. Number of Parameters

From a practical point of view the order of magnitude of N influences the choice of the minimizing algorithm: given some computational facilities, more complex algorithms require more memory storage and/or CPU time (the time devoted to the direct problem being not considered at that stage) and thus can deal with smaller values of N. For values lower than or equal to 100, the implementation of any of the algorithms described above is troubleless even on a personal computer. For larger values Quasi-Newton algorithms have an important drawback: they require to store and to handle a $N \times N$ matrix; in addition the Trust Region strategy leads to a constrained minimization of a quadratic problem. These operations may be time-consuming. As for us, we were able to implement a full version of a Trust Region algorithm (including computations of eigenvectors of the Quasi-Hessian) on a CRAY 1 computer up to $N = 500$. More sophisticated trust region algorithms have been implemented up to $N = 10000$ [35]. For very large values of N only the conjugate gradient algorithms can be employed; with today's computers a value of $N = 100000$ can be dealt with.

d. Scaling and Preconditioning

The basic idea of preconditioning is to introduce a preliminary scaling on the vector of parameters and the objective function. The problem (1) now becomes:

$$\textit{minimize } \tilde{F}(\tilde{X}) : \Re^N \to \Re, \tilde{X} \in \Re^N \tag{1}$$

with

$$\tilde{X} := D \cdot X, \ \tilde{F}(\tilde{X}) := \alpha F(D^{-1} \cdot X) \tag{13}$$

where D is a non singular positive matrix and α a positive scalar. From a theoretical point of view, the scaling matrix is useless and makes the algorithm less clear. However from a practical point of view it is of the highest importance to ensure quick convergence. The steepest descent algorithm as well as the Conjugate Gradient algorithms and the Quasi-Newton algorithms [20] may incorporate preconditioning. Moreover it has been demonstrated that the preconditioned Conjugate

Gradient algorithms are in some sense equivalent to the Quasi-Newton algorithms.

The effect of a good preconditioning can be understood by examining its effects on a quadratic objective function; the gradient and the Hessian of the transformed function are easily expressed as:

$$\tilde{G} = \alpha G \cdot D^{-1^t} \qquad \tilde{H} = \alpha D^{-1^t} \cdot H \cdot D^{-1}$$

Now if a decomposition of the Hessian is known or calculable:

$$H = D_1^t \cdot D_1 \tag{14}$$

setting $D = D_1$ makes the transformed Hessian proportional to the identity matrix; all the methods described above (Steepest Descent, Conjugate Gradients, Quasi-Newton) yield the same descent direction and converge at the first step.

A decomposition such as (14) can be provided by a SVD analysis, i.e. by computing the eigenvectors and eigenvalues of H. However this is a very time-consuming task; thus, more often, preconditioning is implemented on the basis of physical considerations. Indeed, in the framework of inverse problems, preconditioning allows one to rescale parameters of different physical nature. Moreover it may be a way to get close to a quadratic behavior and thus have more numerical efficiency. An example will be given below.

Preconditioning techniques are very useful in practical problems. However very few means are described in the literature to set the preconditioning matrix on the unique basis of mathematical considerations. In fact the most significant indications often come from the physics of the problem, especially for inverse problems.

Figure 12.1 stresses this point with the particular example of the 1D Electromagnetic time dependent inverse problem of the stratified dielectric medium. In this case, the measured signal can be expressed as the convolution of the reflection function (derivative of the optical index with respect to the travel time) by the normal derivative of the field along certain characteristic lines [36] [37]. The result is rigorous and holds even in the case of strong contrasts. Of course the field is a priori unknown in the inverse problem; however *approximations* of it are known and can be used as preconditioning matrices D_k. This technique was already applied in the harmonic inverse problem of the stratified medium [38–39]. Figure 12.1 shows a result obtained with

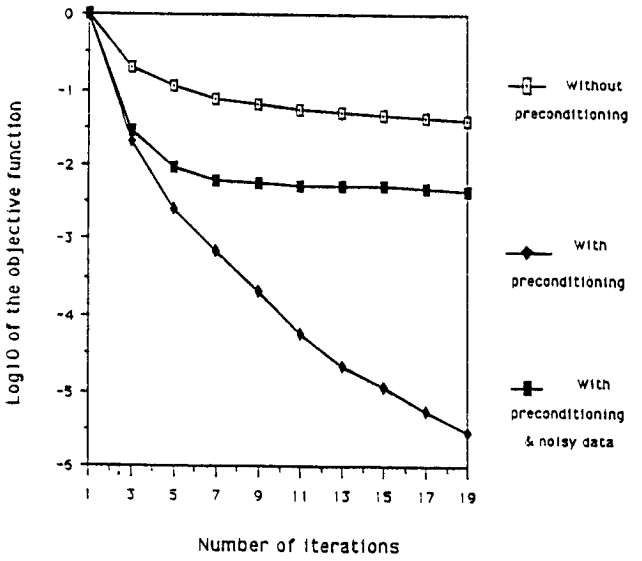

Figure 12.1 Logarithm of the objective function vs the number of iterations. The model is a 1D horizontally stratified medium probed by a time-dependent electromagnetic source. A finite-difference scheme is used to solve the direct problem. The objective function is the L_2 norm of the difference between the actual and computed reflected fields. The minimization algorithm is a Conjugate Gradient using the Fletcher-Reeves formula with Line Search. Note that even in presence of noise the convergence is more rapid with preconditioning.

the Fletcher-Reeves Conjugate Gradient algorithm: the convergence is much quicker with preconditioning and hence permits to gain accurate reconstructions in $N/10$ iterations, a behavior much better than that guaranteed by the general theorems of convergence.

e. Stability and Robustness

The general theorems presented above ensure the global convergence; however the algorithms may yield rather different results, especially if the data are noisy or if the computation of the objective function is corrupted by numerical errors. It is generally admitted that the Quasi-Newton algorithms are "less stable" than the Conjugate Gradients; indeed when the Hessian is badly conditioned, which may occur especially if some of the components of X are non-significant parameters, then the algorithms involving the inverse of the Hessian

are more sensitive to noise. This point was confirmed by numerical experiments [11]. Not surprisingly, preconditioning may also make the algorithm less stable [37].

12.5 Examples

Implementing a fair comparison between minimization algorithms is for sure a complicated task. The first difficulty, and perhaps the most important one, is that different computer codes generally implement different stopping criterions, so that the computing costs are hardly comparable. A second difficulty is that all computer codes require additional parameters in input, such as for instance the length of the step to be used in the first Line Search; a fine tuning of these parameters improves the convergence but this tuning varies with the considered objective function. A third difficulty was already mentioned in the introduction: the testing is significant only when a very large number of objective functions is considered, together with many different starting points. So the final amount of work is quite considerable because the coding of objective functions is a tedious and bothersome work in itself, not to mention that every minimization code requires the objective function to comply with a specific calling sequence.

In order to acquire a well founded opinion, the practitioner should read carefully the extensive testings already available in the literature [1] [40] [41]. Giving a comprehensive set of results would be far beyond the scope of the chapter. Instead, we only provide a few results extracted from our own experiments [24] [37] [42]–[44].

a. Comparison of a Conjugate Gradient and a Quasi-Newton Algorithm

In the following several conventional objective functions are used. The definitions of these functions have been chosen [1] [40] so as to check the algorithm in front of a specific difficulty (singular Hessian, saddle point, circular valley, ...).

The quadratic form is given by:

$$F(X) = \frac{1}{2}|A \cdot X - B|^2$$

with A a real $N \times N$ and $B \in \Re^N$

The circular valley function is given by:

$$F(X) = \left(|X|^2 - 1\right)^2 + 10^{-2}(X_1 - 1)^2$$

where $X_1 \in \Re$ is the first component of X.

Powell's generalized objective function is given by:

$$N = 4p,\ F(X) := \sum_{i=1}^{N} (F_i(X))^2$$

where the F_i s are given by:

$$\begin{aligned} F_{4q+1}(X) &:= X_{4q+1} + 10X_{4q+2} \\ F_{4q+2}(X) &:= \sqrt{5}(X_{4q+3} - X_{4q+4}) \\ F_{4q+3}(X) &:= (X_{4q+2} - 2X_{4q+3})^2 \\ F_{4q+4}(X) &:= \sqrt{10}(X_{4q+1} - X_{4q+4})^2 \end{aligned}$$

where $0 \leq q < p$ and $X_i \in \Re$ is the i^{th} component of X.

Powell's objective function is the same as Powell's generalized objective function with $p = 1$.

Brown's objective function is given by:

$$N = 2,\ F(X) := (X_1 - 1/Q)^2 + (X_2 - 2Q)^2 + (X_1X_2 - 2)^2$$

with $Q = 10^{-6}$.

The following tables show the results obtained by minimizing these cost functions first with the help of a conventional Conjugate Gradient algorithm (Fletcher Reeves) then with a conventional quasi-Newton algorithm (B.F.G.S.), both using the Line Search strategy. The "F" column gives the final value of the objective function; the "$|G|$" column gives the final norm of the gradient; the "n_F" column gives the number of function evaluations; the "n_G" gives the number of gradient evaluations.

The use of underlining indicates that the algorithm has not converged.

C.G. Algorithm	F	$\|G\|$	n_F	n_G
Quadratic form	3.07×10^{-4}	3.01×10^{-1}	146	116
Circular Valley	4.58×10^{-9}	5.00×10^{-6}	376	255
Powell's	2.43×10^{-9}	1.38×10^{-5}	342	253
Generalized Powell's	205.4	3757.8	2008	1505
Brown's	1.38×10^{-15}	7.45×10^{-8}	50	28

Table 12.1 Results obtained with a Conjugate Gradient algorithm.

Q.N. Algorithm	F	$\|G\|$	n_F	n_G
Quadratic form	5.77×10^{-3}	1.05×10^{-1}	50	50
Circular Valley	2.47×10^{-12}	3.75×10^{-7}	1234	1234
Powell's	1.31×10^{-12}	1.39×10^{-7}	1606	1606
Generalized Powell's	6.28×10^{-14}	3.78×10^{-7}	1040	1040
Brown's	9.99×10^{11}	1.99×10^{7}	6	6

Table 12.2 Results obtained with a Quasi-Newton algorithm.

Tables 12.1 and 12.2 suggest that the Quasi-Newton algorithm requires much less evaluations of the objective function than the Conjugate Gradient algorithm when applied to a quadratic objective function. Remember that here the matrix of the quadratic form is supposed to be unknown, so that an elaborated line search is performed at every step; the results can be different when the matrix is known, because no line search is then required.

With the other objective functions, the Conjugate Gradient algorithm proved better than the Quasi-Newton algorithm. Note that these results can hardly be generalized to other implementations of the same algorithms. In particular, our experiments showed that the number of function and gradient evaluations depends heavily on the skill of the Line Search which is performed at every step.

b. Comparison of a Line Search and a Trust Region Algorithm

We constructed a Trust Region algorithm in order to solve geophysical inverse problems. The model is a 2D horizontally stratified acoustic medium probed by a time-dependent acoustic source. It is modeled using a finite-element technique, with about 10000 elements and 400 time steps; 200 seismograms were computed in 30 seconds on the CRAY 1 computer of the Elf-Aquitaine company. In this context, the algorithm was designed to make full use of the local information (objective function value, gradient and Hessian), so as to save solvings of the direct problem. A quasi-Hessian was calculated at every step with the help of the rank-one updating formula [7] which appeared to be more adapted to the Trust Region strategy than the BFGS formula. The minimization of the local quadratic model (11) was performed using a full SVD decomposition of the Hessian; we took advantage of

the features of the rank-one formula to implement a special algorithm, allowing us to directly calculate the SVD decomposition of H_k from that of H_{k-1} [22].

The following tables show the results obtained by minimizing conventional cost functions first with a Quasi-Newton algorithm using the BFGS formula and the Line Search technique, and second with our Trust Region algorithm. The notations are the same as in the preceding paragraph.

Line Search	F	n_F	n_G
Quadratic form	2.41×10^{-8}	148	77
Circular Valley	1.86×10^{-7}	116	51
Generalized Powell's	1.11×10^{-2}	107	61
Brown's	9.32×10^{-11}	69	33

Table 12.3 Results obtained with a Line Search algorithm.

Trust Region	F	n_F	n_G
Quadratic form	2.12×10^{-8}	104	99
Circular Valley	1.45×10^{-7}	148	115
Generalized Powell's	9.19×10^{-3}	22	18
Brown's	1.29×10^{-11}	34	27

Table 12.4 Results obtained with our Trust Region algorithm.

Tables 12.3 and 12.4 show that, apart from the particular case of the circular valley, our Trust Region algorithm requires much less function and gradient evaluations than the Line Search algorithm.

The two algorithms were then applied to the Geophysical inverse problem of interest. Figure 12.2 presents the reconstructed profiles for velocities, the initial profile being a constant one. Note that both algorithms yield profiles very similar but different from the actual profile; yet the objective function has actually been minimized, which indicates the existence of a *local* minimum. Figure 12.3 presents the profiles reconstructed by the two algorithms when the initial profile is chosen as a ramp function: it shows that using a better first estimate enables the two algorithms to reconstruct the solution satisfactorily. The density profiles show the same behavior.

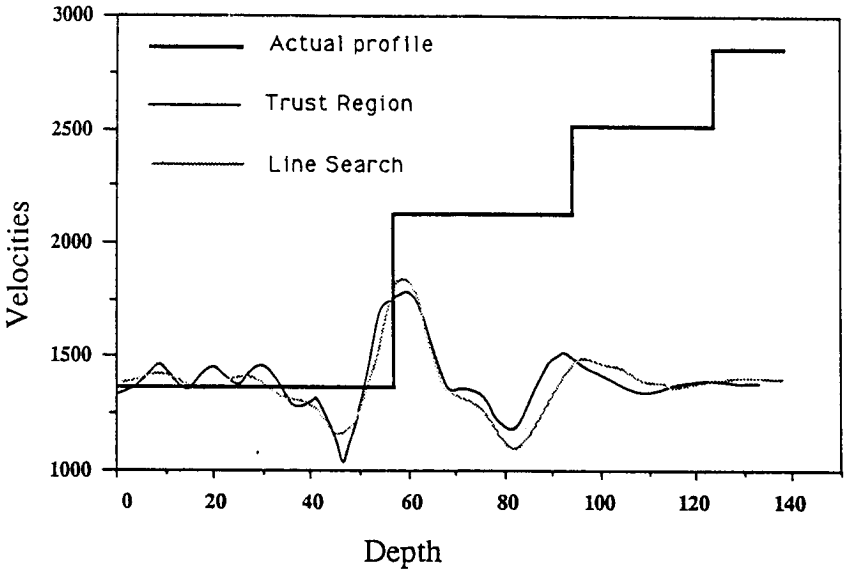

Figure 12.2 Reconstructed profiles for velocities, the initial profile being a constant one. The depth is in meters and the velocities in m/s. The model is a 2D horizontally stratified acoustic medium probed by a time-dependent acoustic source. A finite-element technique is used in order to compute 200 seismograms with about 10000 elements and 400 time steps. The objective function is the sum of the L_2 norms of the differences between the actual and computed seismograms. Two algorithms were used: first a Trust Region algorithm with the rank-one updating formula and a full SVD decomposition of the Hessian; second a standard Quasi-Newton algorithm with Line Search and the BFGS formula for computing the Hessian. Note that both algorithms yield profiles very similar but different from the actual profile; yet the objective function has actually been minimized, which indicates the existence of a *local* minimum.

From the viewpoint of local minima, the Trust Region algorithm is not superior to the Line Search algorithm; however the former required about twice less computations of the direct problem than the latter [24], [44]. This point is illustrated on Fig. 12.4, where the logarithm of the objective function is plotted as a function of an index reflecting the amount of computational work devoted to the modelisation.

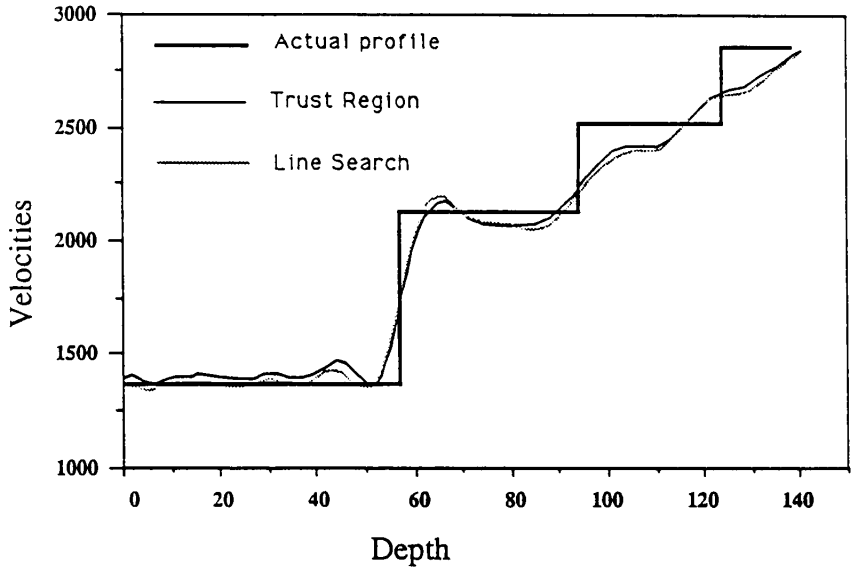

Figure 12.3 Reconstructed profiles for velocities, the initial profile being a ramp function. The model and the two algorithms are the same as in Fig. 12.2. Now the solution is satisfactorily reconstructed.

c. A Geophysical Inverse Problem: Full Acoustic Waveform Inversion

In a seismic experiment, a seismic source induces waves which, after propagation, reflections and refractions in the subsurface, are recorded on geophones or seismographs located at the surface of the earth or in wells. A seismic section for a surface experiment consists of traces recorded on aligned surface receivers identified by their distance to the source. Figure 12.5 illustrates the complexity of the data.

The aim of an inversion procedure in geophysics is to determine the subsurface model which will minimize an objective function built up with the misfit between the model seismic response and the field data. Thus, an optimization problem comes out, usually expressed with a least squares criterion but other norms can also be envisaged (L1 norm, for instance, when spiked reflectors are searched). The two features of full waveform inversion are then a full wave equation modelling and a minimization problem.

The modelling is commonly performed with a finite difference scheme, for which recent experiments begin to enable a 3D modelling of field experiments [45]. Thus, all the phenomena belonging to acous-

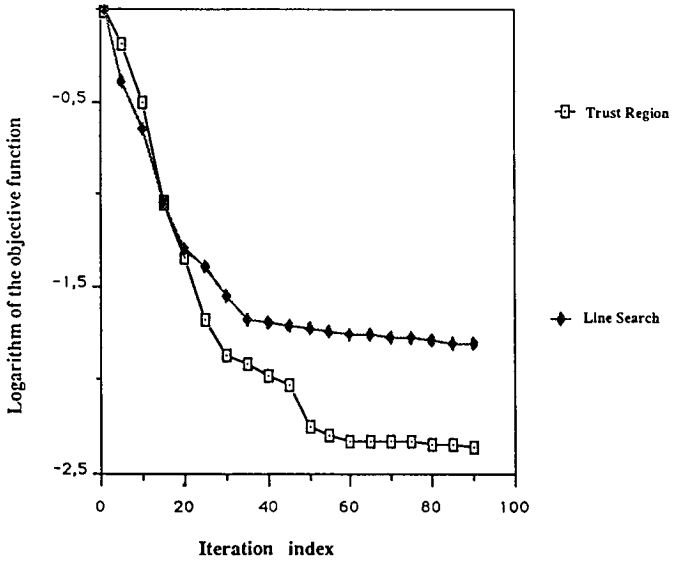

Figure 12.4 Logarithm of the objective function vs the iteration index. The model and the objective function are the same as in Fig. 12.2. The iteration index, defined as the number of function evaluations plus twice the number of gradient evaluations, reflects the amount of computational work devoted to the modelisation.

tic wave propagation such as refraction and multiples are taken into account in the modelling scheme. The gradient is computed using an adjoint state technique which requires two modellings: the modelling of the wave propagation in the assumed model and the modelling of the backward propagation of the residuals.

The seismic data provide two kinds of information expressed by two classes of parameters. The amplitude information is created by the reflection coefficients whereas the phase contains information about the velocity. When the subsurface is supposed to be a stratified acoustic medium, Kolb and Canadas [46] identified the velocities and the reflection coefficients as the least coupled parameters and, thus, as the best fitted parameters for inversion. Making the right choice of parameters can accelerate the convergence of the optimization method and help to avoid the local minima by improving the conditioning of the problem. Kolb and Canadas propose a continuation methodology to try to stabilize the inversion procedure. It has been shown that this stabilization is usually obtained when the low frequency of the velocity is known. The continuation method then consists of an increasing

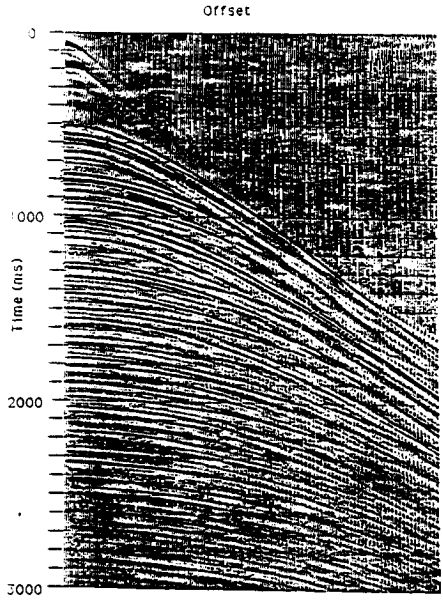

Figure 12.5 A typical set of seismograms, illustrating the complexity of the Geophysical data. Every seismograms is obtained by recording the reflected acoustical field as a function of time and for a given offset, i.e. at a given horizontal distance from the source.

frequency bandwidth inversion starting with a low frequency inversion performed on low pass-band filtered data.

The problem is also better posed when the number of parameters is low. A downward identification technique is then a way to improve the search for a global minimum.

However, the seismic data acquired on the surface lead only to the low frequency component of the velocity and to the high frequency of the impedances (the reflection coefficient is the derivative of the logarithm of the impedance). A question has been raised by Tarantola and Jannane [47] about the possible existence of a gap in the frequency content of the physical quantities which imposes the input of further information in order to obtain a complete impedance and velocity profile in the seismic frequency band. This complementary information is known as a priori information.

Richard et. al. [48] are proposing an Integrated Stratigraphic Interpretation tool based on inversion with a priori information. Such a tool constructs an acoustic impedance section which is supposed to fit the available impedance logs, to agree with the interpreter's knowledge

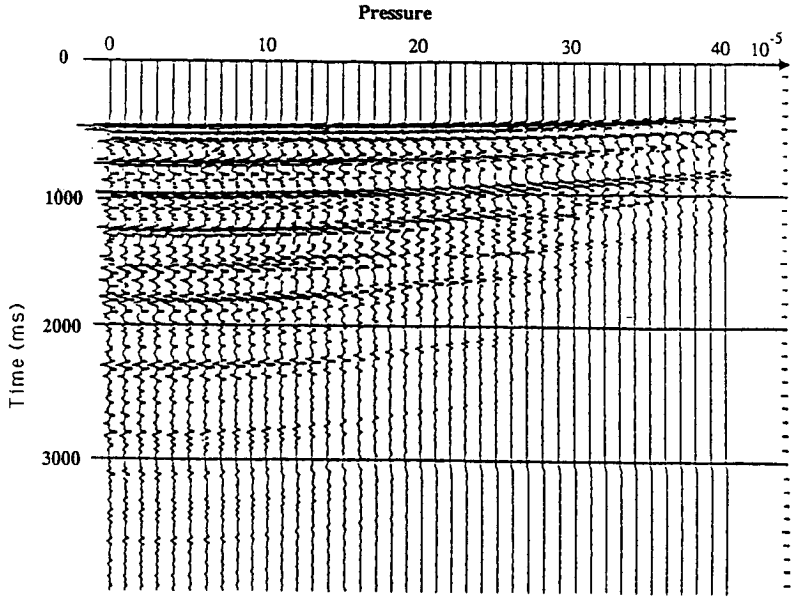

Figure 12.6 Seismic section in (τ, p) domain.

about stratigraphy and which is supposed to generate a synthetic section which is close enough, to the least squares sense, to the observed one. These constraints ensure the consistency of the interpretation.

Recently, Chapel, Kolb and Canadas [49] have developed a prestack inversion procedure for a horizontally layered medium which has the originality of decoupling the search for the velocity profile from the search for the impedances. This is achieved through a conversion of the seismic section to the (τ, p) domain (Fig. 12.6). The acoustic wave equation is then shattered into multiple 1-D wave equations which provide, after inversion, p-dependent reflection coefficients (Fig. 12.7). The optimization algorithm is a Conjugate Gradient method with a preconditioning which simulates the influence of the hessian and which takes into account the effects of the multiples in order to accelerate the convergence. Following ideas extracted from Symes' work [50], a coherency analysis of the demultiplied traces which have been freed of the source effects gives way to a velocity estimation. At each p value there corresponds an image of the subsurface. This redundancy can be used to identify lithologic parameters which are rather reliable when coming out of a Reflectivity Versus Angle analysis.

The inverse seismic problem for a stratified subsurface begins to

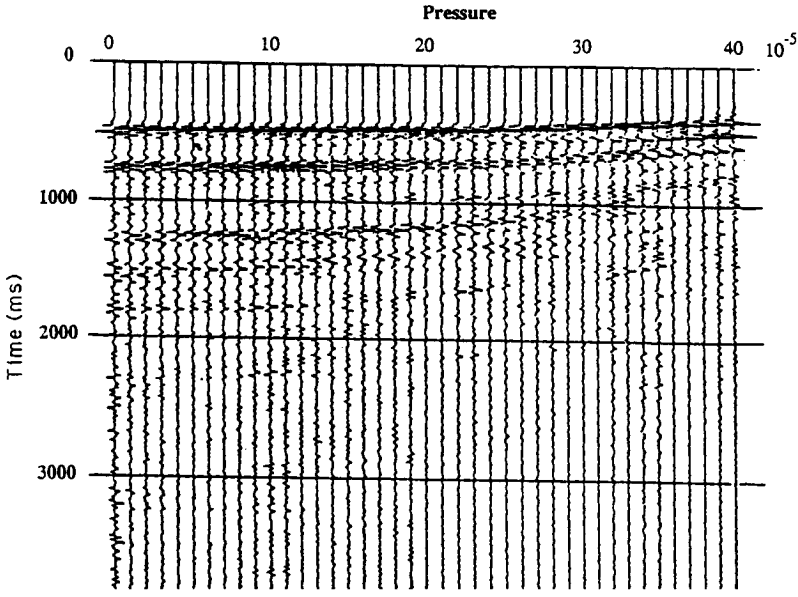

Figure 12.7 Inverted (t, p) seismic secton.

give significant results on field data. However, weak links remain in the methodology such as the determination of the source, which is partly solved in 1D inversion, and the definition of a robust criterion to terminate the optimization.

d. An Electromagnetic Inverse Problem: Full Wave Inversion of Logs

Dielectric logging is implemented with the help of an electromagnetic transmitter which generates an harmonic wave and creates induced voltages on a set of receivers. The whole device, called an electromagnetic tool, is pulled down into a borehole and measurements are performed. The frequency is generally of the order of some tens of MHz, so that the wave penetrates into the layers up to a distance of about ten meters. A dielectric log consists of voltage recorded at regularly spaced points along the well.

The propagation of the electromagnetic waves is governed mainly by the permittivity and conductivity of the surrounding medium. The inversion procedure must determine the permittivities and conductivities of the geological layers passed through. Various techniques have

been employed. In particular, when the medium is homogeneous, the direct model can be solved in a compact form so that a plain inversion procedure can be implemented: this is the so-called Equivalent Homogeneous Medium (E.H.M.) method. However, this naive technique gives bad results in the neighbourhood of an interface between two layers (Fig. 12.6), as one could a priori expect. A full wave analysis is required, and the inversion problem is formulated as the minimization of the gap between the model electromagnetic response and the field data. The objective function is usually a L_2 norm with possibly regularizing terms.

For the sake of simplicity one generally considers models with circular symmetry. An analysis with a Hankel transform is performed, which permits one to separate the vertical and the radial coordinates [51]. The gradient is computed with the help of an adjoint state derived from reciprocity considerations [3] [52], or equivalently from Rumsey's reaction concept [53].

The behavior of the iterative process depends much on the parameterization of the model. Indeed, when the medium is described by a large number of elementary slices with fixed interfaces, the numerical experiments showed that the related inversion process is plagued by a crowd of local minima [52]. Some authors working with an approximate model and using conjugate gradient algorithms [34] had to put additional constraints on the parameters to enforce convergence. On the other hand, when the model consists of a small number of thick layers, the parameters to be determined being now not only the permittivity and conductivity but also the positions of the interfaces, then the reconstruction process works quite satisfactorily and can even be accelerated by using a good first estimate [54].

12.6 Conclusion

Minimization algorithms have been the subject of much mathematical work and many families of algorithms are available in the mathematical literature. The first guideline to choose an algorithm is constituted by its theoretical properties: global convergence seems to be a minimal requirement, but the asymptotic speed of convergence is also of interest. We have presented here a review of the main families of algorithms with their mathematical properties. On the other hand for practical implementations the nature of the objective function to be minimized is of highest importance: if the computational price of

its evaluation is low, the most robust algorithm is the best. However, as often in industrial problems, the model code requires much computation time and thus must not be called too many times; it is then necessary to find a trade-off between robustness and convergence speed, which leads to sophisticated algorithms. Additional difficulties such as the existence of local minima can occur: the solution is to be found in the physics of the problem, which often gives the key of efficient implementations.

References

[1] Moré, J. J., B. S. Gabow, and K. E. Hillstrom, "Testing unconstrained optimization software," *ACM Trans. Math. Soft.*, **7**, 17–41, 1981.

[2] Lions, J. L., *Contrôle optimal des systèmes gouvernés par des équations aux dérivées partielles*, Dunod, Gauthier Villars, Paris, 1968.

[3] Roger, A., "Reciprocity theorem applied to the computation of functional derivatives," *Electromagnetics*, **2**, 69–83, 1982.

[4] Levy, A. V., and S. Gomez, "The tunneling algorithm for the global optimization problem of constrained functions," Technical Report, Universidad National de México, 1980.

[5] Rinnooy Kan, A. H. G., and G. T. Timmer, "Stochastic Global Optimization methods: Clustering methods," *Mathematical Programming*, **39**, 27–56, 1987.

[6] Kirkpatrick, S., C. D. Crelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, **220**, 671–680, 1983.

[7] Fiacco, A. V., and G. P. McCormick, *Non Linear Programming: Sequential Unconstrained Minimization Techinques*, John Wiley, New York, 1968.

[8] Sarkar, T. K., X. Yang, and E. Arvas, "A limited survey of various conjugate gradient methods for solving complex matrix equations arising in Electromagnetic Wave Interaction," *Wave Motion*, **10**, 527–546, 1988.

[9] Tijhuis, A. G., "Born-type reconstruction of material parameters of an inhomogeneous lossy dielectric slab from reflected field data," *Wave Motion,* **11**, 151-173, 1989.

[10] Kennett, B. L. N., and P. R. Williamson, "Subspace methods for large-scale non linear inversion," in *Mathematical Geophysics,* 139–154, Reidel Publishing Co., 1988.

[11] D'Aboville, O., "Etude et mise en oeuvre d'un parallelisme multitache en sismique numerique," Ph.D. dissertation, 1987.

[12] Hestesnes, M., and E. Stiefel, "Method of conjugate gradient for solving linear systems," *J. Res. Nat. Standards,* **49**, 409–436, 1952.

[13] Fletcher, R., and C. M. Reeves, "Function minimization by conjugate gradients," *Computer Journal,* **7**, 149–154, 1964.

[14] Polack, E., *Computational Methods in Optimization, a Unified Approach,* Academic Press, New York, 1971.

[15] Van den Berg, "Iterative computational techniques in scattering based on the integrated squared error criterion," *IEEE Trans. Antennas Propagat.,* **32**, 1063–1071, 1984.

[16] Roger, A., "Newton Kantorovich algorithm applied to an electromagnetic inverse problem," *IEEE Trans. Antennas Propagat.,* **29**, 232–238, 1981.

[17] Powell, M. J. D., "Convergence properties of a class of Minimization Algorithms," *Non Linear Programming,* **2**, Academic Press, New York, 1975.

[18] Broyden, C. G., "A class of methods for solving non linear simultaneous equations," *Math Comp.,* **19**, 577–593, 1965.

[19] Buckley, A., and A. Lenir, "QN-like variable storage Conjugate Gradient," *Mathematical Programming,* 1982.

[20] Sorensen, D. C., "Newton's method with a model trust region modification," *SIAM J. Numer. Anal.,* **19**, 409–426, 1982.

[21] Moré, J. J., "Recent developments in Algorithms and Software for trust Region methods," in *Mathematical Programming: the state of the art,* 258–287, Springer, 1982.

[22] Roger, A., P. Terpolilli, and O. Gosselin, "Trust region methods for seismic inverse problems," *World Scientific,* 93–103, Singapore, 1989.

[23] Shultz, G. A., R. B. Schnabel, and H. R. Byrd, "A family of Trust Region based algorithms for unconstrained minimization with strong global convergence properties," *Siam J. Numer. Anal.*, **22**, 47–67, 1985.

[24] Terpolilli, P., and A. Roger, "Trust Region strategies with Quasi-Newton schemes," in *New Methods in Optimization and their Industrial Uses, ISNM*, **87**, Birkhauser, 1989.

[25] Levenberg, K., "A method for the solution of certain non linear problems in least squares," *Quart. Appl. Math.*, **2**, 164–168, 1944.

[26] Marquardt, D. W., "An algorithm for least squares estimation of non linear parameters," *SIAM J. Appl. Math.*, **11**, 434–441, 1963.

[27] Zangwill, W. L., *Non Linear Programming: A Unified Approach*, Prentice Hall, 1969.

[28] Goldstein, A. A., *Constructive Real Analysis*, Harper, New York, 1967.

[29] Wolfe, P., "Convergence conditions for ascent methods," *S.I.A.M. Review*, **11**, 226–235, 1969.

[30] Powell, M. J. D., "Some global convergence properties of a variable metric algorithm for minimization without exact line searches," *Non Linear Programming, SIAM-AMS Proceedings 9*, Providence, R.I., 1976.

[31] Luenberger, D. G., *Introduction to Linear and Nonlinear Programming*, Addison Wesley, 1973.

[32] McCormick, G. P., and K. Ritter, "Alternative proofs of the convergence Properties of the Conjugate Gradient Method," *J. Optimization Theory and Appl.*, **13**, 497–515, 1974.

[33] Dennis, J. E., and J. J. Moré, "A characterization of superlinear convergence and its application to Quasi-Newton methods," *Mathematics of Computation*, **28**, 549–560, 1974.

[34] Galan-Malaga, H., D. Lesselier, and W. Tabbara, "Dipole approximations applied to the inverse problem in boreholes: a numerical study," *Wave Motion*, **11**, 137–150, 1989.

[35] Gill, P. E., W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press, London, 1981.

[36] Roger, A., and M. Baribaud, "Optimization and the time-dependent inverse problem," *Optique Hertzienne et Diélectriques Meeting,* 1989.

[37] Roger, A., "Convergence et Stabilité en Sismique inverse," in *Rapport Elf-Aquitaine n° 89-445,* 1989.

[38] Roger, A., D. Maystre, and M. Cadilhac, "On a problem of inverse scattering in Optics: the dielectric inhomogeneous medium," *J. Optics,* **9**, 83–90, 1978.

[39] Tijhuis, A., *Electromagnetic Inverse Profiling,* VNU Science Press, Utrecht, 1987.

[40] Hiebert, K. L., "An Evaluation of Mathematical Software That Solves Nonlinear Least Squares Problems," *ACM Trans. Mathematical Software,* **7**, n°1, 1–16, 1981.

[41] Gay, D. M., "Subroutines for Unconstrained Minimization using a model/Trust Region Approach," *ACM Trans. Mathematical Software,* **9**, n°4, 503–524, 1983.

[42] Montarnal, A., "Mise en oeuvre d'une batterie de tests pour algorithmes de minimisation sans contrainte," in *Elf-Aquitaine Report n° 87-752,* 1987.

[43] Hasselot, C., and V. Millet, "Confrontation de divers algorithmes d'optimisation," D.E.S.S., University of Pau, 1987.

[44] Gosselin, O., A. Roger, P. Terpolilli, and J. P. Penot, "Problèmes d'identification: un algorithme robuste et efficace," in *Congrès National d'Analyse Numérique,* 1988.

[45] Holberg, O., "Computational aspects of the choice of operator and sampling interval for numerical differentiation in large-scale simulation of wave phenomena," *Geophysical Prospecting,* **37**, 629–655, 1987.

[46] Kolb, P., and G. Canadas, "Least squares inversion of prestack data: simultaneous identification of density and velocity of stratified media," in *56th Annual SEG Meeting,* Houston, 1986.

[47] Jannane, H. N., "The wavelengths of the earth structures that we can resolve from seismic reflection data," *ETG Report,* 8, 2.1–2.18, 1987.

[48] Richard, V., "High resolution stratigraphic extrapolation: a 1–D inverse problem," *48th E.A.E.G. meeting,* Ostend, 1986.

[49] Chapel, F., P. Kolb, and G. Canadas, "CMP non linear inversion in (τ, p) domain," *59th Annual SEG Meeting,* Houston, 1989.

[50] Symes, W. W., "Velocity inversion by coherency optimization," *58th Annual SEG Meeting,* Anaheim, 1988.

[51] Howard, A. Q., Jr., "Induction logging for vertical structures in the presence of a borehole fluid," *Geophys.,* **46**, n°1, 68–75, 1981.

[52] Roger, A., E. Toro, and D. Maystre, *Contract Schlumberger-Aix Marseille III,* 1984.

[53] Rumsey, V. H., "Reaction concept in Electromagnetic Theory," *Phys. Rev.,* **94**, n°6, 1483–1491, 1954.

[54] Maystre, D., A. Roger, and E. Toro, "Inverse scattering on electromagnetic measurements in a stratified medium, *Revue Phys. Appl.,* **20**, 815–821, 1985.