HAL
open science

Full Waveform Inversion and the truncated Newton method

Ludovic Métivier, Romain Brossier, Jean Virieux, Stéphane Operto

To cite this version:

Ludovic Métivier, Romain Brossier, Jean Virieux, Stéphane Operto. Full Waveform Inversion and the truncated Newton method. SIAM Journal on Scientific Computing, 2013, 35 (2), pp.B401-B437. 10.1137/120877854 . hal-00763702

HAL Id: hal-00763702

https://hal.science/hal-00763702

Submitted on 11 Dec 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

FULL WAVEFORM INVERSION AND THE TRUNCATED NEWTON METHOD

L. MÉTIVIER*, R. BROSSIER* , J. VIRIEUX* , AND S. OPERTO†

Abstract. Full Waveform Inversion (FWI) is a powerful method for reconstructing subsurface parameters from local measurements of the seismic wavefield. This method consists in minimizing a distance between predicted and recorded data. The predicted data is computed as the solution of a wave propagation problem. Conventional numerical methods for the resolution of FWI problems are gradient-based methods, such as the preconditioned steepest-descent, or more recently the l-BFGS quasi-Newton algorithm. In this study, we investigate the interest of applying a truncated Newton method to FWI. The inverse Hessian operator plays a crucial role in the parameter reconstruction. The truncated Newton method allows one to better account for this operator. This method is based on the computation of the Newton descent direction by solving the corresponding linear system through an iterative procedure such as the conjugate gradient method. The large-scale nature of FWI problems requires however to carefully implement this method to avoid prohibitive computational costs. First, this requires to work in a matrix-free formalism, and the capability of computing efficiently Hessian-vector products. To this purpose, we propose general second-order adjoint state formulas. Second, special attention must be payed to define the stopping criterion for the inner linear iterations associated with the computation of the Newton descent direction. We propose several possibilities and establish a theoretical link between the Steihaug-Toint method, based on trust-regions, and the Eisenstat stopping criterion, designed for method globalized by linesearch. We investigate the application of the truncated Newton method to two test cases: the first is a standard test case in seismic imaging based on the Marmousi II model. The second one is inspired by a near-surface imaging problem for the reconstruction of high velocity structures. In the latter case, we demonstrate that the presence of large amplitude multi-scattered waves prevents standard methods from converging while the truncated Newton method provides more reliable results.

Key words. Seismic imaging, Full Waveform Inversion, numerical optimization, large-scale inverse problems

AMS subject classifications. 86-08, 35R30, 49M15, 90C06

1. Introduction. Many imaging methods have been developed during the last century to delineate the Earth structure at various scales. Most of these methods rely on the interpretation of seismic waves. These waves travel inside the Earth and collect information about the subsurface properties. Extracting this information from a locally recorded wavefield is a difficult inverse problem.

1.1. Conventional methods. A standard method to recover quantitative information on the Earth subsurface is based on travel-time tomography. From the measurement of the distance between the sources and the receivers, and the recording of wave first-arrival times, an estimation of the wave velocity field can be computed. This method is known as first arrival travel-time tomography (FATT) [50]. The spatial resolution of the estimation provided by this method is however limited. In addition it requires large offset acquisition systems to record waves that propagate deeply in the Earth[1].

A two-step workflow has been designed for reflection wave arrivals [35] to improve the resolution of images of the subsurface. The first step consists in estimating a smooth background wave velocity distribution through the reflection travel time tomography method, based on the exploitation of the travel-times of the reflected

*ISTerre, Joseph Fourier University, Grenoble

†Géoazur, Nice, France.

[1]In the geophysicists vocabulary, an offset is the horizontal distance between a source and a receiver.

waves. The second step is known as the migration step, and is based on the concept of exploding surfaces introduced by Claerbout [14, 15]. Using the background wave velocity, a map of the subsurface reflectivity can be derived by amplitude stack of the seismic data. Refocusing the energy of the recorded waves is achieved at the subsurface main discontinuities.

The standard method thus consists in a two-step workflow.

- First, estimate a smooth background wave velocity field from reflection tomography
- Second, "migrate" the reflected wavefield using the estimated wave velocity field built at the previous step to refocus the energy of the recorded waves to the main subsurface reflectors.

This method has proven useful in numerous cases, for which the subsurface geometry keeps simple. However, the efficiency of this method relies on a sufficiently accurate estimation of the background wave velocity field. In presence of complex structures such as salt domes, subbasalt targets, thrustbelts, and foothills, the quality of the velocity estimation provided by the reflection tomography method is degraded. In this case, the migration method is unable to focus correctly the energy of the seismic waves, which results in a poor image of the subsurface.

Moreover, this two-step workflow only gives access to the geometrical structure of the subsurface. However, a *quantitative* high resolution estimation of the subsurface physical parameters such as the bulk and shear wave velocities, or the density, is far more useful, as smaller scale information such as petrophysical attributes can be inferred from such an information.

The Full Waveform Inversion (FWI) method has been designed to overcome these difficulties [34, 68].

1.2. The FWI method. The FWI method consists in improving iteratively a given initial subsurface model. This iterative optimization relies on the modeling of seismic waves through the forward problem, which offers the possibility of accounting simultaneously for waves amplitude and travel-times.

In this study, we use the following frequency-domain formulation:

$$S(p(x),\omega)u(x,\omega) = \varphi(x,\omega), \tag{1.1}$$

where

- $x \in \mathbb{R}^l$ $(l = 1, 2, 3)$ is the space variable;
- $p(x) \in \mathcal{M} \subset \left(L^2(\mathbb{R}^l)\right)^m$ denotes the subsurface parameters;
- $\omega \in \mathbb{C}$ is the frequency;
- $u(x,\omega) \in \mathcal{W} \subset \left(L^2(\mathbb{C}^{l+1}\right)$ is the complex-valued seismic wavefield;
- $\varphi(x,\omega)$ is a source term;
- $S(p(x),\omega)$ is a partial differential operator related to the wave equation (from the acoustic case to the visco-elastic anisotropic dynamics).

Note that $m \in \mathbb{N}$ denotes the number of parameter classes that are considered. For the sake of simplicity, we assume that the wavefield has only one component. The extension of our results to a multi-component wavefield is straightforward.

The FWI method consists in minimizing over the parameter space a distance between the data predicted by the forward problem and the recorded data. We formulate it as the minimization problem

$$\min_p f(p) = \frac{1}{2}\sum_{s=1}^{N_s}\sum_{q=1}^{N_\omega} \|R_s\left(u_s(p,\omega_q) - d_s(\omega_q)\right)\|^2, \tag{1.2}$$

where

- N_s is the number of sources;
- N_ω is the number of frequencies;
- $u_s(p, \omega_q)$ is the solution of the forward problem (1.1) for the source term $\varphi_s(x, \omega_q)$, the parameters $p(x)$ and the frequency ω_q;
- R_s is a mapping of the wavefield to the receiver locations;
- $d_s(\omega_q)$ is the data set associated to the source $\varphi_s(x, \omega_q)$ and the frequency ω_q.

The definition of the distance between the data is related to the definition of the norm in the data space. This is actually a crucial issue for FWI. For numerical convenience, the L^2 norm is, however, often chosen. More general L^p norm could be also selected [69]. The L^1 norm is, for instance, a good choice when high-amplitude noise (outliers) corrupts the data [11]. An emphasis on the interpretation of the phases rather than of the amplitudes can also be obtained by using appropriate misfit functions: see for instance the time-frequency misfit used in waveform inversion for a seismology application [20] or the phase-only misfit function [2] for a seismic exploration application.

The formalism of the FWI method has been introduced by Lailly [33] and Tarantola [68], based on a time domain discretization of the wave equation. Its first application to 2D synthetic data in the acoustic approximation was performed by Gauthier et al. [25]. Later on, a hierarchical frequency domain approach has been introduced by Pratt for cross-hole tomography [59, 57]. During the past ten years, the simultaneous advances in acquisition systems (development of wide-azimuth seismic surveys for instance which allows to record larger offset data) and high performance computing facilities have made possible the successful application of FWI to realistic 2D and 3D cases in the acoustic approximation (see for instance [54, 60], or [43, 44] for the 2D acoustic impedance reconstruction from well seismic data), as well as in the elastic approximation [11].

Because its formalism is fairly generic and does not rely on particular scale-dependent assumptions, FWI can be used in various environments and for various targets, for instance

- at the laboratory scale [9, 10];
- at the near-surface scale (500 m depth) in application to localization of water resources [16];
- at the exploration scale (10 km depth) in application to localization of oil and gas, storage or reservoir monitoring [24, 29, 51, 55];
- at the lithospheric scale (100 km depth), in application to fault slip reconstruction and earthquake interpretation [8, 7, 52, 21]

Imaging at very shallow near-surface scale (1 m depth) are also currently investigated [62]. The reconstruction of deep Earth structure at the global scale (10000 km scale) could also be considered.

The advantages of FWI over standard methods can be summarized as follows:

- the method provides a dramatic improvement of the spatial resolution of the estimation [1]
- quantitative images of the subsurface are computed whereas the migration

[1] A theoretical limit for the resolution in the best case has been established by Wu and Toksoz [71] to $\lambda/2$ where λ is the local wavelength. The possibility of reaching this bound mainly depends on the acquisition system geometry. A method for computing more realistic resolution measurements is proposed by Fichtner [23].

step in the standard workflow only provides qualitative structures;
- the formalism allows to consider a wide variety of imaging problems, and also multi-parameter estimation.

FWI has, however, a significant drawback. The problem (1.2) is a large-scale ill-posed and strongly nonlinear minimization problem. Indeed, the forward problem is linear with respect to the wavefield but nonlinear with respect to the subsurface parameters. As this problem is solved using local optimization methods, the starting model which is used should therefore be accurate enough to converge to the global minimum. If not, only a local minimum is found. In the context of wave velocity estimation by FWI, this phenomenon is interpreted physically as the cycle skipping problem: the kinematic information contained in the starting model is not valid and the recorded data is accounted for up to one or several phases. Numerous methods are designed to mitigate this difficulty. It is possible, for instance, to use a hierarchical interpretation of the data, from low frequency to high frequency in the frequency domain, or from short time windows to large time windows in the time domain. Changes in the definition of the misfit function can also be performed, as already mentioned, to increase the importance of phase matching rather than amplitude matching [20]. Indeed, this also lessens the nonlinearity of the problem. FWI can thus be seen as a two-step workflow, the first one consisting in computing an accurate enough starting model using for instance tomography methods. For an overview of FWI and its numerous applications, the reader is referred to the survey of Virieux and Operto [70].

1.3. Numerical resolution. Nonlinear minimization problems are usually solved using Newton-based methods. The Newton method consists in building a sequence p_k from an initial guess p_0, using the recurrence

$$p_{k+1} = p_k + \gamma_k \Delta p_k, \tag{1.3}$$

where γ_k is a scalar parameter associated with a globalization method (linesearch or trust-region) and the increment Δp_k satisfies

$$H(p_k)\Delta p_k = -\nabla f(p_k), \tag{1.4}$$

where the Hessian operator $\nabla^2 f(p_k)$ is denoted by $H(p_k)$

However, in the large-scale context[1] of problem (1.2), the explicit construction of the Hessian operator or of its inverse is beyond the present-day computational capability. As a consequence, quasi-Newton methods, or more generally preconditioned gradient-based method are rather used. The inverse Hessian $H(p_k)^{-1}$ is thus approximated by a matrix Q_k, and the computation of the descent direction Δp_k by the formula (1.4) is replaced by

$$\Delta p_k = -Q_k \nabla f(p_k). \tag{1.5}$$

The simplest choice for Q_k is the identity matrix, which corresponds to the steepest-descent method. This method is known to converge globally, but possibly very slowly. A better choice for Q_k is the l-BFGS approximation, proposed by Nocedal [12]. This method approximates the influence of the inverse Hessian $H(p_k)^{-1}$ by finite differences of l previous values of the gradient

$$\nabla f(p_{k-l+1}), \ldots, \nabla f(p_k). \tag{1.6}$$

[1]For standard FWI applications, the number of reconstructed parameters can go from 10 thousand to 1 million in 2D, and from 10 million to 10 billion in 3D.

The l-BFGS method has proven very useful in numerous large-scale applications.

For geophysicists, the computation of the descent direction Δp_k using the Newton formula (1.4) amounts to filter the gradient by the inverse Hessian operator to produce a better update of the estimate p_k at each iteration of the minimization loop. Pratt et al. [58] provide an accurate analysis of the importance of accounting for the Hessian operator in the inversion process. From the definition of $f(p)$, the Hessian operator $H(p)$ can be written as

$$H(p) = B(p) + C(p), \tag{1.7}$$

where

$$\begin{cases} B(p) = \mathcal{R}\left(\sum_{s=1}^{N_s}\sum_{q=1}^{N_\omega} J_s^\dagger(p,\omega_q) R_s^\dagger R_s J_s(p,\omega_q)\right), \\ C(p) = \mathcal{R}\left(\sum_{s=1}^{N_s}\sum_{q=1}^{N_\omega} R_s^\dagger \left(R_s u_s(p,\omega_q) - d_s\right) \partial_{pp} u_s(p,\omega_q)\right). \end{cases} \tag{1.8}$$

In this expression, $J_s(p,\omega_q) = \partial_p u_s(p,\omega_q)$ is the Jacobian matrix of the wavefield $u_s(p,\omega_q)$, $\partial_{pp} u_s(p,\omega_q)$ is the second-order derivatives of the wavefield $u_s(p,\omega)$, the symbol $\dagger$ denotes the conjugate transpose operator, and $\mathcal{R}$ denotes the real part operator. The matrix $B(p)$ is known as the Gauss-Newton approximation of the Hessian operator when the expression $C(p)$ is neglected.

Consider the simple case $N_s = 1$, $N_\omega = 1$. Assuming the system is discretized, denoting by p_i the discretized parameters, N_r the total number of receivers, and x_r their locations, a coefficient ik of the matrix $B(p)$ is

$$B(p)_{ik} = \mathcal{R}\left(\sum_{r=1}^{N_r} \partial_{p_i} u(x_r,\omega) \partial_{p_k} u(x_r,\omega)\right), \tag{1.9}$$

The expression (1.9) shows that B_{ik} is the zero-lag correlation of the first derivatives of the wavefield $u(p)$ recorded at the receivers locations, with respect to the couple of parameters (p_i, p_k). As a consequence, B_{ik} decreases with the distance which separates the parameters p_i and p_k and reaches its maximum for the autocorrelation of the two derivated wavefields ($i = k$). In the high-frequency approximation, the zero-lag correlation of the derivatives of the wavefield with respect to two different parameters would be zero and $B(p)$ would be diagonal. However, the frequency content of the seismic data is limited and the matrix $B(p)$ is banded. In addition , the amplitudes of the first-order derivatives wavefields $\partial_{p_i} u$ and $\partial_{p_k} u$ decrease for parameters which have less influence on the wavefield. Therefore, filtering the gradient $\nabla f(p)$ with the matrix $B(p)^{-1}$ acts as a refocusing filter. The amplitude of poorly illuminated parameters is compensated in the model update given by the opposite of the gradient of the misfit function.

In the same way, in discrete form, a coefficient ik of $C(p)$ is given by

$$C(p)_{ik} = \mathcal{R}\left(\sum_{r=1}^{N_r} \left(u(x_r,\omega) - d(x_r,\omega)\right) \partial_{p_i p_k} u(x_r,\omega)\right). \tag{1.10}$$

The expression (1.10) shows that C_{ik} is the zero-lag correlation of the differences between the predicted and the recorded data at the receivers (also named residuals)

and the second-order derivatives of the wavefield recorded at the receivers, with respect to the parameters p_i and p_k. This second-order wavefield is also called double scattered wavefield: it represents a recorded signal that has been scattered twice, at the locations of the parameter p_i and p_k. In certain configurations, for instance in the presence of high-amplitude velocity contrasts, the amplitude of the double scattered wavefield (and more generally of the multi-scattered wavefield) is not negligible. This double scattered wavefield generates strong artifacts on the gradient $\nabla f(p_k)$ for the optimization procedure, which only accounts for single scattered waves. Therefore, applying the matrix C_{ik}^{-1} to the gradient allows to compensate the artifacts generated by double scattered waves for the inversion. Let us remind that multiple scattering is included in the forward modeling.

1.4. The truncated Newton method. The analysis carried out by Pratt et al. [58], which we have summarized, underlines the crucial importance of the Hessian operator in the resolution of the FWI problem (1.2), and naturally raises the following question: is it possible to better account for the action of the inverse Hessian operator? This question is at the heart of this study. One possible way to better account for the Hessian influence is solving the problem (1.2) with the truncated Newton method. Contrary to quasi-Newton algorithms, this method is not based upon an approximation of the inverse Hessian operator. Instead, the descent direction Δp_k is computed as the solution of (1.4) using the conjugate gradient algorithm [64]. A global survey on the truncated Newton method has been proposed by Nash [47]. This method is designed for large-scale nonlinear minimization problem. In terms of computational efficiency and convergence speed, some comparisons between the l-BFGS method and the truncated Newton method have been proposed [63]. The performances of the two algorithms are comparable, depending on the test cases. None of the two methods outperform the other[1]. Nonetheless, in the FWI context, given the importance of the Hessian operator, the truncated Newton method may provide better results than the l-BFGS method.

The study we propose aims at giving some answers to this question. In Section 2, we discuss how the truncated Newton method can be implemented at a reasonable computation cost, with the purpose of being competitive with the l-BFGS method. Second-order adjoint formulas yield an efficient algorithm for the computation of Hessian vector products. A matrix-free implementation of the truncated Newton method is thus possible. In Section 3, we give some details about this implementation, with a particular emphasis on the conjugate gradient stopping criterion for the resolution of the linear system (1.4). In Section 4, we present two synthetic FWI applications. The first one is the standard benchmark test MARMOUSI II. The second one is a test case related to the imaging of highly contrasted structures in the near-surface. For both of these test cases, we compare the efficiency of the l-BFGS algorithm and the truncated Newton method. A conclusion and some perspectives are given in the last section.

2. The adjoint state method. The l-BFGS method only requires the capability of computing the gradient of the misfit function $f(p)$. Using the notations that

[1] A mixed algorithm has even been proposed by Nocedal [46] so as to combine the advantages of the two methods. The idea is to alternate between sequences of l-BFGS iterations and sequences of truncated Newton iterations. The l-BFGS approximation computed during the l-BFGS iterations is used to precondition the linear system (1.4) during the truncated Newton iterations.

have been introduced, $\nabla f(p) \in \mathcal{M}$ is expressed as

$$\nabla f(p) = \mathcal{R}\left(\sum_{s=1}^{N_s}\sum_{q=1}^{N_\omega} J_s^\dagger(p,\omega_q) R_s^\dagger \left(R_s u_s(p,\omega_q) - d_s(\omega_q)\right)\right). \tag{2.1}$$

As for the computation of the Hessian operator, the size of the considered applications prevents from directly evaluating and storing the Jacobian matrices $J_s(p,\omega)$.

The truncated Newton method requires the computation of the gradient $\nabla f(p)$ but also the resolution of the linear system (1.4). This can be achieved without forming explicitly the Hessian operator using a matrix-free iterative solver, such as the conjugate gradient. Nonetheless, this requires the capability of computing Hessian-vector products $H(p)v$ for any v in the parameter space $\mathcal{M}$.

In this section, we present how first-order and second-order adjoint state methods yield efficient algorithms for computing the gradient of the misfit function and Hessian-vector products. For the sake of clarity, we may suppose that $N_s = 1$ and $N_\omega = 1$, and drop the corresponding index in this section. Formula for the general context are straightforwardly derived from this particular case.

2.1. First-order adjoint state method and gradient computation. The adjoint state method has been introduced in the late 60s by Lions, in the context of optimal control theory [39]. In the 70s, this method has been applied to parameter identification by Chavent [13], and to weather forecasting by Le Dimet and Talagrand [37]. A survey of its use in seismic imaging has been proposed by Plessix [53]. We briefly recall the principle of this method.

The FWI problem can be formulated as the following constrained optimization problem:

$$\min_p \frac{1}{2}||Ru - d||^2, \quad \textit{subject to} \;\; S(p)u = \varphi. \tag{2.2}$$

The Lagrangian function associated with the problem (2.2) is

$$L(p,u,\lambda) = \frac{1}{2}||Ru - d||^2 + \mathcal{R}\left(S(p)u - \varphi, \lambda\right)_{\mathcal{W}}, \tag{2.3}$$

where $(.,.)_{\mathcal{W}}$ is the scalar product in the wavefield space $\mathcal{W}$ and $\lambda \in \mathcal{W}$ is the adjoint variable. Let $\overline{u}(p)$ denote the solution of the forward problem (1.1). Then

$$L(p,\overline{u}(p),\lambda) = f(p). \tag{2.4}$$

Therefore, we have

$$\partial_p(L(p,\overline{u}(p),\lambda)) = \nabla f(p), \tag{2.5}$$

which yields

$$\mathcal{R}(\partial_p S(p)\overline{u}(p),\lambda) + \partial_u L(p,\overline{u}(p),\lambda)\partial_p \overline{u}(p) = \nabla f(p). \tag{2.6}$$

We define the adjoint state $\overline{\lambda}(p)$ such that

$$\partial_u L(p,\overline{u}(p),\overline{\lambda}(p)) = 0, \tag{2.7}$$

which is equivalent to

$$S(p)^\dagger \overline{\lambda} = R^T(d - R\overline{u}(p)). \tag{2.8}$$

We thus obtain the gradient formula

$$\nabla f(p) = \mathcal{R}(\partial_p S(p)\overline{u}(p), \overline{\lambda}(p)). \tag{2.9}$$

The adjoint state method for the computation of $\nabla f(p)$ thus consists in solving sequentially the equations (1.1) and (2.8), then computing the gradient $\nabla f(p)$ using the equation (2.9).

The overall computation cost of the operation is driven by the resolution of two wave propagation problems: one associated with the forward problem (1.1), one associated with the adjoint problem (2.8). The method does not require to form explicitly the Jacobian matrix. This yields an efficient algorithm for the computation of the gradient. We now investigate how this method can be extended to the computation of Hessian-vector products, in the perspective of using the truncated Newton algorithm for the resolution of the FWI problem (1.2).

2.2. Second-order adjoint state method.

2.2.1. Hessian vector product computation. The computation of Hessian-vector products through second-order adjoint methods is a topic that has already been investigated in the field of data assimilation and weather forecasting [36]. However, the control variable in data assimilation is an initial condition for the system, whereas in seismic imaging, the control variable is a coefficient of the Partial Differential Equation (PDE) that describes the system. A formula for the computation of Hessian-vector products have been given by Pratt [58] in the seismic imaging context, for the Gauss Newton approximation in the discrete frequency domain. Fichtner [22] proposes more general formulas for the computation of Hessian kernels. Epanomeritakis [18] gives the formulas corresponding to the elastic case.

We propose here a general framework in the frequency domain for deriving these formulas, with no assumption on the discretization and the kind of partial differential equations that are used for the wave propagation description. The method can be straightforwardly adapted to the time-domain formulation by adding proper initial and final conditions, and boundary conditions. Based on the definition of a new Lagrangian function, we derive an algorithm to compute, for given $(v, p) \in \mathcal{M}^2$, the matrix-vector product $H(p)v$.

$$g_v(p) = (\nabla f(p), v)_{\mathcal{M}}, \tag{2.10}$$

where $(.,.)_{\mathcal{M}}$ denotes the scalar product on the parameter space $\mathcal{M}$. By definition, the gradient of the functional $g_v(p)$ is

$$\nabla g_v(p) = H(p)v. \tag{2.11}$$

Therefore, applying the adjoint state method to the functional $g_v(p)$ to compute $\nabla g_v(p)$ gives the Hessian-vector product $H(p)v$. Recalling the definition of $\nabla f(p)$ given in equation (2.1), $g_v(p)$ can be rewritten as

$$g_v(p) = (R^\dagger(Ru(p) - d), J(p)v)_{\mathcal{W}}. \tag{2.12}$$

Deriving the forward problem with respect to the parameters p_j in the directions v_j yields

$$\left(\partial_{p_j} S(p).v_j\right) u + S(p)\left(\partial_{p_j} u.v_j\right) = 0, \;\; j = 1, \ldots, m. \tag{2.13}$$

Summing on j gives

$$\sum_{j=1}^{m} \left(\partial_{p_j} S(p).v_j\right) u + \sum_{j=1}^{m} S(p) \left(\partial_{p_j} u.v_j\right) = 0, \tag{2.14}$$

which is equivalent to

$$S(p)(J(p)v) = \Phi_v(p,u), \tag{2.15}$$

where

$$\Phi_v(u) = -\sum_{j=1}^{m} (\partial_{p_j} S(p).v_j)u. \tag{2.16}$$

The expression $J(p)v \in \mathcal{W}$ is denoted by $\overline{\alpha}_v(p)$ and is the solution of the forward problem (1.1) for the source term $\Phi_v(p,u)$. Thus, we may consider the constrained minimization problem

$$\min_p (R^\dagger(Ru-d),\alpha)_{\mathcal{W}}, \quad \textit{subject to} \;\; S(p)u = \varphi, \;\; S(p)\alpha = \Phi_v. \tag{2.17}$$

Introducing the adjoint variables $(\lambda,\mu) \in \mathcal{W}^2$, the Lagrangian function associated with this problem can be written as

$$L_v(p,u,\alpha,\lambda,\mu) = (R^\dagger(Ru-d),\alpha)_{\mathcal{W}} + \mathcal{R}(S(p)u-\varphi,\mu)_{\mathcal{W}} + \mathcal{R}(S(p)\alpha-\Phi_v,\lambda)_{\mathcal{W}}. \tag{2.18}$$

Thus, for $\overline{u}(p)$ solution of (1.1) and $\overline{\alpha}_v(p)$ solution of (2.15), we have

$$L_v(p,\overline{u}(p),\overline{\alpha}_v(p),\lambda,\mu) = g_v(p), \tag{2.19}$$

and

$$\partial_p L_v(p,\overline{u}(p),\overline{\alpha}_v(p),\lambda,\mu) = \nabla g_v(p). \tag{2.20}$$

Therefore, we have

$$\begin{aligned}\nabla g_v(p) = \partial_u L_v(p,\overline{u}(p),\overline{\alpha}_v(p),\lambda,\mu)\partial_p\overline{u}(p) + \partial_\alpha L_v(p,\overline{u}(p),\overline{\alpha}_v(p),\lambda,\mu)\partial_p\overline{\alpha}(p) + \\ \mathcal{R}\left((\partial_p S(p)\overline{u}(p),\mu)_{\mathcal{W}} + (\partial_p S(p)\overline{\alpha}_v(p),\lambda)_{\mathcal{W}} + \sum_{j=1}^{m} v_j \left(\left(\partial_{p_j}\partial_p S(p)\right)\overline{u}(p),\lambda\right)\right)\end{aligned} \tag{2.21}$$

Define the adjoint states $\overline{\lambda}(p)$ and $\overline{\mu}(p)$ such that

$$\begin{cases} \partial_u L_v(p,\overline{u}(p),\overline{\alpha}_v(p),\overline{\lambda}(p),\overline{\mu}(p)) = 0 \\ \partial_\alpha L_v(p,\overline{u}(p),\overline{\alpha}_v(p),\overline{\lambda}(p),\overline{\mu}(p)) = 0, \end{cases} \tag{2.22}$$

which is equivalent to

$$S(p)^\dagger\overline{\mu} = -R^\dagger R\overline{\alpha}_v - \sum_{j=1}^{m} (\partial_{p_j} S(p).v_j)^\dagger\overline{\lambda} \tag{2.23}$$

$$S(p)^\dagger\overline{\lambda} = R^T(d - R\overline{u}(p)). \tag{2.24}$$

We end up with the Hessian-vector product formula

$$H(p)v = \mathcal{R}\left((\partial_p S(p)\overline{u}(p), \overline{\mu}(p))_{\mathcal{W}} + (\partial_p S(p)\overline{\alpha}_v(p), \overline{\lambda}(p))_{\mathcal{W}} + \sum_{j=1}^{m} v_j \left(\left(\partial_{p_j}\partial_p S(p)\right) \overline{u}(p), \overline{\lambda}(p) \right) \right) \tag{2.25}$$

Thererfore, for a given $v \in \mathcal{M}$, and a given subsurface model $p \in \mathcal{M}$, the computation of the Hessian-vector product $H(p)v$ requires to solve four wave propagation problems: two forward problems for the computation of $\overline{u}(p)$ and $\overline{\alpha}_v(p)$, two adjoint problems for the computation of $\overline{\lambda}(p)$ and $\overline{\mu}(p)$. During the process, the Hessian matrix $H(p)$ is never computed explicitly. Note the identity between the adjoint state $\overline{\lambda}(p)$ derived here and the one which is derived for the computation of the misfit gradient through the first-order adjoint state formula.

2.2.2. The Gauss-Newton approximation. In the context of the Gauss-Newton approximation, for given $(p, v) \in \mathcal{M}^2$, we aim at computing the matrix vector product

$$B(p)v = J(p)^{\dagger} R^T R J(p) v. \tag{2.26}$$

The second-order part $C(p)$ of the Hessian operator is neglected. We consider the constrained minimization problem

$$\min_p g_w(p) = (u(p), w)_{\mathcal{W}} \quad \textit{subject to} \quad S(p)u = \varphi, \tag{2.27}$$

for an arbitrary $w \in \mathcal{W}$. Note that

$$\nabla g_w(p) = J(p)^{\dagger} w. \tag{2.28}$$

The Lagrangian function associated with this problem is

$$L_w(p, u, \nu) = (u, w) + \mathcal{R}(S(p)u - \varphi, \nu), \tag{2.29}$$

where $\nu \in \mathcal{W}$ is a new adjoint variable. Following the first-order adjoint state method, we have

$$\nabla g_w(p) = \mathcal{R}(\partial_p S(p)\overline{u}(p), \overline{\nu}(p)), \tag{2.30}$$

where

$$\partial_u L_w(p, \overline{u}(p), \overline{\nu}(p)) = 0, \tag{2.31}$$

which is equivalent to

$$S(p)^T \overline{\nu} = -w. \tag{2.32}$$

The formula (2.30) gives $J^{\dagger}(p)w$ for any $w \in \mathcal{M}$. Replacing w by $R^T R J(p) v$ in equation (2.32) thus yields the formula for the computation of $B(p)v$. Define $\overline{\xi}(p) \in \mathcal{W}$ such that

$$S(p)^T \overline{\xi} = -R^T R \overline{\alpha}_v(p), \tag{2.33}$$

where $\overline{\alpha}_v(p)$ is defined by (2.15), we have

$$B(p)v = \mathcal{R}\left(\partial_p S(p)\overline{u}(p), \overline{\xi}(p)\right). \tag{2.34}$$

As a consequence, for a given $v \in \mathcal{M}$ and a given subsurface model $p \in \mathcal{M}$, the computation of the matrix-vector product $B(p)v$ amounts to solving three wave propagation problems: two forward problems for the computation of $\overline{u}(p)$ and $\overline{\alpha}_v(p)$ and one adjoint problem for the computation of $\overline{\xi}(p)$. Note that

- the computation of $\overline{\xi}(p)$ amounts to setting $\overline{\lambda}(p)$ to 0 in the equation (2.23);
- the computation of $B(p)v$ amounts to setting $\overline{\lambda}(p)$ to 0 in the equation (2.25).

In view of these results, it could appear that the Gauss-Newton approximation requires less computational efforts, as only one adjoint problem has to be solved instead of two in the exact Newton case. Nonetheless, in the context of the truncated Newton method, *the two methods are equivalent in terms of computation cost*, as it is demonstrated in the next paragraph.

2.3. Computation cost. Consider the resolution of the linear system (1.4) for the computation of the descent direction Δp_k. The right hand side in (1.4) is the opposite misfit gradient $-\nabla f(p)$. In the matrix free context we consider, the gradient is computed using the first-order adjoint state method, which requires to compute the incident wavefield $\overline{u}(p)$ (resolution of one forward problem) and the adjoint wavefield $\overline{\lambda}(p)$ (resolution of one adjoint problem). The computation of $H(p)v$ or $B(p)v$ thus only requires to solve two additional problems: one forward problem for the computation of $\overline{\alpha}_v(p)$, and one adjoint problem, either for the computation of $\overline{\mu}(p)$ for $H(p)v$ or $\overline{\xi}(p)$ for $B(p)v$. Even if the computation of $\overline{\lambda}(p)$ is not required for the computation of $B(p)v$, it is imposed by the computation of the right-hand side of the linear system (1.4). The computation cost of the action of the Hessian operator or its Gauss-Newton approximation on an arbitrary vector is thus the same in terms of number of wave equations to be solved.

As a consequence, the resolution of (1.4) with a matrix free conjugate gradient algorithm requires to solve $2 + 2 \times N_{iter}$ wave equations where N_{iter} is the number of conjugate gradient iterations that have to be performed, whether in the Gauss-Newton approximation or in the exact Newton context.

In the next section, we investigate how the truncated Newton method can be efficiently implemented in the FWI context.

3. The truncated Newton method.

3.1. Sketch of the algorithm. We first present the sketch of the truncated Newton method using first-order and second-order adjoint state formulas, in the exact Newton framework, based on the computation of the descent direction by the conjugate gradient algorithm.

The algorithm 1 is composed of two nested loops. We refer to the outer loop as the nonlinear iteration loop, and to the inner loop as the linear iteration loop. The inner loop is devoted to the resolution of the linear system (1.4) using the conjugate gradient algorithm. The solution of this system is the descent direction Δp, which is used to update the current parameter model p.

In his survey, Nash [47] gives two important conditions for the truncated Newton method to be efficient, expressed as:

- each iteration of the conjugate gradient should be performed efficiently;
- the iteration number performed in the internal loop should be reduced as much possible.

The adjoint state method presented in the previous section ensures that the first condition is satisfied. The main question is thus the definition of an adequate stopping criterion to prevent over-solving while ensuring good convergence properties.

Data: p_0, ϵ, φ
Result: $\underset{p}{\operatorname{argmin}}\, f(p)$
$p = p_0$;
while $f(p) > \epsilon$ **do**
  solve $S(p)\overline{u} = \varphi$;
  solve $S(p)^\dagger \overline{\lambda} = (d - \overline{u})$;
  compute $\nabla f(p)$;
  set $\Delta p = 0, \;\; r = \nabla f(p), \;\; x = -r$;
  while *convergence criterion is not met* **do**
    solve $S(p)\overline{\alpha}_v = \Phi_v(\overline{p}, \overline{u})$;
    solve $S(p)^\dagger \overline{\mu} + R^\dagger R \overline{\alpha}_v + \sum_{j=1}^m (\partial_{p_j} S(p).v_j)^\dagger \overline{\lambda} = 0$;
    compute $H(p)x$;
    $\beta_1 = (H(p)x, x)$;
    $\beta_2 = \|r\|^2$;
    $\Delta p = \Delta p + (\beta_1/\beta_2)\, x$;
    $r = r + (\beta_1/\beta_2)\, H(p)x$;
    $x = -r + \left(\|r\|^2/\beta_2\right) x$;
  end
  compute γ (globalization method);
  $p = p + \gamma \Delta p$;
end

Algorithm 1: Truncated Newton method algorithm

One could think naively that the solution of the linear system (1.4) should be computed as accurately as possible, to provide a descent direction that generates the most important decrease of the cost function. However this is not true. Let us recall the principle of the Newton method. The computation of the descent direction Δp_k at each nonlinear iteration amounts to minimizing the quadratic function $q_k(\Delta p_k)$ defined by

$$q_k(\Delta p_k) = f(p_k) + (\nabla f(p_k), \Delta p_k) + \frac{1}{2}(H(p_k)\Delta p_k, \Delta p_k) \tag{3.1}$$

If this approximation is accurate, there is interest in computing a precise solution to the problem (1.4). Conversely, when this quadratic approximation is inexact, computing an exact solution of the system (1.4) amounts to the exploration of an unwanted zone. A measurement of the accuracy of the local quadratic approximation of the misfit function $f(p)$ can thus provide an appropriate stopping criterion. The way this is accounted for depends on the globalization method which is chosen.

3.2. The globalization method. The Newton algorithm is proven to be only locally convergent. A scaling of the descent direction Δp_k is therefore necessary to render the method globally convergent, in other words to ensure that the method converges to a local minimum whatever is the starting point p_0. Two families of methods exist for the computation of this scaling. The choice of the globalization strategy is important, as it conditions the discussion on the stopping criterion.

3.2.1. The linesearch method. The simplest globalization strategy is known as the linesearch procedure. The exact linesearch method consists in solving exactly the sub-problem

$$\min_\gamma f(p_k + \gamma \Delta p_k). \tag{3.2}$$

However this method is never implemented, because of its expensive computation cost. In addition, as it has been mentioned when considering the required accuracy for solving the linear system (1.4), there is no interest in solving exactly the sub-problem (3.2). Both the quadratic approximation $q_k(\Delta p_k)$ and the solution of the linear system (1.4) are potentially inaccurate. Instead, a coefficient γ enforcing the two Wolfe conditions is computed (see for instance [6] or [49]). These two conditions require that the model update $\gamma \Delta p_k$ is not too small and provides a sufficient decrease of the misfit function $f(p)$. Satisfying these two conditions ensures the global convergence.

3.2.2. The trust-region method. An alternative to the linesearch procedure is the trust region globalization method. Instead of computing first the descent direction and then a proper scaling coefficient, the scaling of the descent direction is computed implicitly by minimizing the quadratic function $q_k(\Delta p_k)$ within a *trust region* of radius $r_k \in \mathbb{R}_*^+$. At each nonlinear iteration, the following constrained quadratic problem is solved

$$\min_{\Delta p_k} q_k(\Delta p_k) \;\; \textit{subject to} \;\; \|p_k + \Delta p_k\| < r_k. \tag{3.3}$$

The role of the radius r_k is to reflect the accuracy of the local quadratic approximation $q_k(\Delta p_k)$. It is updated at each nonlinear iteration by comparison between the decrease provided by the selected model update Δp_k predicted by the quadratic form $q_k(\Delta p_k)$ and the actual decrease observed by evaluating the misfit function $f(p)$. The following quantity ρ is computed

$$\rho = \frac{f(p) - q_k(\Delta p_k)}{f(p) - f(p + \Delta p_k)}. \tag{3.4}$$

At each iteration k, the radius r_k is reduced until ρ becomes larger than a threshold $\beta_1 > 0$. If in addition ρ is larger than a second threshold $\beta_2 > \beta_1$, the radius r_k is enlarged.

Note that the optimal solution of the system (3.3) satisfies

$$(H(p_k) + \beta I)\Delta p_k = -\nabla f(p_k), \tag{3.5}$$

for some $\beta \geq 0$, where $H(p_k) + \beta I$ is symmetric definite positive [48]. Different methods are proposed to solve this sub-problem (see Moré [31] or the dogleg method [56]). For large-scale inverse problem, Steihaug demonstrates that an approximate solution of the sub-problem (3.3) can be found using the conjugate gradient method [67]. This amounts to construct a sequence Δp_k^i converging to the solution of (3.3). One important property is that the sequence of the norm of the iterates is an increasing sequence:

$$\|\Delta p_k^{i+1}\| > \|\Delta p_k^i\|. \tag{3.6}$$

Therefore, provided the initial iterate satisfies $\Delta p_k^0 = 0$, a natural stopping criterion for the resolution of (1.4) through the conjugate gradient method can be derived: the inner iterations stop whenever

$$\|p_k + \Delta p_k^i\| > r_k. \tag{3.7}$$

This yields an efficient way to account for the accuracy of the quadratic approximation $q_k(\Delta p_k)$ in the inner iterations.

We have first implemented the truncated Newton method using this framework. However, we have been faced to the following difficulty: the first iterate Δp_k^1 satisfies systematically

$$\|p_k + \Delta p_k^1\| > r_k, \tag{3.8}$$

which yields a premature ending of the conjugate gradient algorithm. Therefore, the computed descent direction is the steepest-descent direction $-\nabla f(p_k)$, and the resulting method is equivalent to the steepest-descent algorithm globalized by the trust region method. This is due to the strong nonlinearity of the misfit function, as mentioned by Le Toint et al [28]. In this article, Le Toint et al propose a sophistication of the initial Steihaug-Toint algorithm using the Lanczos algorithm to mitigate this difficulty. The approach that is developed is however more complex, and requires substantial additive computational efforts. Moreover, the numerical results that are presented do not emphasize significant improvements compared to the original Steihaug algorithm.

Therefore, we have turned to the definition of stopping criterion for the conjugate gradient method accounting for the accuracy of the quadratic approximation $q_k(d)$ in the context of linesearch globalization methods.

3.3. Stopping criterion for linesearch globalization methods. Eisenstat [17] proposes three stopping criteria related to the accuracy of the local quadratic approximation. For each of these criteria, a convergence proof of the truncated Newton method is provided. These criteria take the following form: the linear iterations are stopped whenever

$$\|H(p_k)\Delta p_k + \nabla f(p_k)\| \leq \eta_k \|\nabla f(p_k)\|. \tag{3.9}$$

In the expression (3.9), the quantity η_k is defined as the *forcing term*. The amplitude of η_k controls the required accuracy on the resolution of the system (1.4). The three different stopping criteria proposed by Eisenstat are associated with three possible definitions of η_k.

$$\begin{cases} \eta_{k,1} = \dfrac{\|\nabla f(p_k) - \nabla f(x_{p-1}) - \gamma_{k-1} H(p_{k-1})\Delta p_{k-1}\|}{\|\nabla f(p_{k-1})\|} \\[2ex] \eta_{k,2} = \dfrac{|\|\nabla f(p_k)\| - \|\nabla f(x_{p-1}) + \gamma_{k-1} H(p_{k-1})\Delta p_{k-1}\||}{\|\nabla f(p_{k-1})\|} \\[2ex] \eta_{k,3} = a_1 \left(\dfrac{|\|\nabla f(p_k)\|}{\|\nabla f(p_{k-1})\|} \right)^{a_2}, \quad a_1 \in [0,1], \;\; a_2 \in [1,2]. \end{cases} \tag{3.10}$$

The two first forcing terms $\eta_{k,1}$ and $\eta_{k,2}$ compare the values of the misfit gradient $\nabla f(p_k)$ with its first-order Taylor expansion given by

$$\nabla f(p_k) \simeq \nabla f(p_{k-1}) + \gamma_{k-1} H(p_{k-1})\Delta p_{k-1} + o(\|d_{k-1})\| \tag{3.11}$$

Therefore the expressions $\eta_{k,1}$ and $\eta_{k,2}$ are related to the quality of the quadratic approximation $q_k(\Delta p_k)$: they decrease when the accuracy of the approximation increases, and increase when the accuracy of the approximation decreases. They represent an alternative to the measurement of the accuracy of $q_k(\Delta p_k)$ by the quantity ρ given at equation (3.4) within the trust-region context, which is based on a second-order expansion of the misfit function.

The third forcing term $\eta_{k,3}$ is not directly related to the accuracy of the quadratic approximation $q_k(\Delta p_k)$. However, a convergence proof is also given for the choice of this forcing term in [17]. In practice, with appropriate choices of a_1 and a_2, this forcing term can result in less over-solving than the two first ones.

The Eisenstat approach is thus an interesting alternative to the Steihaug algorithm, which yields an efficient truncated Newton method that can be applied to the strongly nonlinear FWI problem, provided additional safeguards are implemented to complement the Eisenstat stopping criterion.

3.4. Additional safeguards. The choice of the forcing term and the definition of the associated stopping criterion for the linear iteration must be complemented with an appropriate strategy to deal with the detection of negative curvatures. Indeed, while the Gauss-Newton approximation of the Hessian $B(p)$ is symmetric positive, the second-order part $C(p)$ makes the full Hessian operator $H(p)$ symmetric indefinite. Therefore, during the resolution of the linear system (1.4) with the conjugate gradient algorithm, the probability of encountering a curvature associated with a negative eigenvalue of the operator $H(p)$ is not negligible. In this case, the linear iterations are stopped and the last value of the descent direction Δp_k which is computed is returned. If this negative curvature is met at the very first linear iteration, the steepest-descent direction is returned.

Besides, we also implement the "safeguards" proposed by Eisenstat [17]. Far from the solution, it is useless to impose a too restrictive forcing term. Therefore a modification of the forcing term is introduced, based on the estimation of the convergence rate. For the two first forcing terms $\eta_{k,1}, \eta_{k,2}$ this additional step consists in the following:

$$\text{If} \quad \eta_{k-1,j}^{(1+\sqrt{5})/2} > 0.1, \quad \text{then} \quad \eta_{k,j} = \max\{\eta_{k,j}, \eta_{k-1,j}^{(1+\sqrt{5})/2}\}, \quad j = 1, 2. \tag{3.12}$$

For the third forcing term, the "safeguard" step consists in

$$\text{If} \quad \gamma\eta_{k-1,3}^{\alpha} > 0.1, \quad \text{then} \quad \eta_{k,i} = \max\{\eta_{k,i}, \gamma\eta_{k-1,3}^{\alpha}\}. \tag{3.13}$$

Finally, we implement the last forcing condition:

$$\text{If} \quad \eta_{k,j} > 1, \ \text{then} \quad \eta_{k,j} = 0.9, \quad j = 1, \ldots, 3. \tag{3.14}$$

3.5. Preconditioning. Finally, from a computational point of view, the performance of the truncated Newton algorithm can be improved by the use of preconditioners for the resolution of the inner linear systems (1.4). This requires the computation of an approximation of $H(p)^{-1}$ in a matrix free form. The use of standard preconditioners such as the Incomplete LU [38] or the incomplete Choleski [3] is thus impracticable.

One possibility is the use of an l-BFGS approximation of the Hessian operator. This approximation can be updated at each outer nonlinear iteration. Starting from the identity matrix, scaled by an appropriate factor, an l-BFGS approximation Q_k is built, and the linear system (3.2) is replaced by

$$Q_k H(p_k)\Delta p_k = -Q_k \nabla f(p_k). \tag{3.15}$$

If only one iteration of the conjugate gradient is performed, then the descent direction Δp_k satisfies

$$\Delta p_k = -Q_k \nabla f(p_k), \tag{3.16}$$

that is to say Δp_k is the standard l-BFGS descent direction. In this case, the truncated Newton method can be seen as a generalization of the l-BFGS method.

Another possibility, proposed by Nocedal [45], is the computation of an l-BFGS approximation of the Hessian operator at each nonlinear iteration during the resolution of the system (1.4). Starting from the identity matrix, scaled by an appropriate factor, an approximation M_k is built and updated at each conjugate gradient iteration. This approximation is then used to precondition the system (1.4) at the iteration $k+1$. As it has been mentioned, the resolution of the system (1.4) by the conjugate gradient amounts to the minimization of the quadratic form $q_k(d)$. From its definition (3.1), $q_k(d)$ satisfies

$$\nabla^2 q_k(d) = H(p_k), \quad \nabla q_k(d) = H(p_k)d + \nabla f(p_k). \tag{3.17}$$

The quantity $\nabla q_k(d)$ corresponds to the residuals of the linear system (1.4). Therefore, an approximation of $H(p_k)$ can be computed through the conjugate gradient iterations using the successive values of the residuals.

3.6. Final algorithms. We propose the two followings algorithms, corresponding respectively to outer and inner l-BFGS preconditionings, which summarize the discussion on an efficient implementation of the truncated Newton method for FWI.

```
Data: p_0, ϵ, φ
Result: argmin_p f(p)
p = p_0;
η = 0.9;
Q = I;
while f(p) > ϵ do
    solve S(p)ū = φ;
    solve S(p)^† λ̄ = (d − ū);
    compute ∇f(p);
    set Δp = 0;
    set r = ∇f(p);
    set y = Q∇f(p);
    set x = −r;
    while ‖H(x)p + ∇f(p)‖ > η‖∇f(p)‖ do
        solve S(p)ᾱ_v = Φ_v(p̄, ū);
        solve S(p)^† μ̄ + R^† Rᾱ_v + Σ_{j=1}^m (∂_{p_j} S(p).v_j)^† λ̄ = 0;
        compute H(p)x;
        β_1 = (H(p)x, x);
        if β_1 ≤ 0 then
            stop the inner iterations;
        else
            β_2 = (r, y);
            Δp = Δp + (β_1/β_2) x;
            r = r + (β_1/β_2) H(p)x;
            y = Qr;
            x = −r + ((r, y)/β_2) x;
        end
    end
    compute γ (linesearch);
    p = p + γΔp;
    update η with the chosen Eisenstat forcing term formula ;
    update Q with the l-BFGS formula base on the gradient value ∇f(p);
end
```

Algorithm 2: Outer l-BFGS preconditioned truncated Newton algorithm

Note that the preconditioned conjugate gradient version is now proposed in the inner loop, in comparison with the algorithm 1 where no preconditioning is applied. The forcing term η is computed through one of the three Eisenstat formula. An additional stopping criterion is added if a negative curvature is detected ($\beta_1 \leq 0$). The computation of the l-BFGS approximation Q, or the scaling factor γ are not specified so as to keep the things simple. A very efficient way of computing the matrix-vector product Qr and storing the matrix Q in a sparse form is proposed in [48].

Data: p_0, ϵ, φ
Result: $\underset{p}{\operatorname{argmin}}\, f(p)$
$p = p_0$;
$\eta = 0.9$;
$Q_1 = I$;
while $f(p) > \epsilon$ **do**
  solve $S(p)\overline{u} = \varphi$;
  solve $S(p)^{\dagger}\overline{\lambda} = (d - \overline{u})$;
  compute $\nabla f(p)$;
  set $\Delta p = 0$;
  set $r = \nabla f(p)$;
  set $y = Q_1 \nabla f(p)$;
  set $x = -r$;
  set $Q_2 = I$;
  while $\|H(x)p + \nabla f(p)\| > \eta \|\nabla f(p)\|$ **do**
    solve $S(p)\overline{\alpha}_v = \Phi_v(\overline{p}, \overline{u})$;
    solve $S(p)^{\dagger}\overline{\mu} + R^{\dagger} R\overline{\alpha}_v + \sum_{j=1}^{m} (\partial_{p_j} S(p).v_j)^{\dagger}\overline{\lambda} = 0$;
    compute $H(p)x$;
    $\beta_1 = (H(p)x, x)$;
    if $\beta_1 \leq 0$ **then**
      stop the inner iterations;
    else
      $\beta_2 = (r, y)$;
      $\Delta p = \Delta p + (\beta_1/\beta_2)\, x$;
      $r = r + (\beta_1/\beta_2)\, H(p)x$;
      $y = Q_1 r$;
      $x = -r + ((r, y)/\beta_2)\, x$;
      update Q_2 with the l-BFGS formula based on the residual values r;
    end
  end
  compute γ (linesearch) ;
  $p = p + \gamma \Delta p$;
  update η with the chosen Eisenstat forcing term formula ;
  switch preconditioners for the next nonlinear iteration: $Q_1 = Q_2$;
end

Algorithm 3: Inner l-BFGS preconditioned truncated Newton algorithm

4. Numerical results. We begin this section with the presentation of the context in which we perform the different tests on the truncated Newton method. Two test cases are presented. The first is standard in the seismic imaging community, and is based on the MARMOUSI II pressure wave velocity model [41]. We use it as a benchmark to calibrate the truncated Newton method, in particular for the choice of the forcing term that control the stopping criterion for the inner linear iterations. An analysis of the convergence of this method with respect to the forcing term is presented. A comparison of its overall performance with the standard steepest-descent algorithm and the l-BFGS method is also given. The second test case involves a near-surface imaging of concrete structure presenting high velocity contrasts. This test case exhibits an important complexity in the recorded data sets, generated by the presence of high amplitude multiple reflections. As suggested by the analysis of

the role of the inverse Hessian operator by Pratt [58], using the truncated Newton method in this case allows to account more accurately for this operator and improves the estimation of the subsurface parameters, compared to standard methods.

4.1. Context. The numerical tests we present are performed in the 2D frequency domain, using an acoustic modeling of the wave propagation. The density is assumed to be constant, and the subsurface is described by the pressure wave velocity v_p. In this context, the forward problem (1.1) associated with a given circular frequency ω is equivalent to the Helmholtz equation

$$\omega^2 u + v_p^2 \Delta u = \varphi \tag{4.1}$$

The equation (4.1) is discretized with a fourth-order finite difference scheme with a compact stencil [30]. Perfectly Matched Layers (PML) [4, 42] are used to avoid fictitious reflections on the boundaries of the computation domain. The resolution of (4.1) reduces to the resolution of a sparse linear system. This is performed through a parallel LU factorization using the MUMPS algorithm [1]. In addition, the LU factorization of the stiffness matrix associated with the discretization of (4.1) is reused to solve the adjoint problems. This interesting feature is one of the reason for working in the frequency domain: provided the LU factorization of the stiffness matrix system can be stored, this approach largely reduces the computational costs, compared to the time domain approach. This is especially important when the number of data-sets is large: the same LU factorization is used to solve the forward and adjoint problems associated with each data-set.

In the following two tests which are presented, based on this wave propagation modeling, an estimation of the pressure wave velocity is computed using a FWI scheme. For the two tests, the performances of the steepest-descent algorithm, the l-BFGS method, and the truncated Newton algorithm (in the Gauss-Newton approximation and in the exact Newton context) are presented. We implement our own version of all these algorithms to propose an analysis that does not depend on implementation details, In particular, we implement the same linesearch globalization method for the three algorithms. This linesearch method satisfies the Wolfe conditions [6, 48]. In addition, the computational efficiency of the inversion schemes is compared in terms of the required number of forward problem resolutions. This gives a better insight of the computational cost associated with the three methods than the comparison of

- the number of nonlinear iterations: the truncated Newton method performs significantly less nonlinear iterations, but each of these requires a far more important computation effort;
- the overall computation time: the implementation of the two minimization algorithms have not been optimized in the same way, and the computation time depends on these implementation details.

Note that, in all experiments that follow, the memory parameter l for the l-BFGS method, which corresponds to the number of gradient that are stored to compute the approximation of the inverse Hessian, is set to $l = 20$. This rather large value is chosen to produce a reasonably accurate l-BFGS approximation of the Hessian operator. However, performing the same tests with $l = 5$ or $l = 40$ has led us to the same conclusions. Note also that the l-BFGS approximations are built from the identity matrix: no initial estimation of the inverse Hessian is provided.

4.2. The Marmousi II test case.

4.2.1. Presentation. The first Marmousi model is a synthetic pressure wave velocity model designed at IFPEN (Institut Français du Pétrole Energies Nouvelles) in 1988 and proposed as a benchmark for testing seismic imaging method. The geometry of this model is based on a realistic profile. It was created to produce complex seismic data requiring advanced processing techniques to obtain a correct earth image. In 2006, an upgrade to the so-called Marmousi II model is proposed [41]. The initial model is enlarged, and associated shear wave velocity and density models are added for elastic wave propagation modeling. In this study, we restrict ourselves to the acoustic approximation, and we only use the Marmousi II pressure wave velocity model, which is presented in figure 4.1. Note that the ratio between vertical and horizontal distances is not respected to improve the readability of the model figures.

The Marmousi II model is 16 km wide and 3.5 km deep, with a 400 m deep water layer at the top of the model. This layer is kept constant in the inversion scheme for all the following experiments, which results in a reduction of the parameter space and a stabilization of the problem: since these parameters are close from the sources and the receivers, the wavefield is very sensitive to small variations of these parameters.

We use a discretization step of 25 m, which corresponds to approximately 10^5 pressure wave velocity discrete parameters. The PML size is set to 10 discretization points on each side of the domain, hence we do not implement a free-surface condition. We use a marine seismic experiment acquisition system composed of 144 sources and 660 receivers located at the top of the model (25 m depth) in the water layer. We first compute 4 synthetic data-sets, for the frequencies: 3 Hz, 5 Hz, 8 Hz, 12 Hz. The corresponding number of discrete data is approximately $380,000$. We use a smoothed version of the exact Marmousi II model as initial guess (see fig. 4.1). This smoothing is performed using the Seismic Unix `smooth2` function, which performs a quadratic interpolation of the model, with smoothing parameters $r_1 = r_2 = 20$. This model could be for instance the result of a travel-time tomography of the data.

We invert simultaneously the 4 data-sets: the misfit function is computed as the sum of the misfit between the predicted and recorded data. An incremental strategy could be also used: this would consist in inverting each data-set one by one, starting by the lowest frequency data, and using the inversion result of one data-set as a starting model for inverting the next data-set. This incremental strategy can be particularly useful in some cases to mitigate strong nonlinearities of the misfit function [66]. However, in this particular case, we have not noticed any differences between the simultaneous and the incremental approaches.

4.2.2. Convergence analysis of the truncated Newton method. We investigate the convergence properties of the truncated Newton method depending on the choice of the forcing term. Four different methods are compared: the Newton method, the Gauss-Newton approximation, the outer l-BFGS preconditioned Newton method, and the outer l-BFGS Gauss-Newton method. The results yielded by the inner l-BFGS preconditioning are not satisfactory: a systematic premature ending of the algorithm is obtained, because of a linesearch failure (after 20 misfit function and gradient evaluations, no proper scaling of the descent direction is found). This failure indicates that the computed descent direction is not accurate enough. This probably results because the number of conjugate gradient iterations performed in the inner loop varies too rapidly for one nonlinear iteration to the other one for a useful l-BFGS approximation to be computed (see figure 4.3). The nonlinearity of the misfit function $f(p)$ could also cause the l-BFGS approximation of $H(p_k)^{-1}$ computed in the inner loop at iteration k to be far from the inverse hessian $H(p_{k+1})^{-1}$. Compared to the

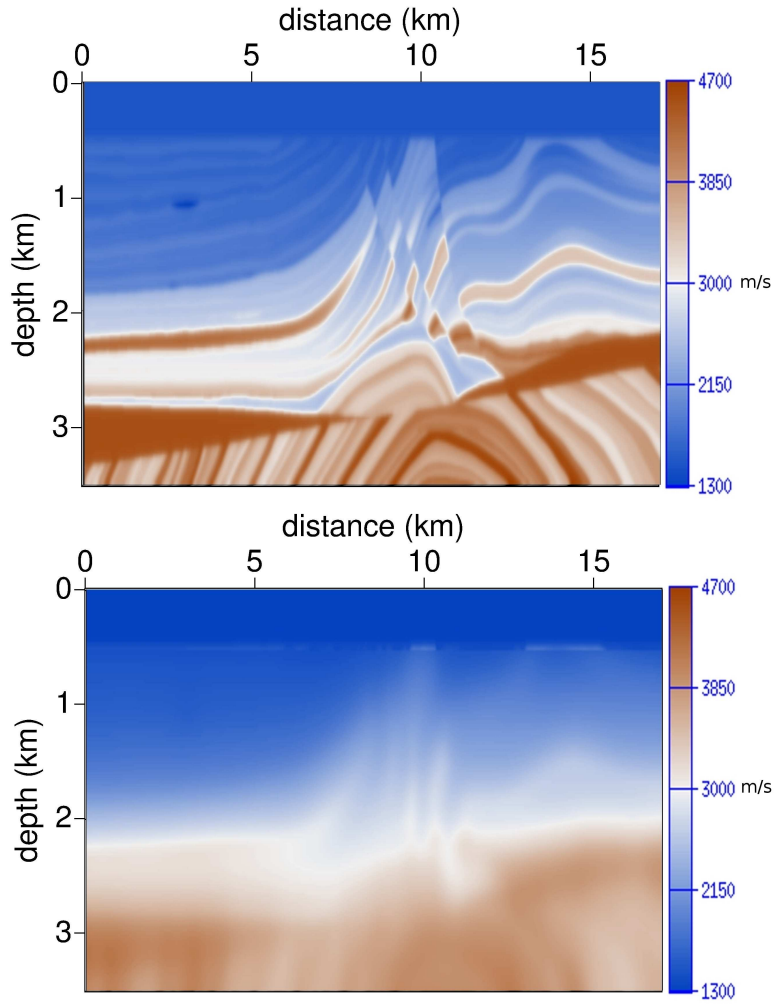

FIG. 4.1. *The Marmousi II pressure wave velocity model (left), initial guess (right).*

inner preconditioning, the outer preconditioning is far more robust.

Three different choices for the forcing terms are compared, following the proposition of Eisenstat (3.10).

- Choice (a): $\eta_k = \eta_{k,1}$.
- Choice (b): $\eta_k = \eta_{k,2}$.
- Choice (c): $\eta_k = \eta_{k,3}$ with $a_1 = 0.95$, $a_2 = 1$.

In standard implementations, a strong additional convergence criterion is used for the inner linear iterations to prevent from over-solving the inner problem: a maximum number of linear iterations is set. In our implementation we deliberately set this limit to a high level: the convergence is forced whenever the number of inner iterations reaches $N_{iter} = 100$, while values of 20 or 30 are usually found in the literature [47]. Our objective is testing the importance of the definition of the forcing term: we want to prevent the inner loop convergence to be driven by a maximum number of authorized iterations instead of the forcing term.

	Newton	Gauss-Newton	Pr. Newton	Pr. Gauss-Newton
Choice (a)	5.96	5.5	6.21	4.44
Choice (b)	4.25	8.75	4.44	5.73
Choice (c)	8.51	1.	4.92	3.79

TABLE 4.1
Average number of inner CG iterations per nonlinear iteration.

The stopping criterion for the outer nonlinear iterations consists in satisfying the condition

$$f(p)/f(p_0) < 10^{-3}. \tag{4.2}$$

The convergence curves associated with these methods and the choice of the forcing term are presented in figure 4.2.

We show that the convergence is reached in each case, except for the forcing term (c) and the Gauss-Newton method, and the forcing term (b) and the Newton method. In these two cases, a local minimum of the misfit function is reached, and the method is unable to minimize the misfit function any further. Conversely, the use of the forcing term (a) yields the convergence for the four methods (Newton, Gauss-Newton, and their preconditioned versions). In terms of convergence speed, the use of the forcing term (a) provides a significant enhancement of the preconditoned Newton method. For the Newton method, the choice of the forcing term (a) induces slightly more computational efforts compared to the forcing term (c). For the Gauss-Newton method and its preconditioned version, the different forcing terms give approximately the same results, except for the forcing term (c) and the Gauss-Newton algorithm, the use of which makes the minimization process to converge towards a local minimum. This first analysis advocates for the choice of the forcing term (a), since it yields convergence in each case, and provides satisfactory convergence speed compared to the others.

We complement these tests with the analysis of the number of inner linear iterations performed and the size of the step computed through the linesearch method at each nonlinear iteration. These results are presented respectively in figures 4.3, and 4.4. The average number of linear iteration performed per nonlinear iteration is also provided in table 4.1.

Figure 4.3 shows that that the forcing term (a) leads to the highest reduction of the number of linear iterations by the number of nonlinear iterations for the Gauss-Newton case and the preconditioned Newton case (Figure 4.3, rows 2 and 3). For the Gauss-Newton case, the forcing term (c) only performs one linear iteration per nonlinear iteration: this emphasizes the convergence failure of the method in this case. For the Newton case (row 1) the forcing term (b) seems more adapted. For the preconditioned Gauss-Newton case (row 4), the three forcing terms give approximately the same results. The table 4.1 shows that the average number of inner linear iteration is more stable for the forcing term (a), with respect to the optimization which is used. Finally, the figure 4.4 emphasizes an interesting property of the forcing term (a): this choice seems to provide a better stability in terms of the step-size computed through the

linesearch procedure. This property is important regarding the computation cost of computing this step-size through the linesearch procedure. A change in the steplength is synonym of at least one additional computation of the gradient of the misfit function, which amounts to an additional forward and adjoint problems.

From these tests, we thus conclude that the forcing term (a) is the best adapted to our purpose. Since the best forcing term has been determined, we now compare the performance of the truncated Newton method using this forcing term with the steepest-descent method and the l-BFGS method.

4.2.3. Comparison with the gradient-based methods. We present in figure 4.5 the convergence curves obtained for the steepest-descent algorithm, the l-BFGS algorithm, and the truncated Newton algorithm, either in the Gauss-Newton approximation or using the exact Hessian operator. As expected, the steepest-descent algorithm converges very slowly. It requires to solve more than ten thousand wave propagation problems, whereas the truncated Newton method and the l-BFGS method require to solve less than two thousand of these problems. This emphasizes the interest of using Newton and quasi-Newton method for the FWI problem.

Fo gain some additional insight on the performances of the truncated Newton method compared to the l-BFGS method, we also present in figure 4.5 the convergence curves obtained for the l-BFGS algorithm, the Gauss-Newton method, the Newton method, and the outer preconditioned versions of the two latter. We show that, for the Marmousi case, the l-BFGS algorithm stays more efficient than the truncated Newton method in terms of computational efficiency. The best results for the truncated Newton method are obtained by the preconditioned Gauss-Newton version. The effect of the outer l-BFGS preconditioning is important: compare for instance the convergence of the Newton method with its preconditioned version. These results indicate that the l-BFGS approximation of the inverse Hessian operator is accurate enough for the l-BFGS method to converge rapidly. Note, however, that, contrary to the steepest-descent method, which converges slowly, the computational effort required by the l-BFGS method and the truncated Newton method using the Gauss-Newton approximation is of the same order.

The wave velocity model estimated with the l-BFGS method and its difference with the exact model are presented in figure 4.6. From a very low resolution approximation of the solution, the FWI method provides a high resolution quantitative estimation of the solution. The results obtained with the different inversion methods are very similar, even for the steepest-descent method, as it can be seen in figure 4.7, where the differences between the l-BFGS estimations and the estimations provided by the other inversion schemes are presented.

From this experiment, we conclude that, compared to the steepest-descent, the truncated Newton method requires approximately the same computation cost than the l-BFGS method, which corroborates the results of Nash and Nocedal [63]. On this particular test case, we also show that the Gauss-Newton approximation performs better than the Newton approximation, the Newton method converging more slowly. Preconditioning the two methods with an outer l-BFGS approximation efficiently enhances the convergence speed. Finally, among all the inversion schemes we have tested, the l-BFGS method seems to be the most efficient. Note, however, that the performances of the truncated Newton method should be enhanced by a more appropriate preconditioning of the inner linear system, or the use of a deflated conjugate gradient algorithm [65] for instance.

In the next section, we investigate another test case inspired from a near-surface

imaging application.

4.3. A near-surface imaging application. We consider the exact wave velocity model presented in figure 4.8. This model is composed of a homogeneous background and two superposed concrete structures. Detecting and correctly imaging these two concrete foundations buried in the subsurface at few meters depth is a challenge in seismic imaging. Note that the average scale is much shorter than for the previous test case. The depth of investigation is limited to 3 m, the width of the exact model is 15 m. The discretization step is 0.15 m, which amounts to 2121 discrete subsurface parameters. As for the Marmousi test case, we add 10 points width PML at all the boundaries. The short distance of propagation inside the model due to the near-surface configuration makes difficult to discriminate between the different type of waves recorded in the seismograms. The very high velocity contrast between the background (300 m.s^{-1}) and the concrete foundations (4000 m.s^{-1}) generates high-amplitude reflections. In addition, the close distance between the two structures is responsible for important multiple scattering: this is the reason why we consider this test case.

We first use a full acquisition system, that is to say using four lines of sources/receivers on each side of the domain. In the second subsection, we investigate a more realistic case, replacing the bottom sources/receivers line with a layer at 500 m.s^{-1}.

4.3.1. Full acquisition case. We compute 9 data-sets, corresponding to the frequencies: 100 Hz, 125 Hz, 150 Hz, 175 Hz, 200 Hz, 225 Hz, 250 Hz, 275 Hz, 300 Hz. As for the Marmousi test case, the 9 data-sets are inverted simultaneously. The initial model is the homogeneous background model.

We show two time-domain data-sets computed with the exact wave velocity model and with the initial homogeneous velocity model to get a better insight of the multiple scattering (figure 4.9). These two data-sets are obtained using a source located at the surface, between the two concrete structures, at $x = 6.75$ m. The signal emitted by the source is a Ricker wavelet, centered on the frequency 150 Hz. We adopt the following data representation: the horizontal axis corresponds to the receivers, the vertical axis corresponds to the time from the source excitation. The amplitude of the pressure wavefield recorded by each receiver corresponds to the intensity of the black and white plots.

From figure 4.9, we show that the first-arrival waves are correctly predicted by the initial homogeneous model. On the left figure, the signal after the first arrival corresponds to the multiple reflections from the two concrete structures. This waves are not predicted by the initial homogeneous model.

We compare the convergence of the steepest-descent algorithm and the l-BFGS algorithm with the truncated Newton method on this test case. The corresponding curves are presented in figure 4.10. As for the previous case, the misfit function values are plotted as a function of the number of forward problem resolutions. From figure 4.10, we show that the steepest-descent, the l-BFGS algorithm and the Gauss-Newton truncated method fail to converge. The convergence curves of the preconditioned versions of the Gauss-Newton and exact Newton truncated methods are not presented, because they also stop at the very first iterations. Only the exact Newton method, without preconditioning, is able to converge.

The corresponding inversion results are presented in figure 4.11. The four results are plotted with the same color scale. As indicated by the misfit function decrease,

the two concrete structures are better reconstructed using the Newton method, even if the wave velocity amplitude is underestimated (this value reaches 810 m.s^{-1}, which is still far from the real value which is 4000 m.s^{-1}). The structures in themselves and their edges appear more clearly.

Why does the Newton method outperform the l-BFGS method and the Gauss-Newton method in this case? To answer to this question, we first compute the data in the time domain associated with each of the estimated wave velocity model, as previously done for the exact and initial model (Fig. 4.12). The residuals in the time domain (difference between the predicted data and the data associated with the exact model) are also presented in figure 4.13.

These figures demonstrate that the multi-scattered waves that appear between the two concrete structures are only correctly interpreted by the full Newton method.

In addition we compute the Hessian matrix $H(p)$ associated with the initial background model, and its Gauss-Newton approximation $B(p)$. The dimension of the problem is small enough for the computation of the Hessian to be tractable. The two matrices are presented on the figure 4.14. We show that the second-order part $C(p)$ of the Hessian operator is far from being negligible: the two matrices $H(p)$ and $B(p)$ are plotted with the same color scale, and the differences are important. This is an indication of the importance of the scattered wavefield and the presence of high amplitude second-order derivatives of the wavefield with respect to the wave velocity parameters. Besides, since the residuals $Ru_s - d_s$ do not decrease rapidly throughout the iterative process (otherwise the misfit function would be rapidly small), the second-order part of the Hessian operator $C(p)$ stays non negligible along the minimization process.

This matrix $C(p)$ may be responsible for the presence of negative eigenvalues in the Hessian operator. We compute the 500 largest eigenvalues of each matrix using the Lanczos algorithm [27] (we use the `FORTRAN` code Blzpack [40]). The result is presented in figure 4.15. As expected, the Gauss-Newton matrix is symmetric positive, with numerous small eigenvalues associated with poorly illuminated parameters. The initial full Hessian matrix presents numerous negative eigenvalues. The "strong" indefiniteness of the Hessian can thus explain the poor performance of the l-BFGS algorithm: the l-BFGS formula provides a symmetric positive definite approximation of the inverse Hessian. Therefore, during the first iterations, this approximation is probably inaccurate and prevents the l-BFGS algorithm from minimizing efficiently the misfit function $f(p)$. Since the amplitude of the residuals keeps large, the second-order part $C(p)$ remains important and the true Hessian matrix remains indefinite, while the successive l-BFGS approximations are positive definite. This discrepancy between the actual Hessian and its l-BFGS approximation is probably responsible for the failure of the l-BFGS inversion scheme. The truncated Newton method under the Gauss-Newton approximation faces the same difficulty: the Hessian operator is approximated using a symmetric positive matrix, whereas the true Hessian matrix has large negative eigenvalues. The use of the Gauss-Newton method is therefore inappropriate in this case. Only the truncated Newton method using the full Hessian can mitigate this difficulty. Finally, the inaccuracy of the l-BFGS approximation of the inverse Hessian operator also explains why the preconditioned version of the Newton and the Gauss-Newton truncated algorithm fails to converge: it degrades the conditioning of the linear systems associated with the computation of the Newton

descent direction

We test the method on a more realistic test case to pursue our investigation of the robustness of the truncated Newton method.

4.3.2. A more realistic case. We consider in this section an acquisition system without the bottom line of sources/receivers. We use a larger domain (30 m large and 3.75 m width), to compensate for this lack of illumination. We also insert a layer of velocity 500 m.s^{-1} at the bottom of the true model. This layer is included in the initial model p_0. The corresponding exact model and initial model are presented in figure (4.16). We keep the same discretization step (0.15 m), which amounts to 5025 discrete subsurface parameters. We use only three lines of sources/receivers on the top and the lateral sides of the domain. We keep 9 data-sets, corresponding to the previous frequencies: from 100 Hz to 300 Hz each 25 Hz. As in the previous case, the 9 data-sets are inverted simultaneously. We also introduce a white noise on each data-set, with different amplitudes, to investigate the robustness of the method. The results corresponding to the steepest-descent, the l-BFGS inversion, the Gauss-Newton inversion and the exact-Newton inversion are presented in figures 4.17, 4.18, 4.19.

The results we obtain on this last test case confirm that the truncated Newton method using the exact Hessian is preferable in this case than the l-BFGS method. The results without noise show clearly that the concrete structures are better reconstructed using the truncated Newton scheme, even if the contrasts are still far from being completely recovered. In addition, introducing noise up to -9 dB does not particularly degrade the results. The reflections from the bottom layer seem to correctly compensate for the lack of illumination consecutive to the removal of the fourth bottom line of sources/receivers. Once again, the better performance of the truncated Newton method should be related to the presence of large negative eigenvalues in the initial Hessian operator. These negative eigenvalues originate from the presence of a large amplitude multi-scattered wavefield. The l-BFGS method and the Gauss-Newton approximation used in the truncated Newton framework are unable to approximate the Hessian operator correctly, and therefore fail to converge.

5. Conclusion and perspectives. The FWI method is a very efficient tool for the computation of quantitative estimations of the subsurface parameters through the interpretation of the wave propagation. From a numerical point of view, the method amounts to the resolution of a large-scale inverse problem, which requires appropriate minimization algorithms. For a long time, gradient-based methods, such as the steepest-descent algorithm or the nonlinear conjugate gradient algorithm, and more recently the quasi-Newton method l-BFGS, have been used to solve this problem. In this study, we investigate the possibility and the interest of using another class of minimization algorithm: the truncated Newton method. Unlike gradient based methods, the truncated Newton method aims at solving the linear system associated with the computation of the Newton descent direction that is raised at each nonlinear iteration of the minimization algorithm. However, given the large-scale nature of the expected applications, the explicit computation of the Hessian operator is intractable. Therefore this linear system has to be solved using a matrix-free iterative solver. This requires the possibility of computing efficiently Hessian vector products.

This first requirement is fulfilled thanks to second-order adjoint state formulas. Using control optimal theory and Lagrangian formulations, we show that the resolution of the linear system for the computation of the Newton descent direction amounts to the resolution of $2 + 2 \times N_{iter}$ wave propagation problems where N_{iter} denotes the

number of conjugate gradient iterations, either in the exact Newton context, or in the Gauss-Newton approximation.

The second requirement for the efficient implementation of the truncated Newton algorithm relies on an efficient stopping criterion of iteration for the inner conjugate gradient algorithm. We show that this stopping criterion must be related to the accuracy of the local quadratic approximation that is made at each nonlinear iteration in the framework of Newton minimization methods. One possible way to account for this accuracy is the use of the Eisenstat stopping criterion [17] and one of the associated forcing term that he has proposed. In addition, an l-BFGS preconditioning of the linear systems associated with the computation of the Newton descent direction can be used to speed-up the convergence. Two algorithms are proposed, corresponding to the computation of an l-BFGS approximation of the inverse Hessian operator during the inner linear iteration or the outer nonlinear iterations.

The numerical tests we perform show that the choice of the forcing term is crucial. Different performances can be obtained depending on this choice, which is illustrated by the different results we obtain on the Marmousi II test case, using three different forcing terms. On the basis of these results we select the forcing term that appears to be the most reliable.

The comparison between the truncated Newton method and the l-BFGS method on the Marmousi test case reveals the efficiency of the l-BFGS method. Among different versions of the truncated Newton method, the preconditioned Gauss-Newton version provides the best performance in terms of computational efficiency. However, the l-BFGS method still provides better results. This emphasizes that for the reconstruction of reasonably complex subsurface structures, quasi-Newton methods shall be preferred to more sophisticated method as the truncated Newton method.

However, a second test case, derived from a near-surface application for imaging high velocity embedded bodies, emphasizes the limitations of the l-BFGS method. In this case, the l-BFGS method is unable to converge, while the truncated Newton method provides reliable results. The high amplitude multi-scattered wavefield generated by the structures makes the starting Hessian operator strongly indefinite. The l-BFGS approximation of the Hessian is positive definite by construction: therefore, in this case, this approximation is far from the true Hessian, and the descent direction that is computed is not accurate. As a consequence, the results provided by the l-BFGS method and the truncated Newton method show that a very substantial improvement of the solution can be obtained using the truncated Newton method. This result is first obtained for a full acquisition system, then for a more realistic configuration, where the bottom line of sources/receivers is removed, and has been replaced by an identified layer. In this configuration the results still demonstrates the interest of using the truncated Newton method instead of the l-BFGS method. In the presence of a white noise on the data, the truncated Newton method still converges while the l-BFGS method still fails.

Based on these preliminary results, our main challenge remains to demonstrate the interest of the truncated Newton method for realistic applications. As a first step, the method could be tested on the BP 2004 benchmark model [5]. This synthetic model, partly inspired from the geology of the Mexico gulf, contains salt domes structures just below the sea bottom. This configuration is responsible for important contrasts in wave velocities between the water layer and the salt zones. This suggests that the same kind of complex data sets as the one presented in our near-surface case study, including high amplitude multiple reflected waves, may be generated in such a model.

In addition, the effect of multi-scattered waves could be amplified by considering multiple-contaminated data. Depending on the results obtained with this particular subsurface configuration, a second step may consist in applying the truncated Newton method to a 2D real data set, for instance the Valhall data set [61, 60, 26, 19]. The results on the near-surface case study we have presented seem to exhibit a good robustness of the truncated Newton method with respect to noise contamination of the data. This is in accordance with the interpretation of the truncated Newton method as an iterative regularization method (see Kaltenbacher [32]). However this has to be confirmed on a more realistic test case such as the BP model. The application of the method to real data such as the Valhall data also requires to adapt the method for anisotropic forward modeling. This then opens the way to the application of the method in a multi-parameter context, in which both P-wave velocity and anisotropic parameters are simultaneously estimated. The use of the truncated Newton method in this context is of particular interest since the inverse Hessian operator, additionally to its already identified properties, should help to treat more efficiently the trade-off between parameters. Since these trade-off are expressed by the off-diagonal blocks of the Hessian operator, one can hope that accounting more precisely for this operator in the inversion scheme should improve the accuracy of the multi-parameter estimations.

6. Acknowledgments. This study was funded by the SEISCOPE consortium ($http://seiscope.oca.eu$), sponsored by BP, CGG-VERITAS, ENI, EXXON-MOBIL, PETROBRAS, SAUDI ARAMCO, SHELL, STATOIL and TOTAL. This study was granted access to the HPC facilities of CIMENT (Université Joseph Fourier Grenoble), and of GENCI-CINES under Grant 2011-046091 of GENCI (Grand Equipement National de Calcul Intensif). The authors would also like to thank S. Bellavia and S. Gratton for useful discussions and advices.

REFERENCES

[1] P. Amestoy, I. S. Duff, and J. Y. L'Excellent, *Multifrontal parallel distributed symmetric and unsymmetric solvers*, Computer Methods in Applied Mechanics and Engineering, 184 (2000), pp. 501–520.

[2] J. Bee Bednar, Changsoo Shin, and Sukjoon Pyun, *Comparison of waveform inversion, part 2: phase approach*, Geophysical Prospecting, 55 (2007), pp. 465–475.

[3] M. Benzi, *Preconditioning techniques for large linear systems: A survey*, Journal of Computational Physics, 182 (2002), pp. 418–477.

[4] J-P Berenger, *A perfectly matched layer for absorption of electromagnetic waves*, Journal of Computational Physics, 114 (1994), pp. 185–200.

[5] F. J. Billette and S. Brandsberg-Dahl, *The 2004 BP velocity benchmark*, in Extended Abstracts, 67th Annual EAGE Conference & Exhibition, Madrid, Spain, 2004, p. B035.

[6] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal, and C. A. Sagastizábal, *Numerical Optimization, Theoretical and Practical Aspects*, Springer series, Universitext, 2006.

[7] A. J. Brenders and R. G. Pratt, *Efficient waveform tomography for lithospheric imaging: implications for realistic 2D acquisition geometries and low frequency data*, Geophysical Journal International, 168 (2007), pp. 152–170.

[8] ———, *Full waveform tomography for lithospheric imaging: results from a blind test in a realistic crustal model*, Geophysical Journal International, 168 (2007), pp. 133–151.

[9] F. Bretaudeau, *Modélisation physique à échelle réduite pour l'adaptation de l'inversion des formes d'ondes sismiques au génie civil et à la subsurface*, thèse de doctorat, Université de Nantes, Nantes, 2010.

[10] F. Bretaudeau, D. Leparoux, R. Brossier, and O. Abraham, *Small scale modeling, a tool to assess subsurface imaging methods : Application to seismic full waveform inversion*, in Expanded Abstracts, EAGE, 2009, p. P041.

[11] R. Brossier, S. Operto, and J. Virieux, *Seismic imaging of complex onshore structures by*

2D elastic frequency-domain full-waveform inversion, Geophysics, 74 (2009), pp. WCC63–WCC76.

[12] R.H. Byrd, P. Lu, and J. Nocedal, *A limited memory algorithm for bound constrained optimization*, SIAM Journal on Scientific and Statistical Computing, 16 (1995), pp. 1190–1208.

[13] G. Chavent, *Analyse fonctionnelle et identification de coefficients rpartis dans les quations aux drives partielles*, PhD thesis, Universit de Paris, 1971.

[14] J.F. Claerbout, *Towards a unified theory of reflector mapping*, Geophysics, 36 (1971), pp. 467–481.

[15] J. F. Claerbout, *Fundamentals of Geophysical Data Processing*, McGraw-Hill Book Co., 1976.

[16] K. Effelsen, *A comparison of phase inversion and traveltime tomography for processing of near-surface refraction traveltimes*, Geophysics, 74(6) (2009), pp. WCB11–WCB24.

[17] S. C. Eisenstat and H. F. Walker, *Choosing the forcing terms in an inexact Newton method*, SIAM Journal on Scientific Computing, 17 (1994), pp. 16–32.

[18] I. Epanomeritakis, V. Akçelik, O. Ghattas, and J. Bielak, *A Newton-CG method for large-scale three-dimensional elastic full waveform seismic inversion*, Inverse Problems, 24 (2008), pp. 1–26.

[19] V. Etienne, G. Hu, S. Operto, J. Virieux, O.I. Barkved, and J. Kommedal, *Three-dimensional acoustic full waveform inversion: algorithm and application to Valhall*, in Expanded Abstracts, 74th Annual EAGE Conference & Exhibition, Copenhagen, EAGE, 2012.

[20] A. Fichtner, B. L. N. Kennett, H. Igel, and H. P. Bunge, *Full waveform tomography for upper-mantle structure in the Australasian region using adjoint methods*, Geophysical Journal International, 179 (2009), pp. 1703–1725.

[21] ———, *Full waveform tomography for radially anisotropic structure: New insights into present and past states of the Australasian upper mantle*, Earth and Planetary Science Lettters, 290 (2010), pp. 270–280.

[22] A. Fichtner and J. Trampert, *Hessian kernels of seismic data functionals based upon adjoint techniques*, Geophysical Journal International, 185 (2011), pp. 775–798.

[23] Andreas Fichtner and Jeannot Trampert, *Resolution analysis in full waveform inversion*, Geophysical Journal International, 187 (2011), pp. 1604–1624.

[24] E. Forgues, E. Scala, and R. G. Pratt, *High resolution velocity model estimation from refraction and reflection data*, in Expanded abstracts, Soc. Expl. Geophys., 1998, pp. 1211–1214.

[25] O. Gauthier, J. Virieux, and A. Tarantola, *Two-dimensional nonlinear inversion of seismic waveforms: numerical results*, Geophysics, 51 (1986), pp. 1387–1403.

[26] Y. Gholami, R. Brossier, Stéphane Operto, V. Prieux, A. Ribodetti, and Jean Virieux, *Two-dimensional acoustic anisotropic (VTI) full waveform inversion: the Valhall case study*, SEG Technical Program Expanded Abstracts, 30 (2011), p. 2543.

[27] G. H. Golub, *Matrix Computation, third edition*, Johns Hopkins Studies in Mathematical Sciences, 1996.

[28] N. I. M. Gould, S. Lucidi, M. Roma, and P.L. Toint, *Solving the trust-region subproblem using the lanczos method*, Siam Journal on Optimization, 9 (1999), pp. 504–525.

[29] G. J. Hicks and R. G. Pratt, *Reflection waveform inversion using local descent methods: estimating attenuation and velocity over a gas-sand deposit*, Geophysics, 66 (2001), pp. 598–612.

[30] B. Hustedt, S. Operto, and J. Virieux, *Mixed-grid and staggered-grid finite difference methods for frequency domain acoustic wave modelling*, Geophysical Journal International, 157 (2004), pp. 1269–1296.

[31] J.Moré, *The levenberg-marquardt algorithm: Implementation and theory*, in Numerical Analysis, G. Watson, ed., vol. 630 of Lecture Notes in Mathematics, Springer Berlin / Heidelberg, 1978, pp. 105–116. 10.1007/BFb0067700.

[32] B. Kaltenbacher, A. Neubauer, and O. Scherzer, *Iterative Regularization Methods for Nonlinear Problems*, de Gruyter, Berlin, New York, 2008.

[33] P. Lailly, *The seismic inverse problem as a sequence of before stack migrations*, in Conference on Inverse Scattering, Theory and application, Society for Industrial and Applied Mathematics, Philadelphia, Robinson Bednar and Weglein, eds., 1983, pp. 206–220.

[34] ———, *The seismic problem as a sequence of before-stack migrations*, in Conference on Inverse Scattering: Theory and Applications, J. Bednar, ed., SIAM, Philadelphia, 1983.

[35] P. Lailly and D. Sinoquet, *Smooth velocity models in reflection tomography for imaging complex geological structures*, Geophysical Journal International, 124 (1996), pp. 349–362.

[36] F. X. Le Dimet, I. M. Navon, and D. N. Daescu, *Second-order information in data assimi-*

lation, American Meteorological Society, 184 (2002), pp. 501–520.

[37] F. X. Le Dimet and O. Talagrand, *Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects*, Tellus, 38A (1986), pp. 97–110.

[38] J. Lee and C. Lu J. Zhang, *Incomplete lu preconditioners for large scale dense dense complex linear systems from electromagnetic wave scattering problems*, Journal of Computational Physics, 130 (2003), pp. 629–648.

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

[40] O. A. Marques, *Blzpack: Description and user's guide*, tech. report, CERFACS, Toulouse, 1995.

[41] G. S. Martin, R. Wiley, and K. J. Marfurt, *Marmousi2: An elastic upgrade for marmousi*, The Leading Edge, 25 (2006), pp. 156–166.

[42] L. Métivier, *Utilisation des équations euler-pml en milieu hétérogène borné pour la résolution d'un problème inverse en géophysique.*, ESAIM: Proc., 27 (2009), pp. 156–170.

[43] ———, *Interlocked optimization and fast gradient algorithm for a seismic inverse problem*, Journal of Computational Physics, 230 (2011), pp. 7502–7518.

[44] L. Métivier, P. Lailly, F. Delprat-Jannaud, and L. Halpern, *A 2d nonlinear inversion of well-seismic data*, Inverse Problems, 27 (2011), p. 055005.

[45] J. L. Morales and J. Nocedal, *Automatic preconditioning by limited memory quasi-newton updating*, Siam Journal on Optimization, 10 (1999), pp. 1079–1096.

[46] ———, *Enriched methods for large-scale unconstrained optimization*, Computational Optimization and Applications, 21 (2000), pp. 143–154.

[47] S. G. Nash, *A survey of truncated Newton methods*, Journal of Computational and Applied Mathematics, 124 (2000), pp. 45–59.

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

[49] ———, *Numerical Optimization*, Springer, 2nd ed., 2006.

[50] G. Nolet, *A Breviary of Seismic Tomography*, Cambridge University Press, Cambridge, UK, 2008.

[51] S. Operto, C. Ravaut, L. Improta, J. Virieux, A. Herrero, and P. Dell'Aversana, *Quantitative imaging of complex structures from dense wide-aperture seismic data by multiscale traveltime and waveform inversions: a case study*, Geophysical Prospecting, 52 (2004), pp. 625–651.

[52] D. Pageot, S. Operto, M. Vallée, R. Brossier, and J. Virieux, *Lithospheric imaging from teleseismic data by frequency-domain elastic full-waveform tomography*, in AGU Fall Meeting Abstracts, 2010, pp. S31A–2036.

[53] R. E. Plessix, *A review of the adjoint-state method for computing the gradient of a functional with geophysical applications*, Geophysical Journal International, 167 (2006), pp. 495–503.

[54] R. E. Plessix, G. Baeten, J. W. de Maag, M. Klaassen, Z. Rujie, and T. Zhifei, *Application of acoustic full waveform inversion to a low-frequency large-offset land data set*, SEG Technical Program Expanded Abstracts, 29 (2010), pp. 930–934.

[55] R. E. Plessix and C. Perkins, *Full waveform inversion of a deep water ocean bottom seismometer dataset*, First Break, 28 (2010), pp. 71–78.

[56] M. J. D. Powell, *A new algorithm for unconstrained optimization*, Nonlinear Programming Academic Press, New York, (1970), pp. 31–66.

[57] R. G. Pratt, *Inverse theory applied to multi-source cross-hole tomography. part II : elastic wave-equation method*, Geophysical Prospecting, 38 (1990), pp. 311–330.

[58] R. G. Pratt, C. Shin, and G. J. Hicks, *Gauss-Newton and full Newton methods in frequency-space seismic waveform inversion*, Geophysical Journal International, 133 (1998), pp. 341–362.

[59] R. G. Pratt and M. H. Worthington, *Inverse theory applied to multi-source cross-hole tomography. Part I: acoustic wave-equation method*, Geophysical Prospecting, 38 (1990), pp. 287–310.

[60] V. Prieux, R. Brossier, Y. Gholami, S. Operto, J. Virieux, O.I. Barkved, and J.H. Kommedal, *On the footprint of anisotropy on isotropic full waveform inversion: the Valhall case study*, Geophysical Journal International, 187 (2011), pp. 1495–1515.

[61] V. Prieux, S. Operto, R. Brossier, J. Virieux, J.H. Kommendal, and O.I. Barkved, *Application of 2d acoustic frequency-domain full-waveform inversion to obc wide-aperture data from the valhall field*, SEG Technical Program Expanded Abstracts, 29 (2010), pp. 920–924.

[62] A. Roques, R. Brossier, J. Virieux, and J. Mars, *High-resolution seismic imaging applied to the characterization of very shallow highly contrasted structures*, in AGU Fall Meeting Abstracts, 2010, pp. NS41A–1500.

[63] J. Nocedal S. G. Nash, *A numerical study of the limited memory bfgs method and trun-*

cated newton method for large scale optimization, Siam Journal on Optimization, 1 (1991), pp. 358–372.

[64] Y. Saad, *Iterative methods for sparse linear systems*, SIAM, Philadelphia, 2003.

[65] Y. Saad, M. Yeung, J. Erhel, and F. Guyomarc'h, *A deflated version of the conjugate gradient algorithm*, SIAM Journal on Scientific Computing, 21 (2000), pp. 1909–1926.

[66] L. Sirgue and R. G. Pratt, *Efficient waveform inversion and imaging : a strategy for selecting temporal frequencies*, Geophysics, 69 (2004), pp. 231–248.

[67] T. Steihaug, *The conjugate gradient method and trust regions in large scale optimization*, Siam Journal on Numerical Analysis, 20 (1983).

[68] A. Tarantola, *Inversion of seismic reflection data in the acoustic approximation*, Geophysics, 49 (1984), pp. 1259–1266.

[69] ———, *Inverse Problem theory and methods for model parameter estimation*, Society for Industrial and Applied Mathematics, Philadelphia, 2005.

[70] J. Virieux and S. Operto, *An overview of full waveform inversion in exploration geophysics*, Society of Exploration Geophysics, 2010, p. ISBN: 9781560802266 (13).

[71] R. S. Wu and M. N. Toksöz, *Diffraction tomography and multisource holography applied to seismic imaging*, Geophysics, 52 (1987), pp. 11–25.

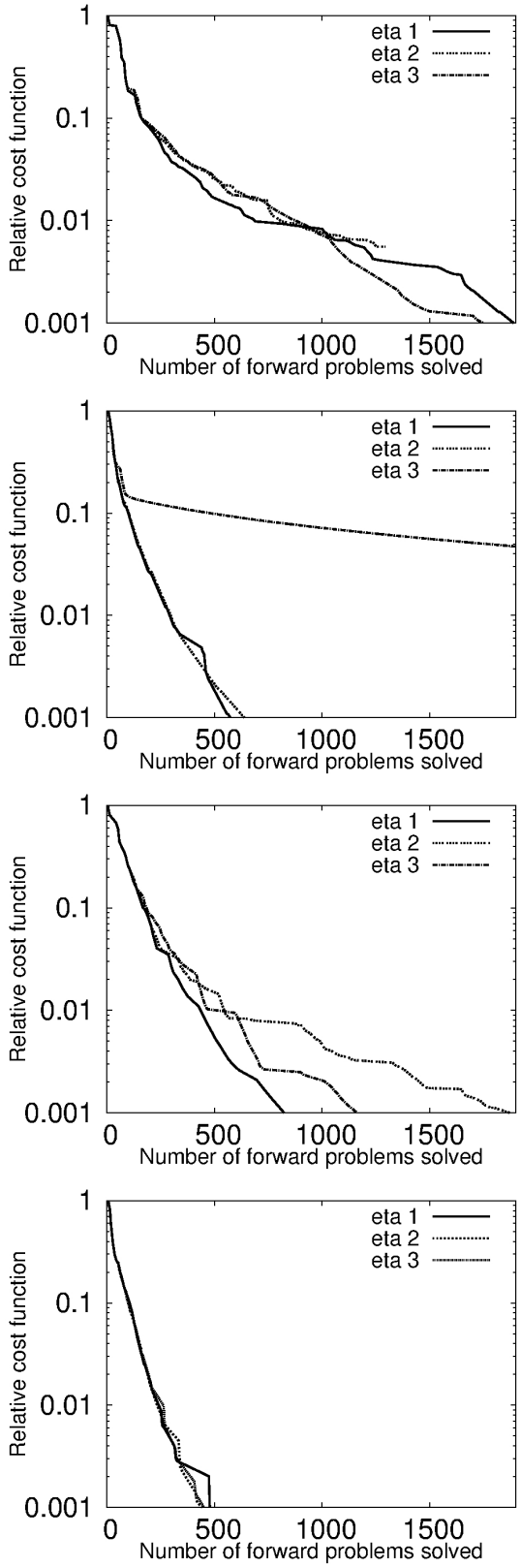

FIG. 4.2. *Comparison of the convergence of the truncated Newton method depending on the forcing term choice. From top to bottom : Newton method, Gauss-Newton method, outer l-BFGS preconditioned Newton method, outer l-BFGS preconditioned Gauss-Newton method.*

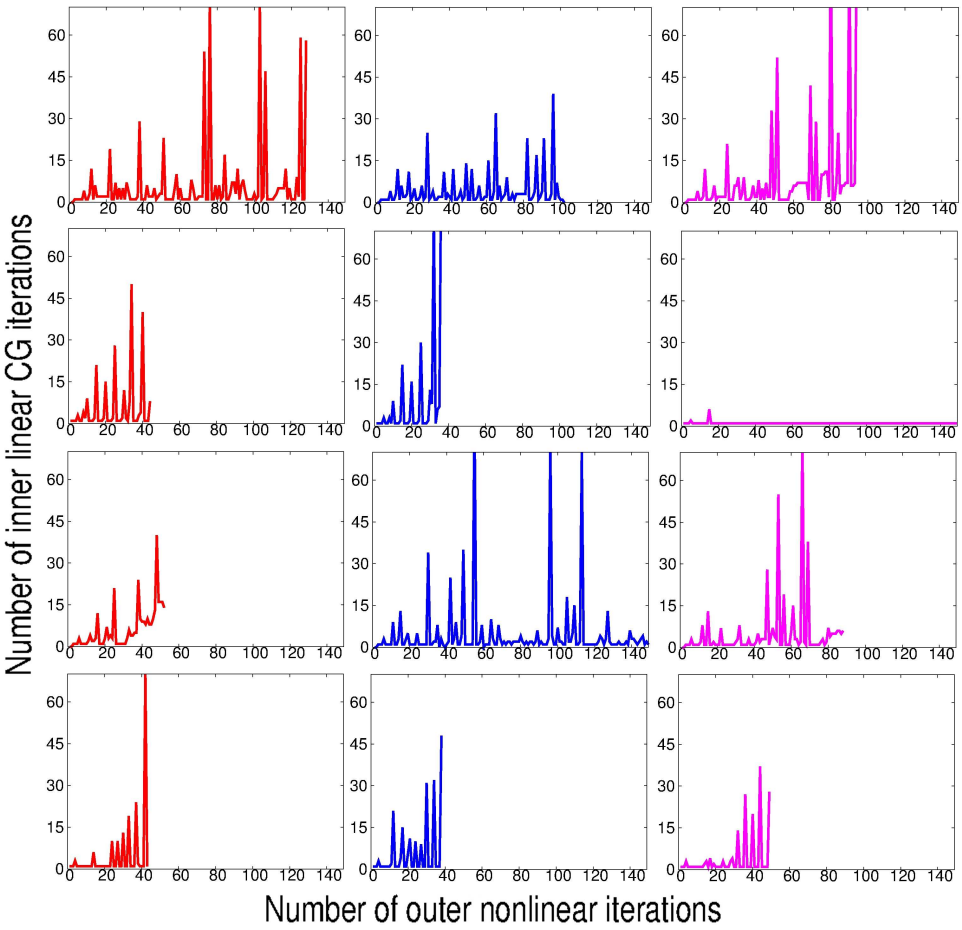

FIG. 4.3. *Number of inner CG iterations plotted against number of nonlinear iterations. Top row to bottom row: Newton method, Gauss-Newton method, outer l-BFGS preconditioned Newton method, outer l-BFGS preconditioned Gauss-Newton method. Left column:* $\eta_k = \eta_{k,1}$. *Middle column:* $\eta_k = \eta_{k,2}$. *right column:* $\eta_k = \eta_{k,3}$ *with* $a_1 = 0.95$ *and* $a_2 = 1$.

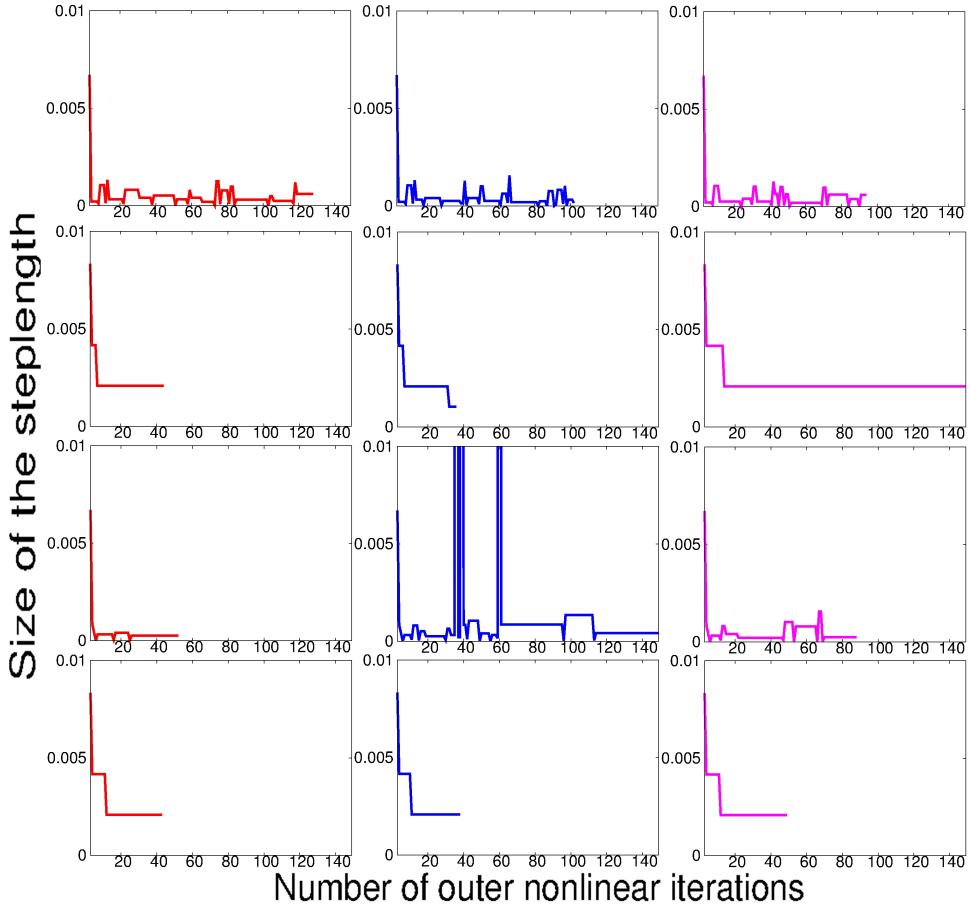

FIG. 4.4. *Size of the steplength α_k plotted against number nonlinear iterations. Top row to bottom row: Newton method, Gauss-Newton method, outer l-BFGS preconditioned Newton method, outer l-BFGS preconditioned Gauss-Newton method. Left column: $\eta_k = \eta_{k,1}$. Middle column: $\eta_k = \eta_{k,2}$. right column: $\eta_k = \eta_{k,3}$ with $a = 0.95$ and $a_2 = 1$.*

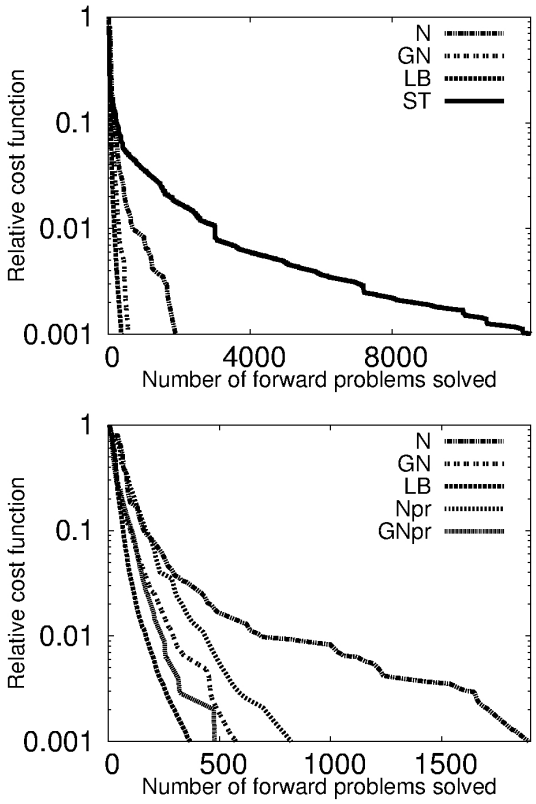

FIG. 4.5. *Convergence curves for the Marmousi II test case. N: Newton, GN: Gauss-Newton, LB: l-BFGS, ST: steepest-descent, Npr: outer preconditioned Newton, GNpr: outer preconditioned Gauss-Newton.*

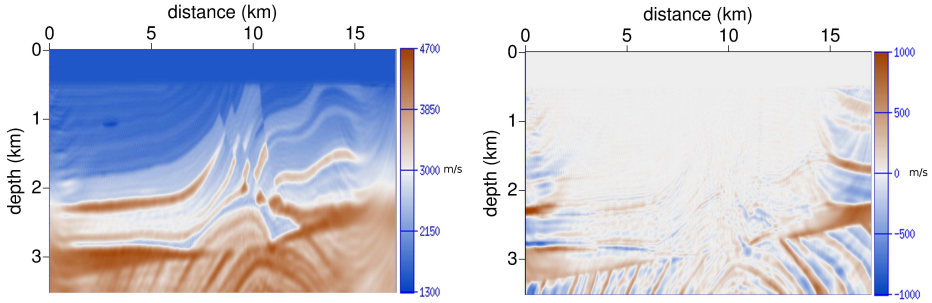

FIG. 4.6. *l-BFGS inversion results. Estimated model (left), difference with the exact model (right).*

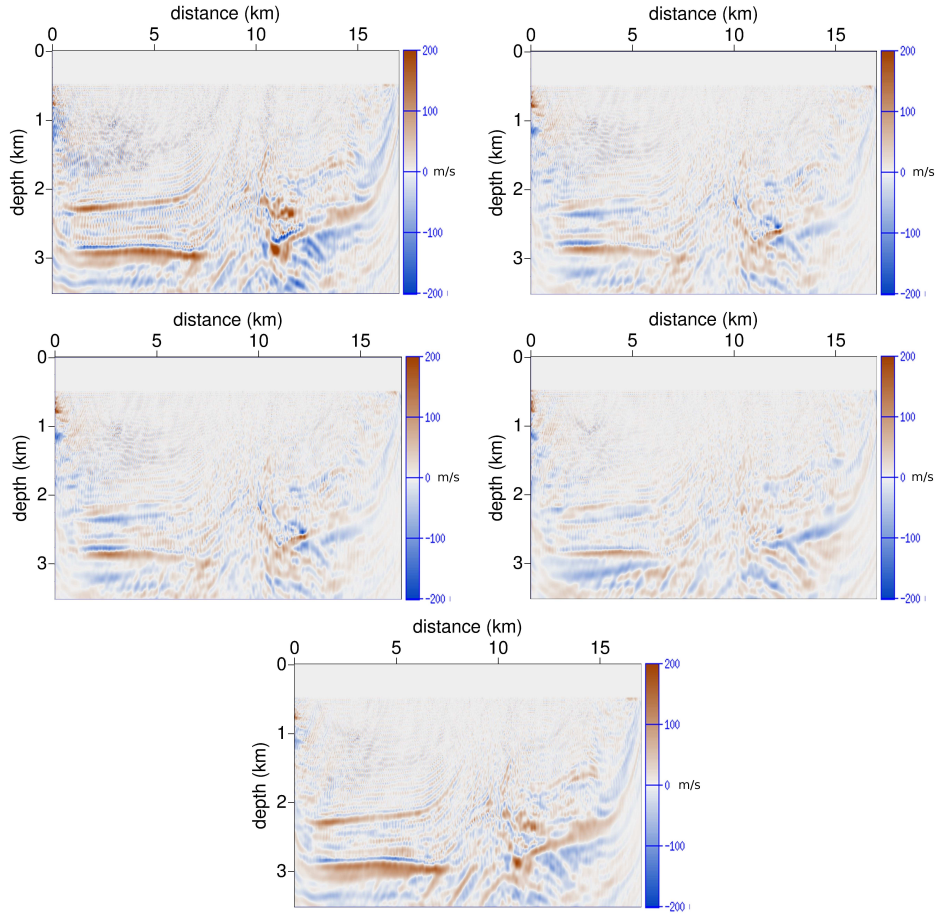

FIG. 4.7. *Difference between the l-BFGS estimated model and the other estimated models. Newton (top left), Gauss-Newton (top right), outer preconditioned Newton (middle left), outer preconditioned Gauss-Newton (middle right), steepest-descent (bottom)*

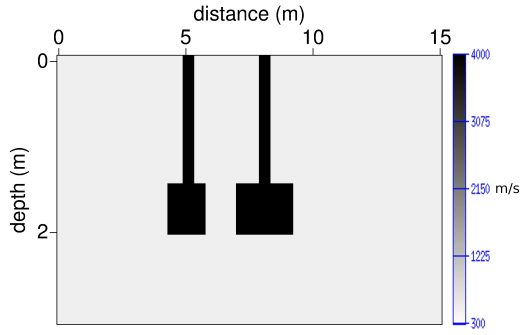

FIG. 4.8. *Exact pressure wave velocity model.*

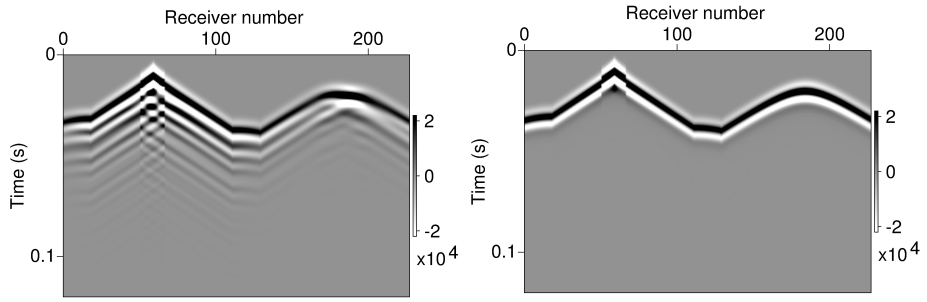

FIG. 4.9. *Dataset computed in the exact model (left), and in the initial homogeneous model (right).*

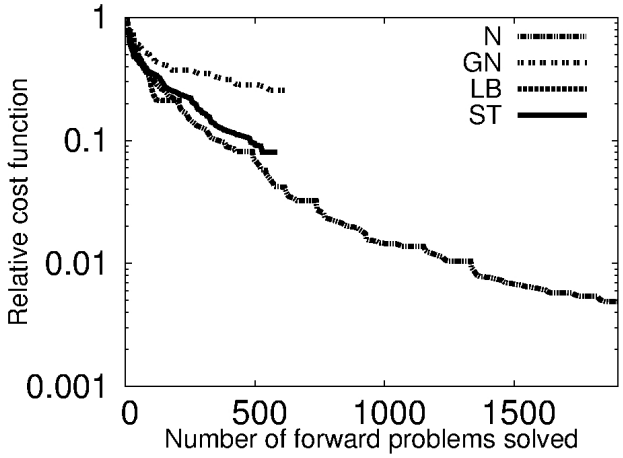

FIG. 4.10. *Convergence curves. N: Newton, GN: Gauss-Newton, LB: l-BFGS, ST: Steepest-descent.*

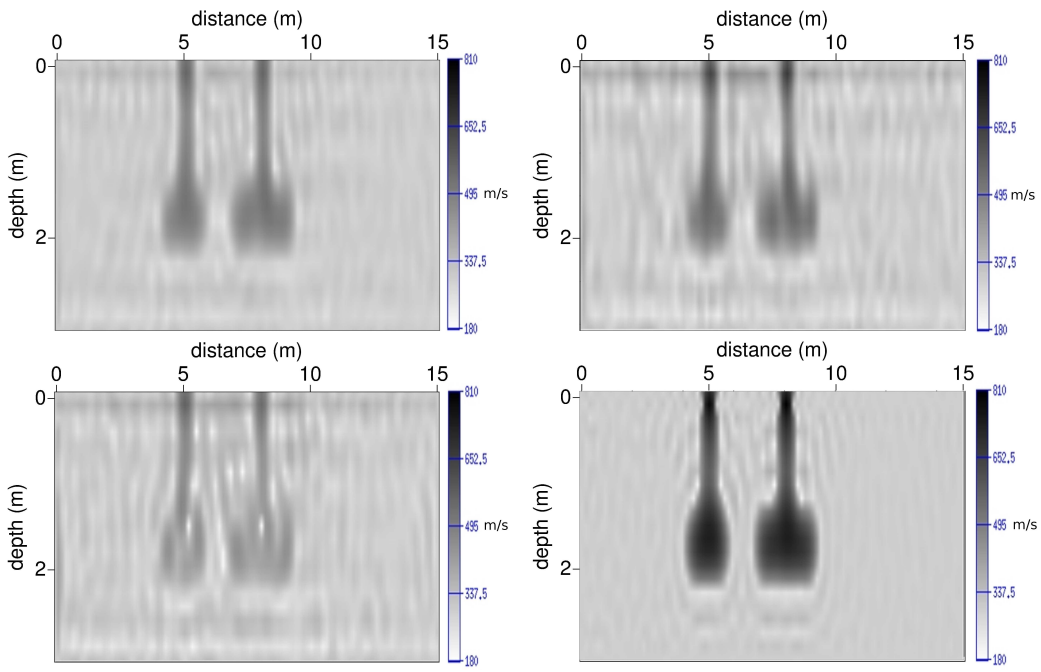

FIG. 4.11. *Pressure wave velocity estimatison, steepest-descent (top left) l-BFGS (top right), Gauss-Newton (bottom left), Newton (bottom right)*

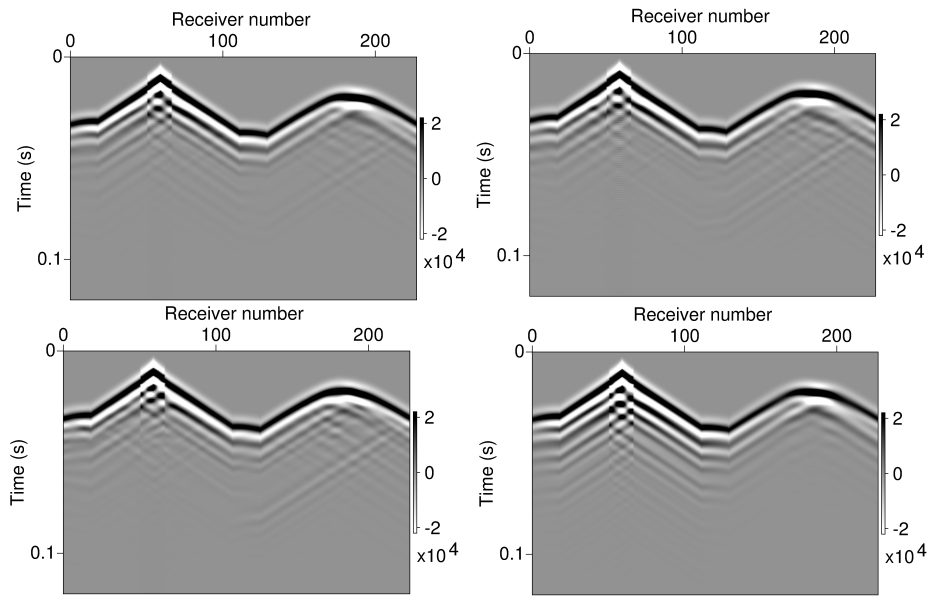

FIG. 4.12. *Dataset computed in the estimated models. Steepest-descent (top left), l-BFGS (top right), Gauss-Newton (bottom left), Newton (bottom right)*

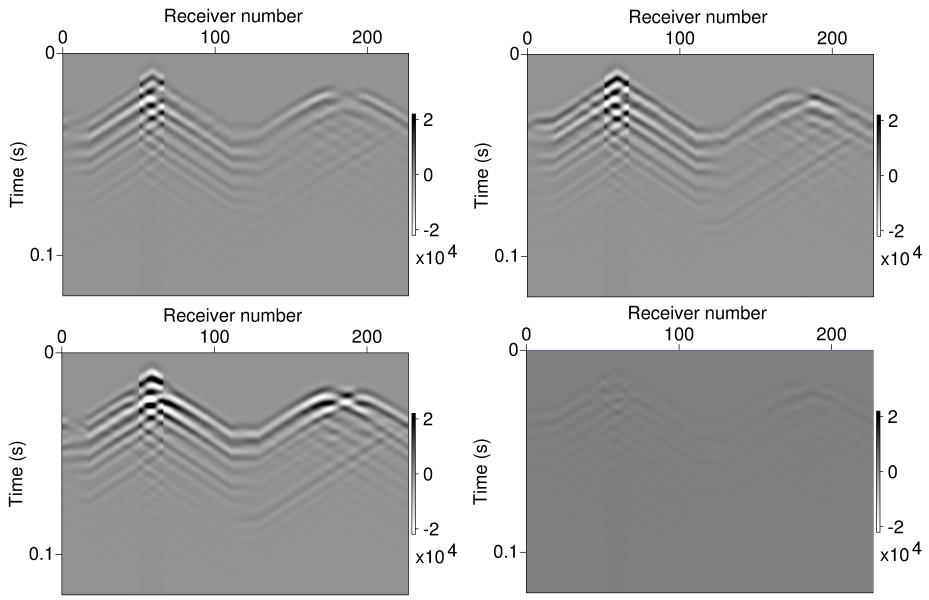

FIG. 4.13. *Residuals associated with the estimated models. Steepest-descent (top left), l-BFGS (top right), Gauss-Newton (bottom left), Newton (bottom right)*

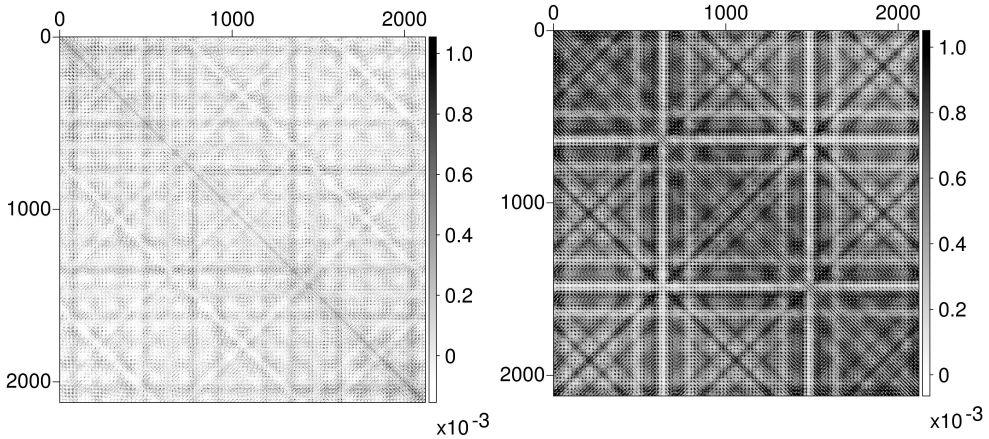

FIG. 4.14. *Hessian matrix (left), Gauss-Newton approximation (right)*

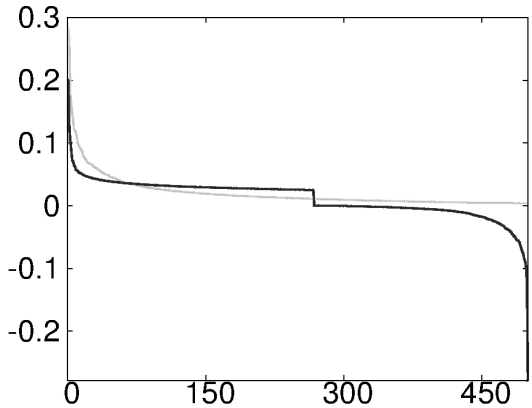

FIG. 4.15. 500 *largest eigenvalues of the Hessian operator (black) and its Gauss-Newton approximation (gray)*

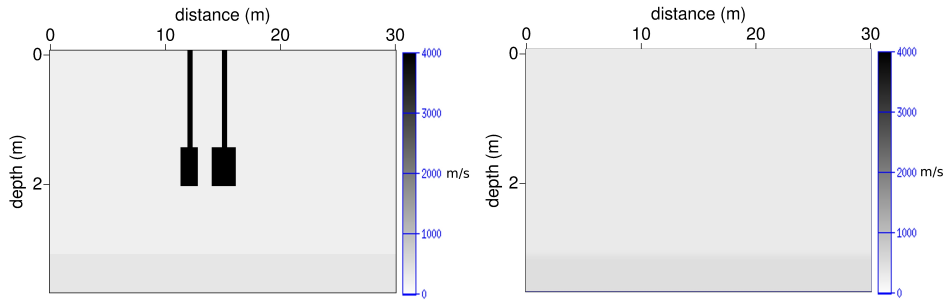

FIG. 4.16. *Larger exact pressure wave velocity model (left) and its corresponding initial guess (right)*

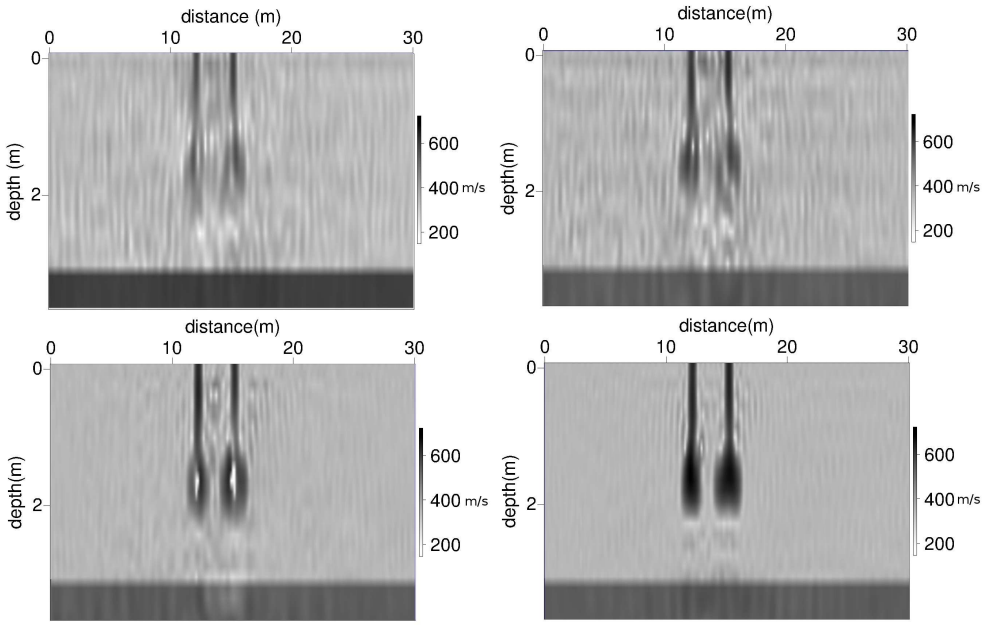

FIG. 4.17. *Wave velocity estimations. Steepest-descent (top left), l-BFGS (top right) , Gauss-Newton (bottom left), exact-Newton (bottom right).*

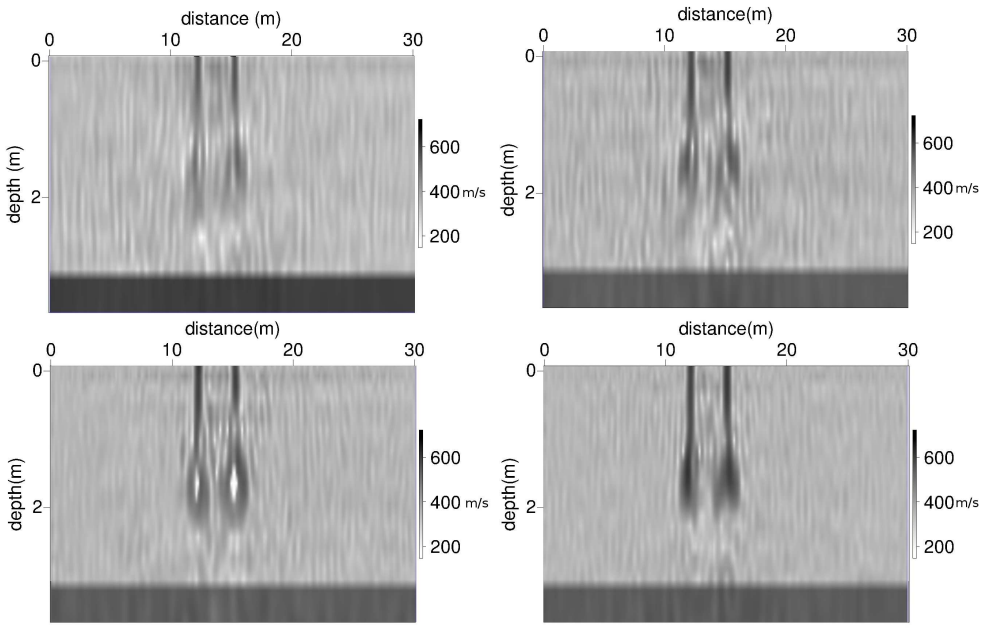

FIG. 4.18. *Wave velocity estimations with -3dB noisy data. Steepest-descent (top left), l-BFGS (top right) , Gauss-Newton (bottom left), exact-Newton (bottom right).*

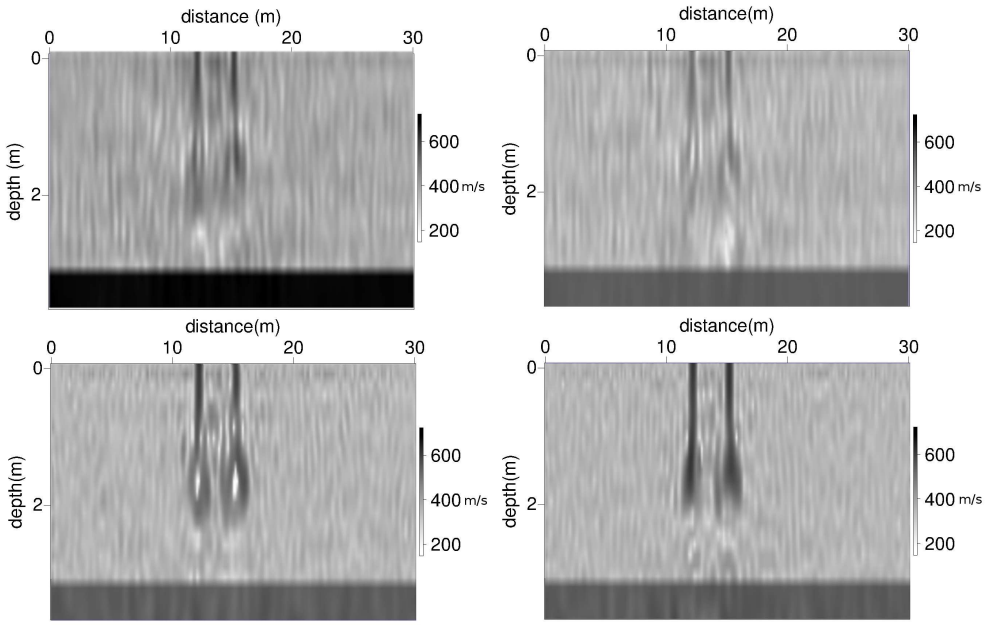

FIG. 4.19. *Wave velocity estimations with -9dB noisy data. Steepest-descent (top left), l-BFGS (top right) , Gauss-Newton (bottom left), exact-Newton (bottom right).*