UNIVERSITE DE NICE SOPHIA-ANTIPOLIS - UFR Sciences
Ecole Doctorale des Sciences Fondamentales et Appliquées

THESE

pour obtenir le titre de

Docteur en Sciences
de l'UNIVERSITE de Nice-Sophia Antipolis

Spécialité: **Sciences de l'Univers**

présentée et soutenue par
Hafedh BEN HADJ ALI

Three Dimensional Visco-Acoustic Frequency Domain Full Waveform Inversion

Thèse dirigée par Stéphane OPERTO et Jean VIRIEUX
soutenue le 01 Décembre 2009

Jury:

M. Henri CALANDRA	Chercheur Senior TOTAL	Examinateur
M. Michel DIETRICH	Professeur IFP	Examinateur
M. Luc GIRAUD	Professeur INRIA	Examinateur
M. Martin LACASSE	Chercheur Senior EXXON-MOBIL	Membre invité
M. Stéphane OPERTO	Chargé de recherche CNRS	Directeur de thèse
M. René-Edouard PLESSIX	Chercheur Senior SHELL	Rapporteur
M. Jean Pierre VILOTTE	Professeur IPGP	Rapporteur
M. Jean VIRIEUX	Professeur UJF-LGIT	Directeur de thèse

To my parents, my brother and sister, my family and my friends.

Résumé

En exploration sismique, il est primordial d'extraire des données enregistrées les paramètres physiques étudiés du sous-sol, typiquement les vitesses de propagation des ondes sismiques, afin de localiser correctement les réservoirs potentiels. Dans ce cadre, l'imagerie sismique est l'une des plus importantes étapes dans cette quête. Le processus d'imagerie a reposé pendant longtemps sur une décomposition par échelles: la première étape consiste à construire un modèle de vitesse des bas nombres d'ondes qui explique correctement la cinématique (la phase) du signal enregistré et la seconde à prendre en compte l'amplitude par un processus de sommation (migration) afin de détecter les contrastes de reflectivité (hauts nombres d'ondes).

Dans les années 80, une méthode d'imagerie quantitative, nommée *inversion des formes d'ondes* ou *inversion du champ d'onde complet*, a été proposée pour rassembler les deux étapes du processus d'imagerie au sein d'une approche intégrée. L'objectif de l'inversion des formes d'ondes est de construire simultanément tout le spectre des nombres d'ondes en exploitant l'ensemble des arrivées enregistrées par des dispositifs d'acquisition fournissant un large éclairage angulaire du milieu. La méthode est formulée sous la forme d'un problème d'otpmisation pour lequel les différences entre les données enregistrées aux récepteurs et les données modélisées sont minimisées au sens des moindres carrés.

Depuis quelques années et vu son potentiel, la méthode d'inversion des formes d'ondes a été un sujet de recherche trés étudié. Des problématiques liées au modèle initial et à la fréquence initiale utilisés dans l'inversion, à l'influence du bruit et au choix du critère de minimisation (autre norme ou autre fonctionnelle), à la reconstruction de plusieurs classes de paramètres, à l'introduction de phénomènes physiques plus réalistes comme l'atténuation, l'anisotropie et l'élasticité, et au passage du 2-D au 3-D et donc du coût numérique ont été abordés.

Dans ce contexte scientifique, l'objectif de cette thèse est de développer et d'évaluer une méthode d'inversion des formes d'ondes en domaine fréquentiel pour la reconstruction de modèles du sous-sol 3-D dans le cadre de l'approximation visco-acoustique où le milieu est paramétré par la vitesse de propagation des ondes de compression, la densité et l'atténuation.

L'efficacité et la faisabilité de l'inversion des formes d'ondes en trois dimensions est étroitement liée à l'efficacité du moteur de modélisation des ondes sismiques, et particulièrement dans une configuration de résolution multi-sources. En domaine fréquentiel, la modélisation des ondes à une fréquence donnée se ramène à la résolution d'un système d'équations linéaire creux dont le terme de droite est la source. L'équation d'onde harmonique a été discrétisée au sens des différences finies en utilisant des schémas numériques compactes permettrant ainsi de minimiser l'ordre et la bande passante numérique de la matrice.

La résolution directe par décomposition LU de la matrice et l'utilisation d'un outil massivement parallèle se sont initialement avérés trés efficaces pour résoudre des problèmes impliquant des modèles de petites dimensions (moins de 10 millions de noeuds de grille). Cette approche a néanmoins rapidement atteint ces limites intrinsèques liées aux complexités en temps de calcul et en mémoire de stockage des facteurs LU et à un faible pouvoir de scalabilité.

Une alternative intéressante a été développée pour réduire le coût mémoire de la modélisation et pour tirer davantage profit des grandes plateformes à mémoire distribuée: un solveur hybride

combinant un solveur direct et itératif est utilisé pour résoudre le système réduit du complément de Schur construit à partir d'une décomposition en sous-domaines du modèle du sous-sol. L'approche hybride m'a permis de traiter des modèles de plus grandes dimensions que l'approche directe au détriment de l'efficacité de la résolution multi-sources.

Les deux outils de modélisation, fondés sur un solveur direct et hybride, ont été interfacés au sein d'un même code d'inversion des formes d'ondes en domaine fréquentiel pour permettre l'utilisation de l'outil de modélisation le plus adapté au problème considéré (plateforme de calcul, nombre de sources, taille physique du milieu, fréquence inversée). D'autres approches fondées sur une résolution itérative du système ou explicite en temps ont été proposées par d'autres auteurs et présentent des coûts mémoire significativement inférieurs aux deux approches développées dans cette thèse. L'implémentation de ces approches dans le code d'inversion pourra s'envisager dans le futur.

Le code d'inversion des formes d'ondes en domaine fréquentiel a été développé en Fortran90 et se fonde sur le langage MPI pour le parallélisme. L'algorithme, fondé sur une méthode de gradients, est structuré sous forme d'inversions itératives successives de groupes de fréquences, le contenu haute fréquence de chaque groupe augmentant au cours de l'inversion. Le gradient de la fonction coût est calculé avec la méthode de l'état adjoint et nécessite la résolution de deux problèmes directs par source.

Au cours de l'inversion d'un groupe de fréquences, le pas de la grille de calcul peut être adaptée à la plus haute fréquence du groupe pour minimiser le coût numérique de l'inversion. La source a été implémentée de manière à pouvoir la positionner à une position arbitraire dans une grille cartésienne. Différents exemples synthétiques de complexité croissante ont permis de valider l'implémentation du code. L'exemple le plus réaliste a consisté à imager le modèle SEG/EAGE overthrust aprés l'inversion de trois fréquences (3, 5 et 7 Hz). Pour effectuer cette application, 48 noeuds bi-processeur dual-core disposant chacun de 8 *Goctets* de mémoire partagée ont été utilisés.

La dernière partie de ma thèse a eu pour objectif de réduire le coût de l'inversion, associé aux résolutions multi-sources du problème direct. L'approche développée est basée sur l'encodage des sources individuelles et sur leur assemblage par sommation pour réduire le nombre de sources impliquées dans la résolution du problème direct. Cette stratégie est plus particulièrement utile lorsque la complexité du problème direct est fortement dépendante du nombre de sources comme c'est le cas pour les solveurs hybrides, itératifs ou explicites en temps.

Différentes stratégies (sommation partielle ou totale des tirs, encodage aléatoire ou déterministe, influence du nombre de fréquences simultanément inversées) ont été évaluées. L'approche permet une réduction du coût de calcul mais révèle également une sensibilité au bruit dans les données, nécessitant d'augmenter le nombre de fréquences dans chaque groupe inversé et le nombre d'itérations de l'inversion.

La principale perspective de ces travaux est l'application à des données rélles pour cerner le potentiel et les limites de l'inversion, l'implémentation d'outils de modélisation complémentaire dans l'algorithme d'inversion et la reconstruction de plusieurs classes de paramètre tels que la densité et l'atténuation pour une meilleure prise en compte des amplitudes.

Abstract

In seismic exploration, it is crucial to extract from the recorded data the physical parameters of the subsurface, typically the seismic waves propagation velocity, in order to correctly locate the potential reservoirs. In this context, seismic imaging is one of most important steps in this quest. The imaging process has been for a long time based on a two-scales strategy: the first step consists in building a smooth (low wavenumbers) velocity model, which correctly explains the kinematics (phase) of the recorded signal, and the second step in taking into account the dynamics (amplitude) through a summation process (migration) to detect reflectivity contrasts (high wavenumbers).

In the eighties, a quantitative imaging method, called waveform inversion or full wavefield inversion, has been proposed to bring together the two stages of the imaging process in an integrated approach. The objective of waveform inversion is to build the whole spectrum of wavenumbers by exploiting all the recorded arrivals acquired by wide aperture (wide azimuth) acquisition geometries. The method is formulated as a least squares optimization problem which aims to minimize the differences between the recorded and the modeled data at the receivers.

During the last few years, the waveform inversion method has been a main research topic in the academic and industrial communities. Many issues related to the starting velocity model and the initial frequency involved in the inversion, the impact of noise and the choice of the minimization criterion, the reconstruction of multiple classes of parameters, the introduction of more realistic physical phenomena as attenuation, anisotropy and elasticity, and the transition from 2-D to 3-D, and therefore the issue of the computational cost, have been investigated and discussed.

In this context, the objective of this thesis is to investigate and to develop a waveform inversion approach in the frequency domain and within the visco-acoustic approximation for the reconstruction of a 3-D subsurface model where the model is parametrized by the P-waves velocity, density and attenuation.

The effectiveness and feasibility of waveform inversion in 3-D is closely related to the efficiency of the seismic waves modeling engine since thousands of sources are involved in 3-D. In the frequency domain, the modeling of seismic waves propagation at a given frequency reduces to solving a system of linear equations whose right hand side term is the source. The harmonic seismic wave equation has been discretized with finite differences using compact stencils which ensure minimizing the order and the bandwidth of the impedance matrix.

The direct resolution by an LU decomposition of the impdedance matrix and the use of a massively parallel solver was initially proved to be very efficient in solving problems involving small models (less than 10 million grid nodes). This approach has quickly reached its inherent limitations related to the computation time and LU factors memory storage complexities and its poor scalability.

An interesting alternative has been developed in order to reduce the memory cost of the direct solver and to take a greater advantage of the large distributed memory platforms: a hybrid solver combining the direct and iterative approaches is used to solve the Schur complement system obtained after a domain decomposition of the whole problem. The hybrid approach has allowed

to handle bigger models than the direct approach to the detriment of the effectiveness of the multi-sources resolution.

The two modeling tools based on direct and hybrid solvers have been interfaced within the same frequency domain waveform inversion software in order to allow using the best suited modeling tool to address a given problem (the available computing platform, the number of sources, the size of the problem, the inverted frequency). Other iterative and explicit time-marching approaches have been proposed by other authors and memory requirements are significantly lower than with the two approaches developed in this thesis. The implementation of these approaches in the inversion software can be envisaged in the future.

The frequency domain waveform inversion software has been written in Fortran90 and is based on the MPI standard for parallelism. The inversion algorithm, based on a gradient method, is structured as an iterative inversion of successive groups of frequencies, increasing the frequency band of each group during the inversion. The gradient of the misfit function is calculated using the adjoint state method which requires the resolution of two direct problems for each source.

During the inversion of a group of frequencies, the grid can be adapted to the highest inverted frequency in order to minimize the computational cost of the inversion. The source has been implemented so as to position it at an arbitrary grid node in the FD cartesian grid. Several synthetic examples of increasing complexity were performed in order to validate the software. The most realistic aims to image the model SEG / EAGE Overthrust after the inversion of three frequencies (3, 5 and 7 Hz). In order to perform this application, 48 dual-processor dual-core nodes, each with 8 Gbytes of shared memory have been used.

The last part of my thesis has aimed to reduce the cost of the inversion associated with the involved multi-sources resolution. The approach is based on encoding and assembling the individual sources in order to reduce the number of the sources involved in the inversion. This strategy is particularly useful when the complexity of the direct problem is strongly dependent on the number of sources as is the case of the hybrid, iterative and explicit time-marching solvers.

Different strategies (partial or total assembling, random or deterministic encoding, impact of the number of simultaneously inverted frequencies) have been discussed. The approach allows to reduce the computational cost, but also reveals to be sensitive with regards to noise and need to increase the number of inverted frequencies in each group and the number of the iterations during the inversion process.

The main perspective of this work is the application of the developed approach to real data study case in order to identify the potentialities and limitations of the waveform inversion method, the implementation of complementary modeling tools in the inversion algorithm and the inversion of multiple classes of parameters such as density and attenuation to better handle the amplitudes.

Remerciements

Contents

Chapter 1

General introduction to seismic imaging

1.1 Seismic imaging: from seismology to seismic exploration

At the end of the twentieth century, major discoveries in seismology have been carried out from the analysis of traveltimes of seismic waves (Earth structure, Inner/Outer Core, Mohorovičić and Gutenberg discontinuities, etc.). Transmission tomography has provided high-resolution upper mantle imaging.

Seismic exploration uses seismic waves to determine the geological structures of the subsurface and locate hydrocarbon reservoirs. A seismic acquisition experiment is performed by triggering an artificial controlled source (explosives, vibroseis, air gun, etc.) near the surface which initiates the seismic wave propagation in the subsurface. These waves are recorded by networks of receivers (geophones or hydrophones). During the propagation, the seismic wave undergoes at heterogeneities several complex physical phenomena: reflection, refraction and diffraction. This confers to seismic data nonlinear properties. The processing of recorded data allows to determine some physical properties of the Earth. *Seismic imaging is the process through which seismograms recorded on the Earth's surface are mapped into representations of its interior properties* (Scales, 1995). Historically, the most common parameter in seismic exploration imaging is wave propagation velocity and the wave propagation often restricts to the acoustic approximation.

The geometry of the seismic acquisition defines the nature of the information contained in the data. Indeed, wide aperture/azimuth and global offset acquisitions are necessary to record the diving waves in order to appropriately image complex targets such as salt dome dipping flanks. However, dense wide azimuth acquisition have often presented financial and deployment challenges. In O&G exploration industry, dense multifold seismic reflection acquisition is the standard acquisition geometry. Thus, it implies that the data processing is mainly based on reflected waves.

1.2 Conventional imaging procedure

Due to the small range of offsets of seismic reflection acquisition geometries and the frequency content of the source, the seismic imaging process have suffered from a lack of sensitivity to intermediate wavelengths (Jannane et al., 1989). Thus, the seismic imaging procedure has been splitted into two steps.

A first step consists in using only the kinematics information (traveltime) of the data in order to recover the large wavelengths (low wavenumbers) of the model. There are diverse methods to recover these latter. The most frequently used technique is based on the derivation of the velocity model by stacking velocity analysis in which the velocities are defined from the normal moveout (NMO) of the reflection hyperbola (Yilmaz, 2001). The method is limited to 1-D stratified velocity models. Traveltime tomography techniques do not suffer from such limitation and can handle laterally varying velocity models. It consists in solving an inverse problem, generally linear, that seeks to minimize the mismatch between predicted and recorded traveltimes (Bishop et al., 1985). Observed traveltimes are typically picked manually and coherent events such as reflected or refracted waves are identified (Phillips and Fehler, 1991). The computation of predicted traveltimes relies generally on ray tracing techniques (Červený et al., 1977; Zelt and Smith, 1992) or finite differences solver of the Eikonal equation (Vidale, 1988, 1990; Podvin and Lecomte, 1991). First arrival traveltime tomography (FATT) accounts only for first arrival traveltimes (diving/refracted events). This implies that wide aperture data are required for a good illumination of deep targets. In addition to mentioned drawbacks of these techniques, manual picking can be a tough and cumbersome job. Promising methods that avoid hand picking such as stereotomography (Billette, 1998; Lambaré, 2008) and migration velocity analysis (Yilmaz and Chambers, 1984; Symes and Carazzone, 1991) have been proposed. Stereotomography (Billette et al., 2003) approach is based on an automatic pick of locally coherent events. Migration velocity analysis (MVA) (Symes and Carazzone, 1991; Liu and Bleistein, 1995; Docherty et al., 1997) relies on the optimization of a defined coherency condition in the migrated Common Image Gathers (CIG) domain. This approach is very appreciated and used since it optimizes the *migration* through an iterative MVA-migration procedure.

A second step takes account of the amplitude through different *migration* algorithms to recover the short wavelengths (high wavenumbers) of the model. By short wavelengths we mainly mean reflectors. Two main families of migration methods can be distinguished: integral and wavefield-continuation methods (Biondi, 2006). For each category, several approaches were proposed and differ with regards to the choice of time or depth formulation, applied to stacked (*poststack*) or unstacked data (*prestack* (Judson et al., 1980; Schultz and Sherwood, 1980), more computationally expensive), to the approximation of the wave equation which is used, ...

Integral methods rely on summation surfaces (diffraction hyperboloids) principle. Kirchhoff migration is probably the most popular algorithm of this category (Berkhout and Wapenaar, 1989; Docherty, 1991; Audebert et al., 1997; Biondi, 2001). For many years, poststack time migration were preferred due to its efficiency and robustness provided that the velocity model is not complex. Nowadays, prestack depth migration is commonly used in industry since exploration targets are more complex (for example, dipping flanks of salt domes). Pioneering algorithms have used ray tracing. Then, more robust one-way and full wave equation have been used to overcome ray tracing limitations.

Wavefield-continuation methods (Claerbout, 1985) are known to be more efficient and computationally expensive than Kirchhoff migration (Lambaré et al., 2002; Biondi, 2006). Nonetheless, they are more sensitive to acquisition regularity. These methods are based on the Exploding Reflectors Principle (Claerbout, 1976). The first algorithms, called downward-continuation migration algorithms, have been implemented in the frequency domain and based on the one-way wave equation: phase-shift ($f-k$) migration (Stolt, 1978; Gazdag, 1978), source-receiver migration (Biondi, 2006), shot-gather migration (Reshef and Kosloff, 1986), etc. Nowadays, reverse time migration (RTM) (Claerbout, 1985; Chen and McMechan, 1992; Zhu and Lines, 1996) is the method of choice for industrial applications. RTM is based on full seismic wave modeling and the imaging condition reduces to a summation of the cross-correlations between the shot wavefields and their respective receivers backpropagated wavefields.

The two-step imaging strategy has been successfully applied to many areas of seismic exploration. Nonetheless, it has shown limited performances for complex exploration targets such as subsalt and subbasalt targets and foothills areas imaging.

1.3 Waveform inversion

Due to complexity of nowadays seismic exploration targets such as in deep offshore, the acquisition geometry is moving from narrow to wide azimuth acquisitions. The two-steps standard imaging procedure based exclusively on the pre-critical reflected events is then questioned. Waveform inversion, called also full waveform inversion (FWI), is an alternative to overcome the limitations of the two-steps standard imaging procedure. Contrary to the two-steps procedure, FWI is a quantitative seismic imaging method that takes into account for the entire waveform information (phase (traveltime,kinematics) and amplitude (dynamics)). If appropriate frequency bandwidth sources and acquisition system are used, it is potentially able to recover of all arrivals range of wavenumbers including the intermediate wavenumbers (Mora, 1989) and to provide high-resolution imaging results.

In the eighties, the problem was formulated, as in tomography, as a least squares optimization problem (Tarantola, 1984a; Lailly, 1984). The resolution of this inverse problem as a local optimization problem seeks to minimize the misfit between the recorded and predicted data (seismograms while traveltimes in tomography). Tarantola (1984a) and Lailly (1984) have formulated the problem in the time domain. Next, other formulations in the frequency domain (Pratt and Worthington, 1990; Pratt, 1990) and in the Laplace domain (Shin and Cha, 2008) have been proposed.

During the pioneering attempts, FWI had difficulty to impose as an efficient imaging tool. Indeed, FWI shows to be very computer demanding, even in 2-D, and to suffer from intrinsic ill-posedness and nonlinearity. In fact, the first applications were restricted to short offset reflection acquisition geometries which are not adapted to image low and intermediate wavenumbers (Gauthier et al., 1986; Jannane et al., 1989). In addition, since low frequencies have better properties with regards to problem nonlinearity than high frequencies, an appropriate initial couple (low frequency, starting model) is compulsory for the FWI applicability. Applications of FWI on wide aperture or large offset data have highlighted this statement and demonstrated FWI efficiency and efficacy to lead to high resolved velocity models (Mora, 1987, 1988; Pratt and Worthington, 1990; Pratt et al., 1996a).

It was also highlighted in the pioneering works of Pratt and collaborators (Pratt and

Worthington, 1990; Pratt, 1990), and next by Bunks et al. (1995), that it is necessary to first tackle low frequencies (which mitigate nonlinearity as explained in the previous paragraph) and include progressively higher frequencies, therefore define a multi-scale/multi-resolution imaging strategy. In addition, in order to reduce the computational cost of frequency domain FWI, Sirgue and Pratt (2004) has proposed to decimate wavenumbers redundancy by limiting the inversion to a few discrete frequencies, which has conferred to the frequency domain FWI formulation a privileged position.

During the last two decades, several 2-D FWI applications on synthetic data (Pratt and Goulty, 1991; Pratt et al., 1996b; Brenders and Pratt, 2006; Brossier et al., 2008; Sears et al., 2008) and real data case studies (Dessa et al., 2004; Ravaut et al., 2004; Operto et al., 2006b; Gao et al., 2006) have been presented.

1.4 From 2-D to 3-D

During the last years, efforts have been concentrated on including more physics in the modeling and imaging such as anisotropy, attenuation, elasticity, multi parameter inversion and on evolving from 2-D to 3-D. The transition from 2-D to 3-D requires a very efficient modeling of the predicted data and a clever strategy to deal with huge data set.

Many 3-D frequency domain FWI tools using different seismic wave modeling approaches have been developed.

Sirgue et al. (2007b) use a time domain modeling and compute monochromatic responses through a discrete Fourier transform. Sirgue et al. (2007a) have studied the impact of the acquisition geometry on FWI application on synthetic SEG/EAGE overthrust model and have highlighted the importance of wide azimuth data to obtain high-resolution images. Sirgue et al. (2009) have applied frequency domain FWI on Valhall (Munns, 1985; Kommedal et al., 2004) real data set and have shown the imaging resolution improvement with regards to reflection tomography velocity model which was used as an initial model for FWI.

Warner et al. (2007) and Erlangga and Herrmann (2008); Plessix (2007) have recommended to use an iterative solver to solve the Helmholtz system. Plessix and Perkins (2009); Plessix (2009) have presented an application of anisotropic FWI on deep-water OBS data recorded in the Golf of Mexico.

Operto et al. (2007) have assessed the classes of problems that ca be tackled with a massively parallel direct solver, commonly used for 2-D problems to take advantage of its multi-RHS resolution efficiency and Ben-Hadj-Ali et al. (2008) have shown the efficiency of FWI based on direct solver on the medium size problems with an application on the synthetic SEG/EAGE overthrust velocity model.

Vigh and Starr (2008) have proposed a time domain prestack plane-wave full waveform inversion software which performs the seismic wave modeling in the time domain.

However, due to the computational limitations of the full seismic wave modeling in 3-D, the applications restrict to low frequencies ($\approx 7\ Hz$). Therefore, these computational limitations question the position of FWI in the imaging procedure flowchart. FWI has been proposed to replace the two-step "velocity model building/migration" imaging procedure. In 3-D, FWI is still investigated as an alternative approach to improve the velocity macro-model which would be used for migration.

In addition, many approaches have been proposed to reduce the huge amount of data involved in the inversion procedure.

Joint simultaneous-shot and phase encoding has been investigated by Krebs et al. (2009). Simultaneous-shot technique consists in gathering many sources in one assemblage and processing the assemblages as shots. This introduces cross-talks between the different sources in the same assemblage. In order to reduce these cross-talks, phase shifts between sources are applied. This technique is called phase encoding. The plane-wave approach is a particular parameterization of phase encoding.

Herrmann et al. (2009) have promoted the compressive censing (CS) method and highlighted the connection with the joint simultaneous-shot and phase encoding strategy. The CS method suggests that compressible signal can be recovered from sub-Nyquist sampling by solving a program promoting sparsity and exploiting a specific transform-domain sparsity, such as the curvelets transform. Herrmann et al. (2009) have demonstrated the equivalence between applying the CS operator on the full data set and sampling compressively the shot wavefields, similarly to the simultaneous-shot strategy. The sparsifying transform have to be appropriately defined in order to retrieve the single source wavefields. The source wavefields are reconstructed by solving a nonlinear optimization problem.

1.5 Objectives of the thesis

When I started to work on my thesis, our principle goal has been to investigate the feasibility of 3-D visco-acoustic frequency domain FWI and develop an efficient software. At that moment, only few research teams have been working on this topic. Today, most of seismic exploration companies have their own softwares. FWI has without contest become a topical problem.

1.6 Outline

This manuscript is divided into three parts.

Part I is devoted to the *modeling* or *forward problem*. In the **chapter 2**, I am interested in expanding the 3-D finite differences stencil used to discretize the continuous Helmholtz wave equation (Operto et al., 2007). Staggered grids, parsimonious strategy, mixed stencils, anti-lumped mass and perfectly matched layers (PML) absorbing boundary conditions are reviewed. The stencil has been validated on homogeneous and highly heterogeneous velocity models. The **chapter 3** is dedicated to the presentation of the adopted modeling solvers: the direct solver (MUMPS) and the hybrid direct-iterative solver. Theoretical aspects as well as complexity and scalability analysis are expounded. A comparative analysis of the adopted solvers and the iterative and time domain solvers concludes the chapter.

Part II is devoted to the *inverse problem*. In the **chapter 4**, a brief overview on the *line search methods* used to solve a local optimization problem is presented. In the **chapter 5**, I am interested in full waveform inversion and especially the frequency domain formulation. The least squares formalism is reviewed. Advantages and drawbacks of the frequency domain formulation are enumerated and discussed. The parallel implementation of the FWI algorithm is presented in the **chapter 6**. In the **chapter 7**, the FWI software has been validated on simple and complex velocity models.

Part III is dedicated to the *simultaneous-shot & phase encoding* techniques. In the **chapter 8**, the related principles are introduced. Different strategies of partial or full simultaneous-shot with several choices of phase encoding are proposed and studied. The different strategies have been validated on the SEG/EAGE overthrust velocity model for 2-D and 3-D configurations.

Part I

3-D seismic wave modeling

Chapter 2

3-D finite differences frequency domain visco-acoustic wave modeling

Seismic imaging with Full Waveform Inversion (FWI) requires an accurate and efficient wavefield modeling engine, especially in 3-D heterogeneous media. To solve numerically the wave equation and get the wavefield solution, a class of methods called volumetric methods is widespread. The most commonly used methods are finite elements, finite differences, finite volumes and discontinuous Galerkin methods. In the seismic community, finite differences method are popular because of its simplicity and robustness.

A 3-D parsimonious mixed-grid staggered-grid finite differences stencil for 3-D Helmholtz equation was developed by Operto et al. (2007). Only four grid nodes per wavelength are required to accurately model the wavefield in a complex medium.

In this chapter, I will briefly review the methodology adopted in Operto et al. (2007). More details can be found in the paper itself. I will then validate the stencil precision on homogeneous and complex media.

2.1 3-D wave propagation equation in visco-acoustic media

In this section, I shall derive the well known partial derivatives equation (PDE), Helmholtz equation. It can be obtained through an acoustic approximation of the elastic system introduced in Aki and Richards (1980, chapter 2). An other approach to derive the Helmholtz equation can be found in Erlangga (2005).

2.1.1 First-order hyperbolic system in velocity-pressure

Combining the equation of motion and the Hooke's law (Aki and Richards, 1980), the elasto-dynamic system in an isotropic heterogeneous media can be expressed as

$$
\begin{aligned}
\frac{\partial \sigma_{xx}(x,y,z,t)}{\partial t} &= (\lambda(x,y,z)+2\mu(x,y,z))\frac{\partial v_x(x,y,z,t)}{\partial x} + \lambda(x,y,z)\{\frac{\partial v_y(x,y,z,t)}{\partial y} \\
&+ \frac{\partial v_z(x,y,z,t)}{\partial z}\} \\
\frac{\partial \sigma_{yy}(x,y,z,t)}{\partial t} &= (\lambda(x,y,z)+2\mu(x,y,z))\frac{\partial v_y(x,y,z,t)}{\partial y} + \lambda(x,y,z)\{\frac{\partial v_x(x,y,z,t)}{\partial x} \\
&+ \frac{\partial v_z(x,y,z,t)}{\partial z}\} \\
\frac{\partial \sigma_{zz}(x,y,z,t)}{\partial t} &= (\lambda(x,y,z)+2\mu(x,y,z))\frac{\partial v_z(x,y,z,t)}{\partial z} + \lambda(x,y,z)\{\frac{\partial v_x(x,y,z,t)}{\partial x} \\
&+ \frac{\partial v_y(x,y,z,t)}{\partial y}\} \\
\frac{\partial \sigma_{xy}(x,y,z,t)}{\partial t} &= \mu(x,y,z)\{\frac{\partial v_x(x,y,z,t)}{\partial y} + \frac{\partial v_y(x,y,z,t)}{\partial x}\} \\
\frac{\partial \sigma_{xz}(x,y,z,t)}{\partial t} &= \mu(x,y,z)\{\frac{\partial v_x(x,y,z,t)}{\partial z} + \frac{\partial v_z(x,y,z,t)}{\partial x}\} \\
\frac{\partial \sigma_{yz}(x,y,z,t)}{\partial t} &= \mu(x,y,z)\{\frac{\partial v_y(x,y,z,t)}{\partial z} + \frac{\partial v_z(x,y,z,t)}{\partial y}\} \\
\frac{\partial v_x(x,y,z,t)}{\partial t} &= \frac{1}{\rho(x,y,z)}\{\frac{\partial \sigma_{xx}(x,y,z,t)}{\partial x} + \frac{\partial \sigma_{xy}(x,y,z,t)}{\partial y} + \frac{\partial \sigma_{xz}(x,y,z,t)}{\partial z}\} \\
\frac{\partial v_y(x,y,z,t)}{\partial t} &= \frac{1}{\rho(x,y,z)}\{\frac{\partial \sigma_{xy}(x,y,z,t)}{\partial x} + \frac{\partial \sigma_{yy}(x,y,z,t)}{\partial y} + \frac{\partial \sigma_{yz}(x,y,z,t)}{\partial z}\} \\
\frac{\partial v_z(x,y,z,t)}{\partial t} &= \frac{1}{\rho(x,y,z)}\{\frac{\partial \sigma_{xz}(x,y,z,t)}{\partial x} + \frac{\partial \sigma_{yz}(x,y,z,t)}{\partial y} + \frac{\partial \sigma_{zz}(x,y,z,t)}{\partial z}\},
\end{aligned}
\tag{2.1}
$$

where wavefields $v_x(x,y,z,t)$, $v_y(x,y,z,t)$ and $v_z(x,y,z,t)$ are the particle velocity components, $\sigma_{ij}(x,y,z,t), i,j = x,y,z$ are the stress components while $\rho(x,y,z)$ is the density, $\lambda(x,y,z)$ and $\mu(x,y,z)$ are Lamé parameters.

To derive the acoustic wave equation approximation, we consider that the shear modulus is zero ($\mu(x,y,z) = 0$) and only normal components of stress are non-zeros.

System (2.1) reduces to

$$
\begin{aligned}
\frac{\partial \sigma_{xx}(x,y,z,t)}{\partial t} &= \lambda(x,y,z)\{\frac{\partial v_x(x,y,z,t)}{\partial x} + \frac{\partial v_y(x,y,z,t)}{\partial y} + \frac{\partial v_z(x,y,z,t)}{\partial z}\} \\
\frac{\partial \sigma_{yy}(x,y,z,t)}{\partial t} &= \lambda(x,y,z)\{\frac{\partial v_y(x,y,z,t)}{\partial y} + \frac{\partial v_x(x,y,z,t)}{\partial x} + \frac{\partial v_z(x,y,z,t)}{\partial z}\} \\
\frac{\partial \sigma_{zz}(x,y,z,t)}{\partial t} &= \lambda(x,y,z)\{\frac{\partial v_z(x,y,z,t)}{\partial z} + \frac{\partial v_x(x,y,z,t)}{\partial x} + \frac{\partial v_y(x,y,z,t)}{\partial y}\} \\
\frac{\partial v_x(x,y,z,t)}{\partial t} &= \frac{1}{\rho(x,y,z)} \frac{\partial \sigma_{xx}(x,y,z,t)}{\partial x} \\
\frac{\partial v_y(x,y,z,t)}{\partial t} &= \frac{1}{\rho(x,y,z)} \frac{\partial \sigma_{yy}(x,y,z,t)}{\partial y} \\
\frac{\partial v_z(x,y,z,t)}{\partial t} &= \frac{1}{\rho(x,y,z)} \frac{\partial \sigma_{zz}(x,y,z,t)}{\partial z}.
\end{aligned}
\tag{2.2}
$$

The three first equations of system (2.2) imply that $\frac{\partial \sigma_{xx}(x,y,z,t)}{\partial t} = \frac{\partial \sigma_{yy}(x,y,z,t)}{\partial t} = \frac{\partial \sigma_{zz}(x,y,z,t)}{\partial t}$. Let us now define the pressure field $p(x,y,z,t)$ by

$$
p(x,y,z,t) = \frac{\sigma_{xx}(x,y,z,t) + \sigma_{yy}(x,y,z,t) + \sigma_{zz}(x,y,z,t)}{3}.
\tag{2.3}
$$

We combine the three first equations and obtain the first-order hyperbolic system below,

$$
\begin{aligned}
\frac{\partial p(x,y,z,t)}{\partial t} &= \kappa(x,y,z)(\frac{\partial v_x(x,y,z,t)}{\partial x} + \frac{\partial v_y(x,y,z,t)}{\partial y} + \frac{\partial v_z(x,y,z,t)}{\partial z}) \\
&+ s(x,y,z,t) \\
\frac{\partial v_x(x,y,z,t)}{\partial t} &= b(x,y,z) \frac{\partial p(x,y,z,t)}{\partial x} \\
\frac{\partial v_y(x,y,z,t)}{\partial t} &= b(x,y,z) \frac{\partial p(x,y,z,t)}{\partial x} \\
\frac{\partial v_z(x,y,z,t)}{\partial t} &= b(x,y,z) \cdot \frac{\partial p(x,y,z,t)}{\partial z},
\end{aligned}
\tag{2.4}
$$

where $p(x,y,z,t)$ is the pressure, $\kappa(x,y,z)$ the bulk modulus and $b(x,y,z) = \frac{1}{\rho(x,y,z)}$ is the buoyancy. The pressure source term $s(x,y,z,t)$ has been introduced in the first equation of the system.

In the Fourier domain, the previous system of equations reads

$$\begin{aligned}
\frac{-\iota\omega}{\kappa(x,y,z)}p(x,y,z,\omega) &= \frac{\partial v_x(x,y,z,\omega)}{\partial x} + \frac{\partial v_y(x,y,z,\omega)}{\partial y} + \frac{\partial v_z(x,y,z,\omega)}{\partial z} \\
&+ s(x,y,z,\omega) \\
v_x(x,y,z,\omega) &= \frac{\iota b(x,y,z)}{\omega}\frac{\partial p(x,y,z,\omega)}{\partial x} \\
v_y(x,y,z,\omega) &= \frac{\iota b(x,y,z)}{\omega}\frac{\partial p(x,y,z,\omega)}{\partial y} \\
v_z(x,y,z,\omega) &= \frac{\iota b(x,y,z)}{\omega}\frac{\partial p(x,y,z,\omega)}{\partial z},
\end{aligned} \tag{2.5}$$

where $\iota = \sqrt{-1}$. This system allows the computation of the pressure and the particle velocity fields in the frequency domain.

2.1.2 Second-order elliptic system in pressure

The expressions of the particle velocity fields (the last three equations of system (2.5)) are injected into the first equation of system (2.5) and lead to the second-order elliptic wave equation (in pressure), also known as the Helmholtz equation,

$$\begin{aligned}
\left[\frac{\omega^2}{\kappa(x,y,z)} + \frac{\partial}{\partial x}b(x,y,z)\frac{\partial}{\partial x} + \frac{\partial}{\partial y}b(x,y,z)\frac{\partial}{\partial y} + \frac{\partial}{\partial z}b(x,y,z)\frac{\partial}{\partial z}\right]. \\
p(x,y,z,\omega) = s(x,y,z,\omega).
\end{aligned} \tag{2.6}$$

The equation can be expressed in a compact form as,

$$\mathcal{A}(x,y,z,\omega).p(x,y,z,\omega) = s(x,y,z,\omega), \tag{2.7}$$

where $\mathcal{A}$ denotes the differential operator.

2.2 Absorbing boundary conditions (ABC)

The systems of equations (2.4) and (2.5) define the wave propagation in an infinite medium, respectively in the time and frequency domains. However, the wavefield is numerically modeled in a finite domain. Therefore, reflections from the computation domain edges need to be attenuated or suppressed. Clayton and Engquist (1977) have proposed the 45° paraxial approximation. Cerjan et al. (1985) have introduced the sponge-like absorbing conditions. The computation domain is augmented with sponge layers. The wave propagation is correctly modeled in the computation domain. However in the sponge layers the wavefield is progressively attenuated through a damping function. Berenger (1994) improved the Cerjan sponge-like absorbing conditions and proposed the perfectly matched layers (PML) approach. The PML conditions only attenuate the normal component of the wavefield while the whole wavefield is attenuated in the Cerjan's approach.

To implement the PML conditions in the system of equations (2.4), the pressure wavefield needs to be splitted. System of equations (2.4) becomes

$$\begin{aligned}
\frac{\partial p_x(x,y,z,t)}{\partial t} + \gamma_x(x)p_x(x,y,z,t) &= \kappa(x,y,z)\frac{\partial v_x(x,y,z,t)}{\partial x} + s(x,y,z,t)\\
\frac{\partial p_y(x,y,z,t)}{\partial t} + \gamma_y(y)p_y(x,y,z,t) &= \kappa(x,y,z)\frac{\partial v_y(x,y,z,t)}{\partial x}\\
\frac{\partial p_z(x,y,z,t)}{\partial t} + \gamma_z(z)p_z(x,y,z,t) &= \kappa(x,y,z)\frac{\partial v_z(x,y,z,t)}{\partial z}\\
\frac{\partial v_x(x,y,z,t)}{\partial t} + \gamma_x(x)v_x(x,y,z,t) &= b(x,y,z)\frac{\partial p(x,y,z,t)}{\partial x}\\
\frac{\partial v_y(x,y,z,t)}{\partial t} + \gamma_y(y)v_y(x,y,z,t) &= b(x,y,z)\frac{\partial p(x,y,z,t)}{\partial x}\\
\frac{\partial v_z(x,y,z,t)}{\partial t} + \gamma_z(z)v_z(x,y,z,t) &= b(x,y,z)\frac{\partial p(x,y,z,t)}{\partial z},
\end{aligned} \tag{2.8}$$

where the pressure $p(x,y,z,t)$ is splitted into three unphysical components $p_x(x,y,z,t)$, $p_y(x,y,z,t)$ and $p_z(x,y,z,t)$, therefore satisfies $p = p_x + p_y + p_z$.

The 1-D functions γ_x, γ_y and γ_z define the damping functions in the PML layers surrounding the computation medium. These functions differ from zero only inside the PML layers. In the PML layers, I used $\gamma(x) = c_{pml}\ cos(\frac{\pi}{2}\frac{x}{L})$ where L denotes the width of the PML layer and x is a local coordinates in the PML layer, the origin of which is located at the outer edges of the model. A choice of c_{pml} that minimizes the reflection coefficient in the boundaries, defined in Collino and Tsogka (2001) and optimal for a homogeneous velocity model, is used even for heterogeneous models.

The system of equations is written into the Fourier domain and the functions $\xi_x(x) = 1 + \imath\gamma_x(x)/\omega$, $\xi_y(y) = 1 + \imath\gamma_y(y)/\omega$ and $\xi_z(z) = 1 + \imath\gamma_z(z)/\omega$ are introduced to simplify the equations,

$$\begin{aligned}
\frac{-\imath\omega\xi_x(x)}{\kappa(x,y,z)}p_x(x,y,z,\omega) &= \frac{\partial v_x(x,y,z,\omega)}{\partial x} + s(x,y,z,\omega)\\
\frac{-\imath\omega\xi_y(y)}{\kappa(x,y,z)}p_y(x,y,z,\omega) &= \frac{\partial v_y(x,y,z,\omega)}{\partial y}\\
\frac{-\imath\omega\xi_z(z)}{\kappa(x,y,z)}p_z(x,y,z,\omega) &= \frac{\partial v_z(x,y,z,\omega)}{\partial z}\\
-\imath\omega v_x(x,y,z,\omega) &= \frac{b(x,y,z)}{\xi_x(x)}\frac{\partial p(x,y,z,\omega)}{\partial x}\\
-\imath\omega v_y(x,y,z,\omega) &= \frac{b(x,y,z)}{\xi_y(y)}\frac{\partial p(x,y,z,\omega)}{\partial y}\\
-\imath\omega v_z(x,y,z,\omega) &= \frac{b(x,y,z)}{\xi_z(z)}\frac{\partial p(x,y,z,\omega)}{\partial z}.
\end{aligned} \tag{2.9}$$

The system (2.9) allows for a re-composition of the pressure field and therefore implies the first-order hyperbolic wave equation system when considering PML conditions as shown below

$$\begin{aligned}
\frac{-\iota\omega}{\kappa(x,y,z)}p(x,y,z,\omega) &= \frac{1}{\xi_x(x)}\frac{\partial v_x(x,y,z,\omega)}{\partial x}+\frac{1}{\xi_y(y)}\frac{\partial v_y(x,y,z,\omega)}{\partial y}\\
&+ \frac{1}{\xi_z(z)}\frac{\partial v_z(x,y,z,\omega)}{\partial z}+s(x,y,z,\omega)\\
v_x(x,y,z,\omega) &= \frac{\iota b(x,y,z)}{\omega\xi_x(x)}\frac{\partial p(x,y,z,\omega)}{\partial x}\\
v_y(x,y,z,\omega) &= \frac{\iota b(x,y,z)}{\omega\xi_y(y)}\frac{\partial p(x,y,z,\omega)}{\partial y}\\
v_z(x,y,z,\omega) &= \frac{\iota b(x,y,z)}{\omega\xi_z(z)}\frac{\partial p(x,y,z,\omega)}{\partial z}.
\end{aligned} \tag{2.10}$$

The second-order elliptic wave equation is obtained simply by injecting the equations of motion in the Hooke's law equation,

$$\left[\frac{\omega^2}{\kappa(x,y,z)}+\frac{1}{\xi_x(x)}\frac{\partial}{\partial x}\frac{b(x,y,z)}{\xi_x(x)}\frac{\partial}{\partial x}+\frac{1}{\xi_y(y)}\frac{\partial}{\partial y}\frac{b(x,y,z)}{\xi_y(y)}\frac{\partial}{\partial y}+\frac{1}{\xi_z(z)}\frac{\partial}{\partial z}\frac{b(x,y,z)}{\xi_z(z)}\frac{\partial}{\partial z}\right]. \\ p(x,y,z,\omega)=s(x,y,z,\omega). \tag{2.11}$$

2.3 Viscosity (attenuation)

One of the main advantages of working in the frequency domain is the straightforward implementation of attenuation. In fact, the implementation of attenuation is directly done through complex velocities (Toksöz and Johnston, 1981) on the contrary of the time domain where a cumbersome convolution operation is necessary. A common used relation is the Kolsky-Futterman model (Kolsky, 1956; Futterman, 1962) defined as

$$\bar{c}=c\left[\left(1+\frac{1}{\pi Q}|log(\omega/\omega_r)|\right)+\iota\frac{sgn(\omega)}{2Q}\right]^{-1}, \tag{2.12}$$

where $\bar{c}$ represents the complex velocity, Q the quality/attenuation factor, ω_r a given reference angular frequency and ι such that $\iota^2=-1$.

Practically, I use in the code an approximation of the Kolsky-Futterman model (Ribodetti et al., 1998) given by

$$\bar{c}=c\left(1-\iota\frac{sgn(\omega)}{2Q}\right). \tag{2.13}$$

The approximation implies first that the set of angular frequencies is close to ω_r, thus the term $\frac{1}{\pi Q}|log(\omega/\omega_r)|$ is neglected, and that only first-order approximation with regards to the quality factor Q of the remaining part of the equation is taken into account.

2.4 Finite-differencing of 3-D wave equation

In the prospect of using the direct solver approach to resolve the discrete system of equations, the differentiation is restricted to compact second-order stencils in order to lead to a reduced bandwidth impedance matrix. This property implies the minimization of the matrix fill-in during the LU decomposition. Indeed, the fill-in defines the extra non-zero terms generated due to the LU procedure. For band diagonal matrices, the fill-in occurs mostly in the matrix band (George and Liu, 1981; Hustedt et al., 2004) (Figure 2.1).

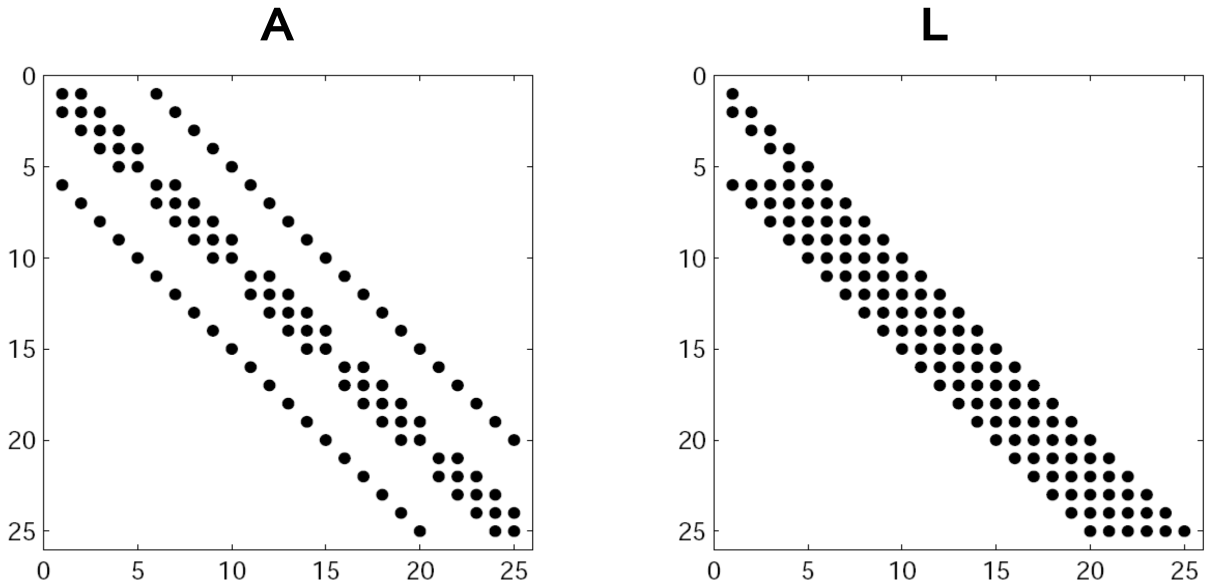

Figure 2.1: Matrix fill-in due to LU decomposition.

On the other hand, second-order stencils suffer from limited numerical precision and suggests to develop an appropriate differentiation procedure. The finite-differencing of equation (2.6) using the parsimonious mixed-grid approach is derived from the following procedure:

1. Eight 3-D coordinates systems are defined such that their axes span as many directions as possible in a cubic cell. These coordinates systems must be consistent with the 3-D second-order staggered geometry (Virieux (1984) for the 2-D case);

2. The first equation of system (2.5) is discretized on each of the coordinates system using second-order centered staggered-grid stencils. The discrete equation involves particle velocities on staggered grids;

3. The particle velocities at the grid nodes involved in the first equation of system (2.5) are inferred from the last three equations of system (2.5) using the same staggered-grid stencils that for the first equation;

4. The expression of the particle velocities are injected in the first equation of system (2.5) leading to a second-order parsimonious staggered-grid wave equation in pressure;

5. Once steps 2, 3 and 4 have been performed for each coordinates system, all the discrete wave equations are combined linearly such that numerical anisotropy and dispersion is minimized;

6. The accuracy of the stencil is greatly improved by a redistribution of the mass term over the different grid nodes surrounding the collocation node involved in the finite differences stencils following an anti-lumped mass approach.

The discretization procedure leads to a compact 27-nodes stencil with a numerical bandwidth of $\mathcal{O}(n^2)$ (Figure 2.2).

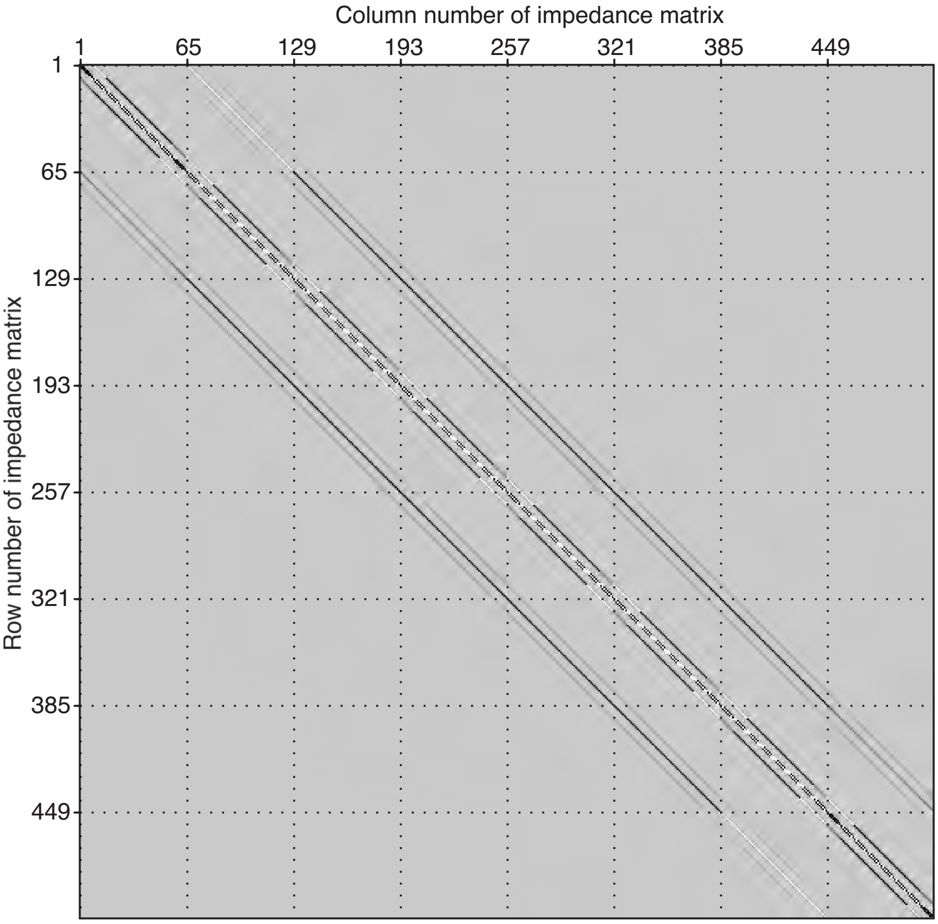

Figure 2.2: 3-D finite differences finite matrix, with 27 non zero terms per row, for a problem size $n_x \times n_y \times n_z = 8 \times 8 \times 8$.

2.4.1 Finite-differencing staggered grids

Staggered grids approach was originally developed by Yee (1966) for electromagnetic Maxwell's equation and used by Virieux and Madariaga (1982) to overcome the instabilities produced by the singularities at the edges of the fault thanks to its robustness with regards to solid-solid interface. Then, it was extensively used by Virieux (1984, 1986) to solve the two dimensions (2-D) elasto-dynamic wave equation.

In addition, staggered grids prevent the well-known black-red pattern. This pattern is due to the decoupling between the pressure and particle velocity grids if standard non-staggered procedure of finite-differencing is adopted. In fact, if the pressure grid is excited by an impulse source, one grid out of two remains unexcited on the particle velocity and pressure grids while the correct solution is computed at the intermediate positions. These unexcited subgrids are not solved in the discretization used in staggered grids (Hustedt et al., 2004).

2.4.2 Finite-differencing parsimonious strategy

Once the system of equations (2.5) is discretized, equations of motion are injected in the Hooke's law equation and discrete particle velocity components are eliminated. This strategy is called the parsimonious formulation and was introduced by Luo and Schuster (1990). The main advantage of this strategy is the memory saving. In fact, only one wavefield (pressure) is considered and therefore stored instead of four wavefields (pressure and three particle velocities).

2.4.3 Finite-differencing mixed grids

Mixed FD grids strategy was initially introduced by Jo et al. (1996) for the 2-D acoustic wave equation in a homogeneous medium. This technique aims to improve the numerical accuracy of the stencil. Jo et al. (1996) have combined the standard stencil and the 45° rotated stencil (Saenger, 2000). Hustedt et al. (2004) implemented the same strategy for the 2-D acoustic wave equation in a heterogeneous medium using a staggered grids formulation and the parsimonious approach. Stekl and Pratt (1998) developed it for the 2-D elastic heterogeneous wave equation. In 3-D geometry, three types of coordinates systems can be considered for the velocity-stress system:

1. One standard Cartesian coordinate system, denoted $\mathcal{S}_{\mathcal{S}}$ (Figure 2.3-a)).
2. Three coordinate systems, each one obtained by a 45° rotation of one of the axes of the standard coordinates system, denoted $\mathcal{S}_{\mathcal{R}}$ (Figure 2.3-b)).
3. Four coordinate systems, each one obtained by considering only three axes from the four big cube diagonals, denoted $\mathcal{S}_{\mathcal{D}}$ (Figure 2.3-c)).

The different stencils are mixed such that

$$w_1\mathcal{S}_{\mathcal{S}} + w_2\mathcal{S}_{\mathcal{R}} + w_3\mathcal{S}_{\mathcal{D}} = s, \tag{2.14}$$

where the weights w_1, w_2 and w_3 should verify the relationship

$$w_1 + \frac{w_2}{3} + \frac{w_3}{4} = 1. \tag{2.15}$$

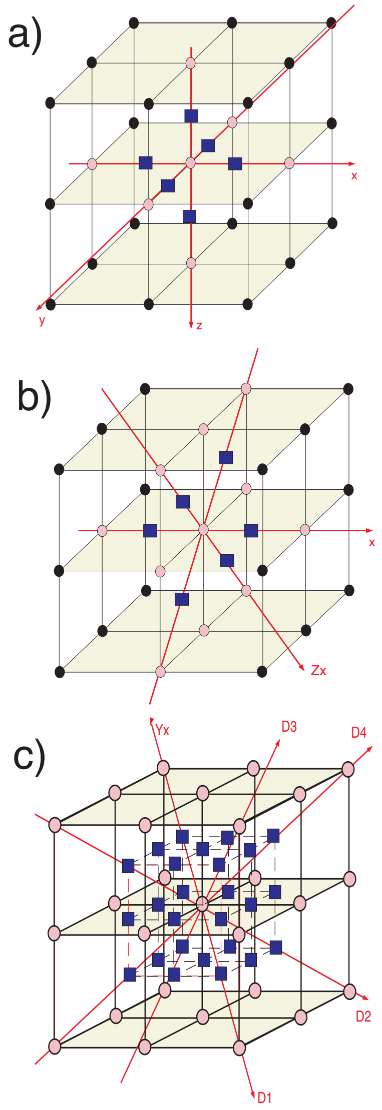

Figure 2.3: Different 3-D finite differences stencils involved in the mixed grids strategy.

2.4.4 Anti-lumped mass strategy

Following the standard procedure of finite element methods (Marfurt, 1984), the diagonal mass term is distributed through weighted values such that

$$\frac{\omega^2}{\kappa_{000}} p_{000} \Longrightarrow \omega^2 \left(w_{m1} \left[\frac{p}{\kappa}\right]_0 + w_{m2} \left[\frac{p}{\kappa}\right]_1 + w_{m3} \left[\frac{p}{\kappa}\right]_2 + w_{m4} \left[\frac{p}{\kappa}\right]_3 \right), \tag{2.16}$$

where $\left[\frac{p}{\kappa}\right]_0$ is related to the collocation grid node of the different considered stencils, $\left[\frac{p}{\kappa}\right]_1$ to extra grid nodes forming the standard coordinates system stencil (6 grid nodes), $\left[\frac{p}{\kappa}\right]_2$ to extra grid nodes forming the three rotated coordinates systems stencils (12 grid nodes) and $\left[\frac{p}{\kappa}\right]_3$ to extra grid nodes forming the four big diagonal coordinates systems stencils (8 grid nodes). This induces the relationship

$$w_{m1} + \frac{w_{m2}}{6} + \frac{w_{m3}}{12} + \frac{w_{m4}}{8} = 1. \tag{2.17}$$

This technique of mass redistribution reduces dramatically the numerical dispersion (Marfurt, 1984; Jo et al., 1996; Hustedt et al., 2004).

2.5 Dispersion analysis of the 3-D finite differences stencil

Now that the parsimonious mixed staggered grids finite differences procedure has been introduced, I am interested in the analysis of the numerical dispersion of the obtained stencil for a plane wave in an infinite homogeneous medium.

Consider an infinite homogeneous velocity model of velocity c and a constant density equal to 1. From Appendix C of Operto et al. (2007), the discrete wave equation (without PML conditions) reduces to

$$\begin{aligned} & \frac{\omega^2}{c^2}(w_{m1}p_{000} + \frac{w_{m2}}{6}p_1 + \frac{w_{m3}}{12}p_2 + \frac{w_{m4}}{8}p_3) + \frac{w_1}{h^2}(p_1 - 6p_{000}) \\ + & \frac{w_2}{3}\left[\frac{1}{h^2}(p_1 - 6p_{000}) + \frac{1}{4h^2}(2p_2 - 24p_{000})\right] + \frac{w_3}{4}(6p_3 - 4p_2 + 8p_1 - 48p_{000}), \end{aligned} \tag{2.18}$$

where

$$\begin{aligned} p_1 &= p_{100} + p_{010} + p_{001} + p_{-100} + p_{0-10} + p_{00-1}, \\ p_2 &= p_{110} + p_{011} + p_{101} + p_{-110} + p_{0-11} + p_{-101} + p_{1-10} \\ &+ p_{01-1} + p_{10-1} + p_{-1-10} + p_{0-1-1} + p_{-10-1}, \\ p_3 &= p_{111} + p_{-1-1-1} + p_{-111} + p_{1-11} + p_{11-1} + p_{-1-11} + p_{1-1-1} + p_{-11-1}. \end{aligned} \tag{2.19}$$

Following a classic harmonic approach, I insert the discrete expression of a plane wave, $p_{lmn} = e^{-\iota hk(l\cos\phi cos\theta + m\cos\phi\sin\theta + n\sin\phi)}$ where $\iota^2 = -1$, in equation (2.18). The phase velocity is given by $\frac{\omega}{k}$. The normalized phase velocity is defined by $V_{ph} = \frac{v_{ph}}{c}$ and the number of nodes per wavelength (λ) by $G = \frac{\lambda}{h} = \frac{2\pi}{kh}$. After some straightforward although cumbersome manipulations, the following expression for the numerical phase velocity is obtained,

$$V_{ph} = \frac{G}{\sqrt{2J}\pi}\sqrt{w_1(3 - C) + \frac{w_2}{3}(6 - C - B) + \frac{2w_3}{4}(3 - 3A + B - C)}, \tag{2.20}$$

where $J = (w_{m1} + 2w_{m2}C + 4w_{m3}B + 8w_{m4}A)$ with

$$\begin{aligned} A &= \cos a \cos b \cos c, \\ B &= \cos a \cos b + \cos a \cos c + \cos b \cos c, \\ C &= \cos a + \cos b + \cos c. \end{aligned} \tag{2.21}$$

and $a = \frac{2\pi}{G}\cos\phi\cos\theta$, $b = \frac{2\pi}{G}\cos\phi\sin\theta$ and $c = \frac{2\pi}{G}\sin\phi$. We can easily check that, $V_{ph} \longrightarrow 1$ when $G \longrightarrow \infty$ for $J = 1$ and for the 3 cases $(w_1, w_2, w_3) = (1,0,0), (0,1,0)$ and $(0,0,1)$ whatever are ϕ and θ. This validates the expression of the phase velocity in equation (2.20).

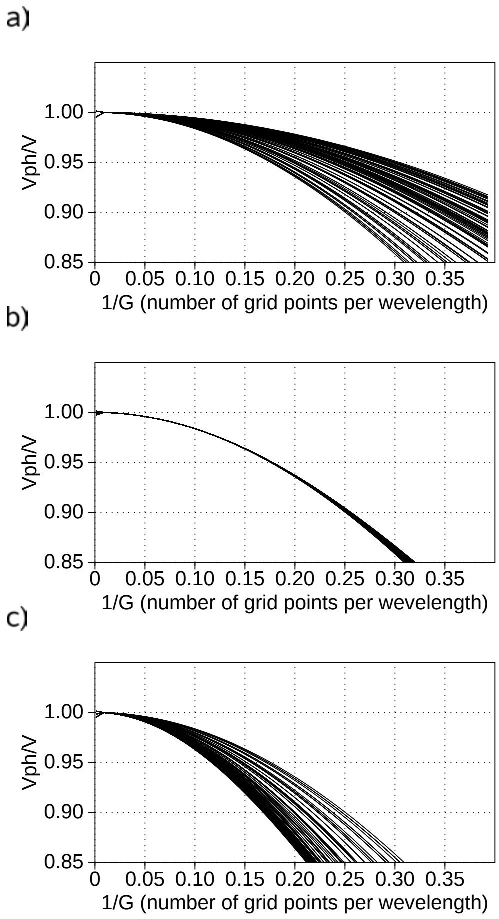

Figure 2.4: Dispersion curves for phase velocity. a) Stencil 1 without mass averaging. b) Stencil 2 without mass averaging. c) Stencil 3 without mass averaging. The curves are plotted for angles θ and ϕ ranging from 0 to 45^o.

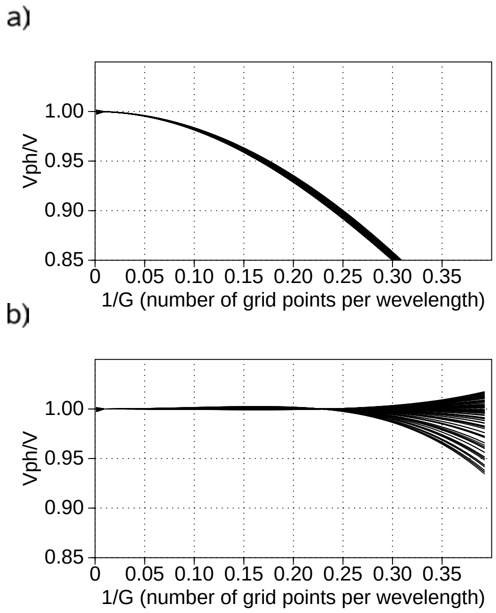

Figure 2.5: Phase velocity dispersion curve for mixed-grid stencil: a) without mass averaging. b) with mass averaging. The curves are plotted for angles θ and ϕ ranging from 0 to 45^o.

The five independent parameters w_{m1}, w_{m2}, w_{m3}, w_1, w_2 which minimize the least squares norm of the misfit $1 - V_{ph}$ need to be evaluated. These coefficients are estimated by a global optimization procedure based on a Very Fast Simulating Annealing algorithm (Sen and Stoffa, 1995). The cost function is minimized for 5 angles ϕ and θ spanning between 0 and 45^o and for 4 values of G ranging between 4 and 10.

The values $w_{m1} = 0.4964958$, $w_{m2} = 7.516875E-02$, $w_{m3} = 4.373936E-03$, $w_1 = 1.8395265E-05$ and $w_2 = 0.890077$ are found, which imply $w_{m4} = 5.69038E-07$ and $w_3 = 0.1099046$. The coefficients show that stencils 2 and 3 have a dominant contribution in the mixed-grid stencil. On the other hand, the mass coefficients show a dominant contribution of the coefficients located at the collocation node and at the nodes associated with stencil 1.

The dispersion curves for the three stencils 1, 2 and 3 without mass averaging are shown in Figure 2.4. These stencils used individually would require up to 40 grid nodes per wavelength. The phase velocity dispersion curve for the mixed stencil without mass averaging are shown in Figure 2.5.

Note how the dispersion curves for different incidence angles are focused illustrating the isotropy of the stencil. However, the accuracy of the stencil remains poor (Figure 2.5-a)). Combining the mixed-grid discretization strategy with the mass averaging allows us to mitigate both numerical anisotropy and dispersion (Figure 2.5-b)). The phase velocity dispersion curves suggest that a discretization rule of 4 grid nodes per wavelength can be used. If the wave propagation modeling algorithm is used as an engine for full-waveform inversion, this discretization rule is optimal in the sense that the theoretical resolution of full-waveform inversion at normal incidence is half the wavelength (Miller et al., 1987). In fact, the Nyquist criterion suggests that in order to correctly sample any given signal and avoid aliasing issues, more than two grid nodes per minimum wavelength are necessary.

Conclusion

In this chapter, I was interested in expounding the design of the 3-D finite differences stencil which is used to discretize the Helmholtz equation based on Operto et al. (2007). The governing idea is to design both compact second-order and accurate stencil to mitigate the matrix fill-in during the LU decomposition. I have introduced the different ingredients to build an efficient stencil. The staggered particle velocity-pressure grids avoid the red-black pattern phenomenon. The parsimonious strategy reduces the memory storage. The mixed grids strategy improve the behavior with regards to the numerical anisotropy. The anti-lumped mass approach guarantees a small numerical dispersion. This strategy leads to a discretization rule of four grid nodes per propagated wavelength. This rule is optimal for our imaging strategy based on full waveform inversion since theoretical resolution is half the wavelength.

Chapter 3

Numerical methods for frequency domain wave modeling

In the previous chapter, I introduced a suitable discretization of the frequency domain wave equation. Once the wave equation is discretized, it can be recast as a linear matrix system $\mathbf{A}\mathbf{p} = \mathbf{s}$ and resolved either in the time or the frequency domain. In the time domain, explicit time marching scheme is preferred since it avoids to build and invert the big impedance matrix $\mathbf{A}$, necessary at each time step. In the frequency domain, several approaches are available to solve the linear matrix system: the direct solver approach through an LU decomposition (Press et al., 1992; Operto et al., 2007), the iterative solver approach (Riyanti et al., 2007; Plessix, 2007) and the hybrid solver approach based on domain decomposition (Haidar, 2008; Sourbier et al., 2008b).

Direct solvers are the method of choice when the robustness is the primary concern. This approach is known for its multi right hand sides (RHS) resolution efficiency, an essential feature in the prospect of imaging where the seismic wave modeling for thousands of seismic sources is involved. It is possible nowadays to solve 3-D problems with a few million unknowns in a robust way with solvers exploiting modern parallel supercomputers. Unfortunately, direct methods suffer from their high time complexity and memory storage and from their poor scalability which prevent to tackle problems involving several tens of millions of unknowns.

On the other hand, iterative solvers require fewer storage and often fewer computation operations depending on the required accuracy. Nevertheless, the performance of iterative methods depends strongly on the spectral properties of the linear system to solve. To improve efficacy and robustness, an efficient preconditioner need to be found. This task is critical and can be cumbersome.

The third class of solvers, hybrid methods, tries to find a compromise between the two previous solvers through a domain decomposition method. The direct solver is applied to small domains (subdomains), thus the storage and computation issues are mitigated. The iterative solver is applied to a reduced system, built only for the interface grid nodes, leading to a better conditioned system than the full system. In addition, several efficient algebraic preconditioners can be used independently of the physics behind the numerical problem to solve.

Both direct and hybrid solvers have been implemented as a modeling engine in our FWI software. Indeed, I will first expand theoretical and practical properties of the direct and hybrid solvers. Then, I will validate the hybrid solver with regards to analytic solutions in homogeneous media and direct solver solutions in complex heterogeneous media. I will finally discuss their numerical complexities with regards to each other and against iterative and time domain solvers.

3.1 Direct solver

The most accurate and robust approach to solve a linear system is the direct method. Solution precision in this case is the machine precision. Nevertheless, this approach is nowadays limited to sparse matrices problems of few millions of unknowns due to its numerical cost in terms of CPU time and memory storage requirements and to limitations related to matrix conditioning issues. Unfortunately, the condition number increases when the size of the considered physical problem increases.

3.1.1 Theory

Direct methods are based on Gauss elimination technique. Direct methods aim to rewrite the system

$$\mathbf{A}\mathbf{p} = \mathbf{s}, \tag{3.1}$$

into a more manageable form like

$$(\mathbf{L}\mathbf{U})\mathbf{p} = \mathbf{s}, \tag{3.2}$$

where **L** is a lower triangular matrix and **U** is an upper triangular matrix for an asymmetric matrix.

This system is then efficiently solved in two steps, upward and downward elimination phases, through inserting the temporary vector **y**,

$$\begin{aligned} \mathbf{L}\mathbf{y} &= \mathbf{s}, \\ \mathbf{U}\mathbf{p} &= \mathbf{y}. \end{aligned} \tag{3.3}$$

For sparse matrices, only non zero matrix terms are stored. In the same way, only non zero terms introduced in the LU decomposition are computed. However, the matrix decomposition leads to L and U matrices denser than the initial matrix and less than the full one. This issue is called *fill-in*. During the last decades, many techniques to reduce the fill-in have been developed. These techniques renumber/reorder the rows/columns of the matrix based on its graph. For these reasons, these techniques are called reordering techniques (George and Liu, 1981; Amestoy et al., 1996; Ashcraft and Liu, 1998).

3.1.2 Complexity & scalability analysis

3.1.2.1 Theory

The main advantage of direct methods is their efficiency to deal with multiple right hand side (RHS) terms (Marfurt, 1984). In fact once the LU decomposition is done and L and U matrices are stored in the core memory, solutions are obtained by simple substitutions of the RHS terms. This can be written in a numerical way as

$$(\mathbf{LU})\,[\mathbf{p_1},\mathbf{p_2},\mathbf{p_3},...,\mathbf{p_N}] \quad = \quad [\mathbf{s_1},\mathbf{s_2},\mathbf{s_3},...,\mathbf{s_N}]\,. \tag{3.4}$$

This property is very important for applications such as seismic imaging based on waveform inversion where thousands of sources, i.e. RHS, are involved. On the other hand, the main drawback of the direct methods is the LU decomposition itself and its CPU computation and memory storage cost. Indeed, if we consider a cubic numerical domain $n \times n \times n$, i.e. the impedance matrix order is n^3, the CPU computation complexity is $\mathcal{O}(n^9)$ and memory storage complexity is $\mathcal{O}(n^6)$. For sparse matrices and using efficient reordering techniques such as METIS, based on nested dissection, these complexities can be lowered to respectively $\mathcal{O}(n^6)$ and $\mathcal{O}(n^4)$ (Ashcraft and Liu, 1998; George and Liu, 1981).

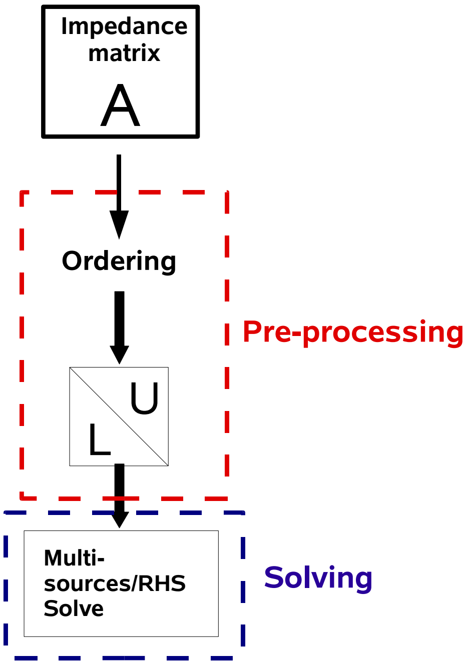

Figure 3.1: Direct solver modeling flowchart.

3.1.2.2 Parallel direct solver MUMPS

I propose in this section to assess the complexity and scalability properties of the massively parallel direct solver MUMPS for wave modeling applications (Amestoy et al., 2006; MUMPS-team, 2007). The MUMPS solver is based on a multifrontal method (Duff and Reid, 1983; Liu, 1992). The multifrontal method consists of a succession of factorizations of dense matrices called frontal matrices to take advantage of the optimized libraries for dense linear algebra (BLAS, BLACS, SCALAPACK libraries). The frontal matrices form the supernodes of an assembly tree (a particular form of an elimination tree) which describes the order in which the matrix can be factored. The assembly tree is built during the analysis or symbolic factorization phase. During this phase, ordering techniques are performed to minimize the dependencies in the assembly tree and so the fill-in during factorization. For ordering, I used METIS which is based on a hybrid multilevel nested dissection and multiple minimum degree algorithm (Karypis and Kumar, 1998).

I assessed the performances of MUMPS solver for our specific application considering cubic FD grids of increasing dimension n. n varies from 40 to 130. The number of MPI processes also varies. I used one MPI process with 8 GB of memory for n between 40 and 70 and 8 MPI processes between 80 and 130. For the values of n between 120 and 130, the analysis phase succeeded while the factorization phase has not been performed because of lack of core memory.

The Figure 3.2 plots the ratio between LU CPU-operations complexity and the theoretical time complexity $\mathcal{O}(n^6)$. The curve tends to a constant value. This confirms the theoretical time complexity. The number of LU factors divided by memory complexity is plotted in Figure 3.3. The curve still presents a slope and suggests that the memory complexity is higher than $\mathcal{O}(n^4)$. The same statement is suggested by Operto et al. (2007, Figure 11). If the same plot is drawn for the complexity $\mathcal{O}(n^4 Log_2(n))$, the curve becomes nearly flat (constant ≈ 4) and implies that the memory complexity is closer to $\mathcal{O}(n^4 Log_2(n))$ than $\mathcal{O}(n^4)$ (Figure 3.4). Nevertheless, this memory does not correspond to the real used memory. In fact, extra memory called overhead memory is added. Overhead memory explains supplementary memory needed for parallel implementation of LU factorization. Figure 3.5 shows the ratio between LU factors memory and total used memory. The difference represents overhead memory. This explains the jump at $n = 80$ since the simulations for $n \leq 70$ are executed in sequential and $n \geq 80$ in parallel.

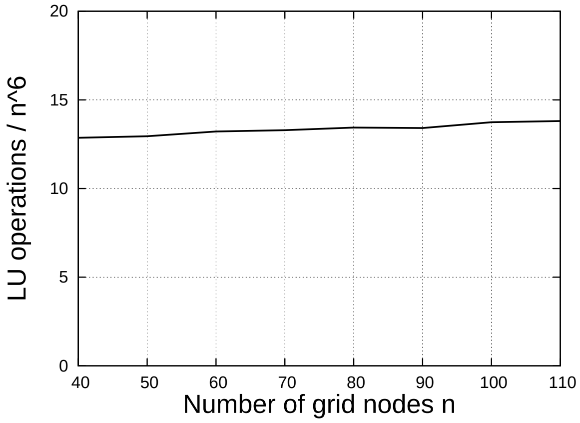

Figure 3.2: MUMPS LU decomposition time complexity.

3.2 Hybrid solver based on domain decomposition

To overcome the overburden memory complexity of the LU decomposition, I implemented, in collaboration with F. Sourbier and A. Haidar, the hybrid (direct-iterative) solver based on domain decomposition method (DDM) (Haidar, 2008). For domain decomposition, an element-partitioning without overlapping is followed (Saad, 2003). The direct solver is applied on smaller numerical computation subdomains thus memory requirement is less

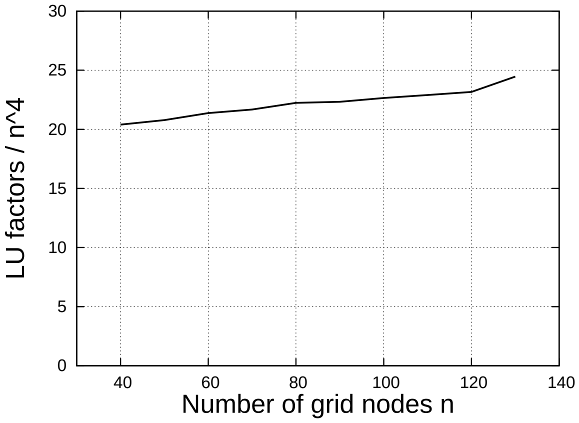

Figure 3.3: MUMPS LU decomposition memory complexity.

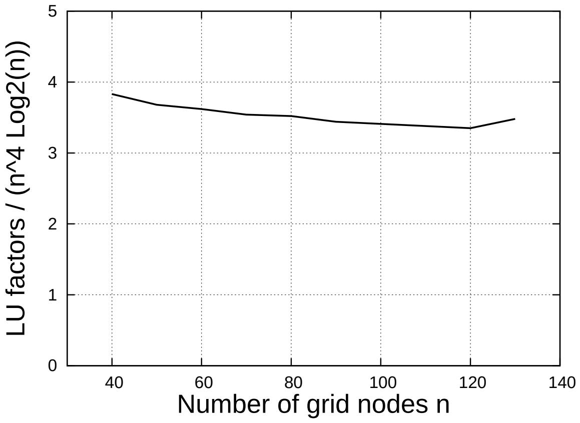

Figure 3.4: MUMPS LU decomposition memory complexity.

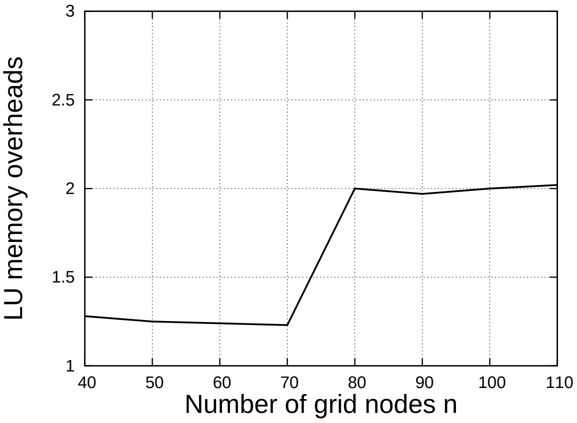

Figure 3.5: MUMPS memory overheads.

restricting and the iterative solver is applied on a reduced system (built at only boundary grid nodes) thus convergence and preconditioning issues are less challenging.

On the other hand, the hybrid solver should be less efficient with regards to multi-RHS resolution than the direct solver since the iterative component linearly increases with the number of RHSs.

3.2.1 Theory

The domain decomposition method consists in splitting the computational domain Ω into subdomains Ω_i (Figure 3.6). Each subdomain shares interface nodes with its adjacent subdomains. Interface nodes belong to the domain Γ. Therefore, each subdomain Ω_i can be divided into two subgroups: interior nodes $\mathcal{I}_i$ (p_i in Figure 3.6) and interface/boundary nodes Γ_i (p_b in Figure 3.6).

After reordering the interior nodes by subdomain Ω_i and labeling the interface nodes

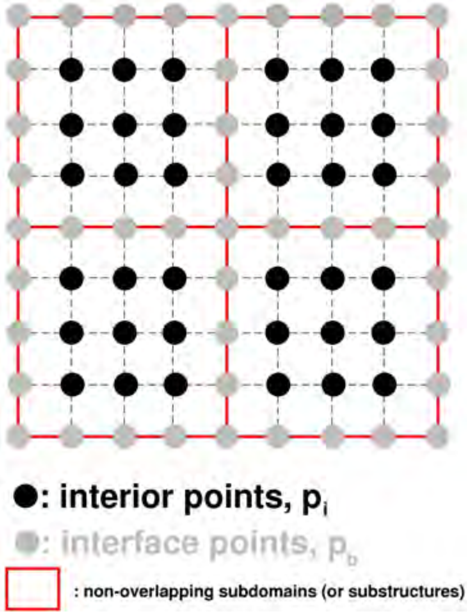

Figure 3.6: Domain splitting in the hybrid method.

last, the system $\mathbf{A}\mathbf{p} = \mathbf{s}$ becomes

$$\begin{bmatrix} A_{\mathcal{I}_1\mathcal{I}_1} & 0 & \cdots & \cdots & 0 & A_{\mathcal{I}_1\Gamma_1} \\ 0 & A_{\mathcal{I}_2\mathcal{I}_2} & \ddots & \cdots & \vdots & A_{\mathcal{I}_2\Gamma_2} \\ \vdots & \ddots & A_{\mathcal{I}_3\mathcal{I}_3} & \ddots & \vdots & A_{\mathcal{I}_3\Gamma_3} \\ \vdots & \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \cdots & \cdots & 0 & A_{\mathcal{I}_n\mathcal{I}_n} & A_{\mathcal{I}_n\Gamma_n} \\ A_{\Gamma_1\mathcal{I}_1} & A_{\Gamma_2\mathcal{I}_2} & A_{\Gamma_3\mathcal{I}_3} & \cdots & A_{\Gamma_n\mathcal{I}_n} & \bar{A}_{\Gamma\Gamma} \end{bmatrix} \begin{bmatrix} p_{\mathcal{I}_1} \\ p_{\mathcal{I}_2} \\ p_{\mathcal{I}_3} \\ \vdots \\ p_{\mathcal{I}_n} \\ \bar{p}_\Gamma \end{bmatrix} = \begin{bmatrix} s_{\mathcal{I}_1} \\ s_{\mathcal{I}_2} \\ s_{\mathcal{I}_3} \\ \vdots \\ s_{\mathcal{I}_n} \\ s_\Gamma \end{bmatrix}, \tag{3.5}$$

and in a more compact form

$$\begin{bmatrix} A_{\mathcal{I}_i\mathcal{I}_i} & A_{\mathcal{I}_i\Gamma_i} \\ A_{\Gamma_i\mathcal{I}_i} & \bar{A}_{\Gamma\Gamma} \end{bmatrix} \begin{bmatrix} p_{\mathcal{I}_i} \\ \bar{p}_\Gamma \end{bmatrix} = \begin{bmatrix} s_{\mathcal{I}_i} \\ s_\Gamma \end{bmatrix}, \tag{3.6}$$

where $p_{\mathcal{I}_i}$ denotes unknowns located at interior domains $\mathcal{I}_i$ and $\bar{p}_\Gamma$ denotes unknowns located at interface nodes domain Γ.

Eliminating $p_{\mathcal{I}_i}$ from the second block row of equation (3.6) leads to the following reduced system for $\bar{p}_\Gamma$,

$$\begin{aligned} (\bar{A}_{\Gamma\Gamma} - A_{\Gamma_i\mathcal{I}_i} A^{-1}_{\mathcal{I}_i\mathcal{I}_i} A_{\mathcal{I}_i\Gamma_i})\bar{p}_\Gamma &= s_\Gamma - A_{\Gamma_i\mathcal{I}_i} A^{-1}_{\mathcal{I}_i\mathcal{I}_i} s_{\mathcal{I}_i} \\ S\bar{p}_\Gamma &= s_\Gamma - A_{\Gamma\mathcal{I}_i} A^{-1}_{\mathcal{I}_i\mathcal{I}_i} s_{\mathcal{I}_i}. \end{aligned} \tag{3.7}$$

The matrix $S = \bar{A}_{\Gamma\Gamma} - A_{\Gamma\mathcal{I}_i} A^{-1}_{\mathcal{I}_i\mathcal{I}_i} A_{\mathcal{I}_i\Gamma}$ is the Schur complement matrix (Saad, 2003) (note the use of Einstein notation). This Schur complement system (3.7) can be expanded to a useful expression

$$\begin{aligned} S &= \sum_i R_{\Gamma_i}^T S_i R_{\Gamma_i} \\ &= \sum_i R_{\Gamma_i}^T (A_{\Gamma_i \Gamma_i} - A_{\Gamma_i \mathcal{I}_i} A_{\mathcal{I}_i \mathcal{I}_i}^{-1} A_{\mathcal{I}_i \Gamma_i}) R_{\Gamma_i}, \end{aligned} \tag{3.8}$$

where S_i is referred to as local Schur complement and involves submatrices from the impedance matrix A. The local Schur complement matrix S_i is a dense matrix even if A is sparse. $R_{\Gamma_i} : \Gamma \rightarrow \Gamma_i$ is the canonical restriction which maps vectors defined on Γ into vectors defined on Γ_i, and $R_{\Gamma_i}^T : \Gamma_i \rightarrow \Gamma$ its transpose. This form is interesting because it highlights how it is natural to parallelize matrix-vector products involving S since each local Schur complement can be computed and stored independently on different processors.

The Schur complement system (3.7) is significantly better conditioned than the original matrix system. For example, if we consider the system resulting from the discretization of the Laplace operator on a mesh with spacing h, the condition number of the relative matrix A is $\kappa(A) = \mathcal{O}(h^{-2})$. Using two non-overlapping subdomains reduces the condition number of the Schur complement matrix to $\kappa(S) = \mathcal{O}(h^{-1})$ (Haidar, 2008).

3.2.2 Additive Schwarz preconditioner

3.2.2.1 Standard approach

As mentioned previously, it is crucial to have an efficient preconditioner when solving a linear system with an iterative method. It is the case of the Schur complement equation (3.7). The preconditioner transforms the original system into another one that have the same solution but better convergence properties. If the initial Schur complement system is $Sp = s$, the preconditioned system reads $SMp = s$ where M approximates S^{-1}.

I focus on a class of preconditioner called additive Schwarz preconditioner (Saad, 2003; Haidar, 2008). This preconditioner is related directly to domain decomposition methods. Such preconditioners belong to the class of algebraic preconditioners. This suggests that this type of preconditioners does not depend on the physics of the linear system to solve.

Consider the Schur complement system (3.8). The additive Schwarz preconditioner is defined as

$$M_{AS} = \sum_i R_{\Gamma_i}^T \bar{S}_i^{-1} R_{\Gamma_i}. \tag{3.9}$$

where $\bar{S}_i$ is called the local assembled Schur complement and defined such as, $\bar{S}_i = R_{\Gamma_i} S R_{\Gamma_i}^T$. $\bar{S}_i$ corresponds to the restriction of the Schur complement to the interface Γ_i.

The evaluation of the Schur complement and the computation of the additive Schwarz preconditioner can be implemented intuitively in parallel. In fact, the preconditioner is the sum up of the inverses of local assembled Schur complements. Nevertheless, the compututation of the local assembled Schur complement interface nodes terms implies to apply some MPI point-to-point communications between neighbor subdomains since the interface nodes are shared by neighbouring subdomains.

3.2.2.2 Improved approach

In this section, I present a new class of additive Schwarz preconditioners. I call it "shifted Laplace additive Schwarz" preconditioner for Schur complement system. To build this new preconditioner, I combine the standard approach of additive Schwarz preconditioner with the well known shifted Laplace preconditioner used in the iterative methods for the Helmholtz equation (Vuik et al., 2003; Erlangga et al., 2003, 2004; Erlangga, 2005; Riyanti et al., 2006).

Consider the simplified Helmholtz equation,

$$\mathcal{A}p := [\Delta + k^2]p = s, \tag{3.10}$$

where k is the wavenumber and $\mathcal{A}$ is the Helmholtz operator. The general shifted Laplace operator $\mathcal{M}$ reads

$$\mathcal{M}p := [\Delta + (\beta_1 + \iota\beta_2)k^2]p = s, \tag{3.11}$$

where $(\beta_1, \beta_2) \in \mathbb{R} \times \mathbb{R}$ and $\iota = \sqrt{-1}$.

Erlangga (2005, chapter 4, 6) widely discusses the choice of the optimal values of the pair (β_1, β_2) and bases his discussion on the study of the eingenvalues of the 1-D Helhmoltz problem. Figure 3.7 is extracted from Erlangga (2005) and shows the impact of different shifted Laplace preconditioners on the eigenvalues distribution, therefore on the convergence.

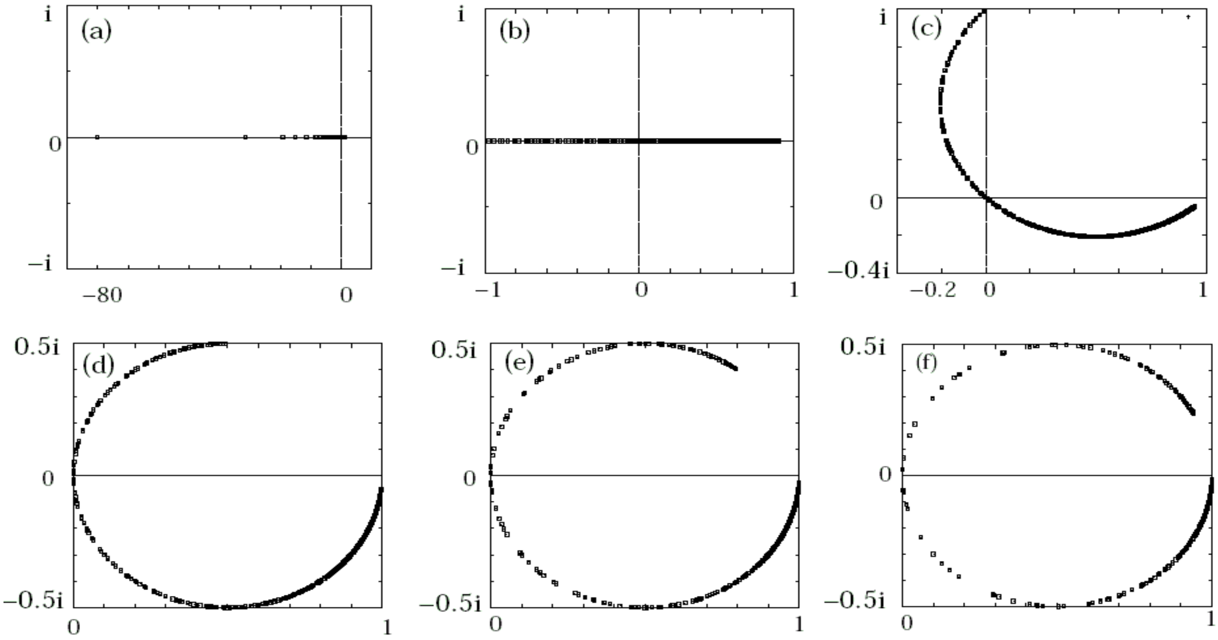

Figure 3.7: Spectral pictures of $\mathcal{AM}^{-1}$ without damping and different values of (β_1, β_2): (a) (0,0), (b) (-1,0), (c) (0,1), (d) (1,1), (e) (1,0.5), (f) (1,0.3). (From Erlangga (2005)).

Erlangga (2005) has also shown that the introduction of viscosity (attenuation) improves the convergence since the system has better properties (Figure 3.8). This may explain the efficiency of the complex shifted Laplace preconditioner since the introduction of the imaginary part of the complex shift is equivalent to the introduction of attenuation. In fact, the damped simplified Helmholtz equation can be expressed as,

$$\mathcal{A}_\alpha p := [\Delta + (1 - \iota\alpha)k^2]p = s, \tag{3.12}$$

which corresponds to a particular choice of $(\beta_1, \beta_2) = (1, -\alpha)$.

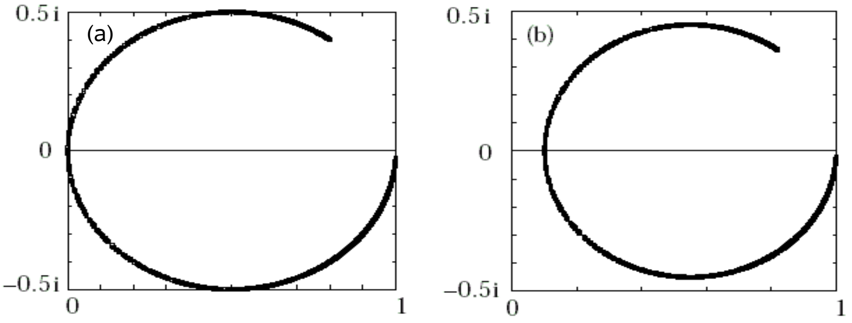

Figure 3.8: Spectral pictures of $\mathcal{A}\mathcal{M}^{-1}$ with (β_1, β_2)=(1,0.5): (a) without attenuation, (b) with attenuation. (From Erlangga (2005)).

An empirical optimal choice of the pair (β_1, β_2) for our Schur problem is $(1, \beta_2 \in [0.01 : 0.02])$ while Erlangga (2005) has advised $(1, 0.5)$.

3.2.2.3 Comparison of the two classes of preconditioners

Impact of the attenuation on the GMRES convergence: 3-D SEG/EAGE salt model A simulation is performed with the SEG/EAGE salt model originally discretized on a $676 \times 676 \times 210$ FD grids with a grid interval of 20 m leading to a computational domain of dimensions $13.5 \times 13.5 \times 4.18$ km^3. The minimum and maximum velocities are 1.5 and 4.482 km/s respectively. The salt model is a slowly varying sedimentary medium within which is embedded a salt body leading to a strong velocity contrast between sediments and salt (Figure 3.9-a).

The salt model was resampled with a grid interval of 30 m to perform a simulation at a frequency of 12.5 Hz corresponding to 4 grid nodes per minimum propagated wavelengths (Figure). The number of grid nodes in the PML layers is 6 grid nodes. The simulation was performed on the IBM Blue Gene (Babel) of IDRIS computer center on $14 \times 14 \times 5 = 980$ processors using subdomains of $33 \times 33 \times 30$ grid nodes. Two Gigabytes of core memory were allocated per MPI process. The source coordinates are (1.05 km, y = 1.05 km , z = 0.3 km). The PML absorbing boundary conditions are implemented along the six faces of the model and, therefore, no free surface is considered in this simulation.

When $Q = 10000$, i.e. attenuation is neglected, the number of GMRES iterations required to decrease the error down to $\epsilon = 10^{-3}$ and $\epsilon = 10^{-2}$ were 1375 and 954 respectively. The monochromatic wavefield for $\epsilon = 10^{-3}$ is shown in Figure 3.9-b).

When a heterogeneous attenuation model is used with $Q = 50$ in the salt body (Figure 3.10-a), these numbers of iterations decrease to 111 and 57 respectively. This highlights the dramatic positive impact of attenuation on the convergence of the iterative resolution of the Schur complement system.

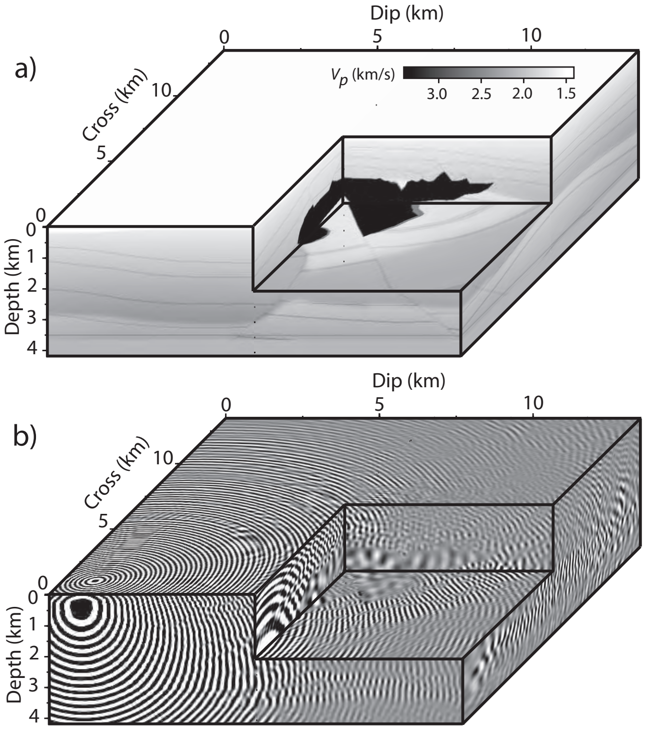

Figure 3.9: SEG/EAGE salt model. Q= 10000. a) Velocity model. b) Monochromatic wavefield at 12.5 Hz.

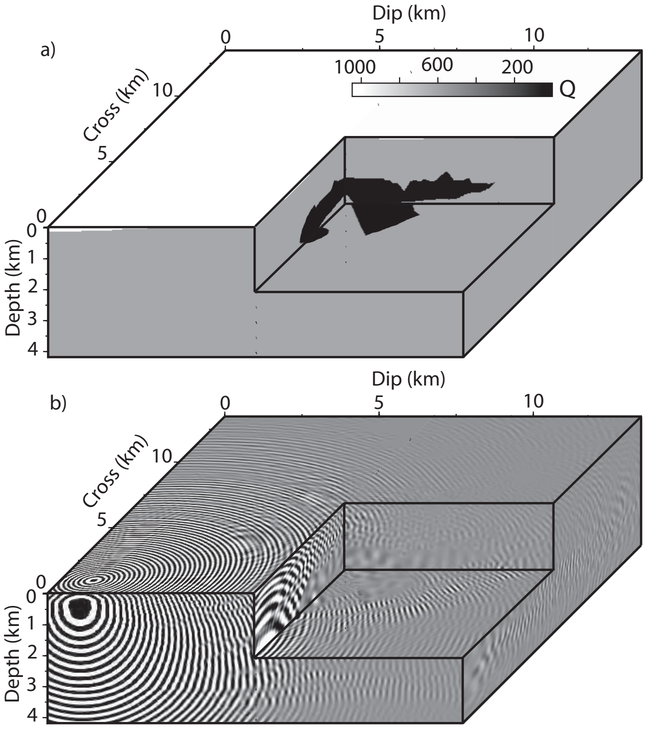

Figure 3.10: SEG/EAGE salt model. a) Attenuation model. b) Monochromatic wavefield at 12.5 Hz.

2-D overthrust model In order to study the impact of the preconditioner on the convergence, I apply the previously defined additive Schwarz preconditioners. I have considered a 2-D complex velocity model extracted from the SEG/EAGE overthrust model, a constant attenuation model $Q = 10000$ and seven frequencies ranging from 3.5 to 20 Hz. Note that the computation grid is not adapted to the frequency and only the 20-Hz finest grid ($227 \times 3 \times 841$) is considered. The adopted decomposition is $4 \times 1 \times 10$ subdomains. The

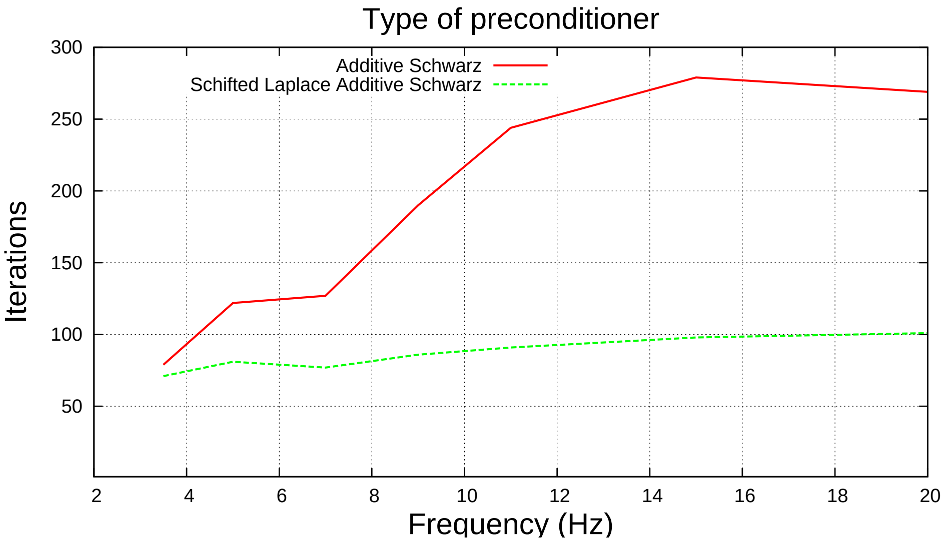

Figure 3.11: Impact of preconditioning type on GMRES convergence. *solid red*) Additive Schwarz preconditioner. *dashed green*) Schifted Laplace Additive Schwarz.

Figure 3.11 illustrates the convergence rate of each of the used preconditioners and plots the number of iterations in the GMRES part with respect to the frequency. It shows clearly that the convergence rate is deteriorating (the number of iterations is increasing) with increasing frequency with the additive Schwarz preconditioner (Erlangga, 2005). However, if the shifted Laplace additive Schwarz is used, the convergence rate is improved and the number of iterations tends to reach a plateau around one hundred iterations.

3-D overthrsut model The second application has been applied on the 3-D overthrust velocity model. The discretization is adapted to the frequency 7 Hz. The FD grid dimension is $71 \times 275 \times 275$ grid nodes including PML layers (≈ 5.5 million unknowns). The adopted domain decomposition is $3 \times 9 \times 9$. In this application, two different Q factor values have been used. The first, Q=10000, means that no attenuation is considered while the second introduces the attenuation (Q=300).

Table 3.1 gathers the numbers of iterations obtained with the different configurations related to the Q factor value and the choice of the preconditioner and for a precision criterion $\epsilon = 10^{-3}$. Two main conclusions emerge: first, the introdution of the attenuation improves the GMRES convergence (Erlangga, 2005) and second, the shifted Laplace preconditioner has a better behaviour than the standard approach.

Table 3.1: Impact of preconditioning type on GMRES convergence.

Attenuation (Q factor)	Standard	Shifted Laplace
$Q = 10000$	1150	330
$Q = 300$	350	170

3.2.3 Parallel implementation & performances

In parallel distributed environments, each subdomain is assigned to one MPI process. During the hybrid solver resolution procedure, we can distinguish two different phases: a preprocessing phase which does not depend on the source/RHS term and a solving phase. Figure 3.12 summarizes the two phases of the hybrid solver resolution.

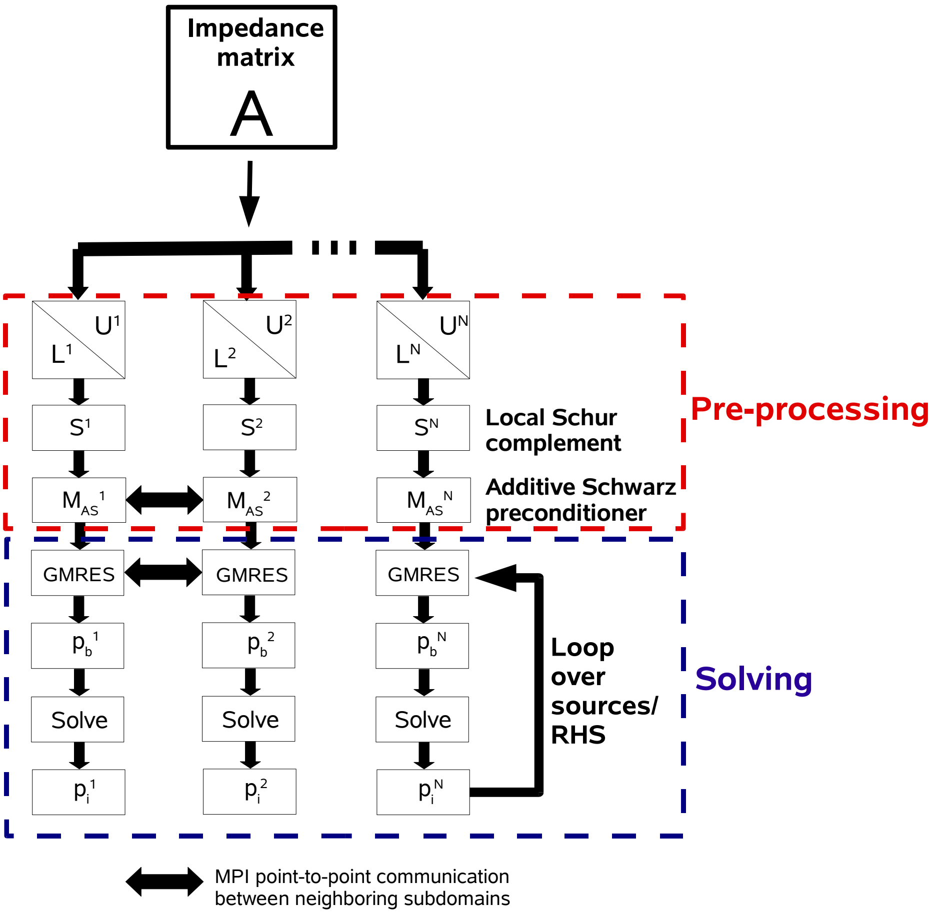

Figure 3.12: Hybrid solver modeling flowchart.

During the preprocessing phase, local matrices $A_{\mathcal{I}_i\mathcal{I}_i}$, $A_{\Gamma_i\mathcal{I}_i}$, $A_{\mathcal{I}_i\Gamma_i}$ and $A_{\Gamma_i\Gamma_i}$ are extracted from the assembled matrix A. Concurrent factorizations applied to local matrices $A_{\mathcal{I}_i\mathcal{I}_i}$ are performed by each MPI process using the direct solver MUMPS. MUMPS can provide the local Schur complement during the LU decomposition. However, the S_i matrix is dense and its storage in core may be memory demanding. An alternative is to avoid the explicit building of S_i to perform the matrix-vector product of the form $S_i x = (A_{\Gamma_i\Gamma_i} - A_{\Gamma_i\mathcal{I}_i} A_{\mathcal{I}_i\mathcal{I}_i}^{-1} A_{\mathcal{I}_i\Gamma_i})x$, which must be repeated for each RHS, by implementing the matrix-vector products $A_{\Gamma_i\Gamma_i}x$, $y = A_{\mathcal{I}_i\Gamma_i}x$, $A_{\Gamma_i\mathcal{I}_i}z$ and the resolution phase $z = A_{\mathcal{I}_i\mathcal{I}_i}^{-1}y$.

The last step in the preprocessing phase is to build the preconditioner. As mentioned in the previous section, the local preconditioner consists in the inverse of the assembled local Schur complement $\bar{S}_i$. First, the local Schur complement S_i is built with MUMPS. Depending whether additive Schwarz or shifted additive Schwarz preconditioner is considered, one or two LU decompositions with MUMPS are respectively necessary: one on the shifted Helmholtz system to get the local Schur complement necessary to build the preconditioner and one on the Helmholtz system to get the LU decomposition of the local matrix $A_{\mathcal{I}_i\mathcal{I}_i}$ and/or the local Schur complement. Then, some point-to-point communications between neighbor subdomains are necessary to compute interface nodes terms. An LU decomposition on the dense matrix $\bar{S}_i$ with the function CGETRF of the LAPACK package.

The second phase depends on the source/RHS term and is referred to as the solving phase. It is splitted into two main steps: the resolution of the interface nodes system (preconditioned Schur complement system) and the interior nodes system.

The preconditioned Schur complement system is resolved with the iterative method called GMRES (Generalized Minimal RESidual) (Saad, 2003). GMRES approximates the solution by the vector in the Krylov subspace with a minimum residual. The stopping criterion for the iterative method GMRES is the backward error $\epsilon = \|Ap - s\|/\|s\|$. The implemented parallel version of GMRES has been developed by Frayssé et al. (1997).

Once the interface nodes solution $\bar{p}_\Gamma$ in system (3.7) is computed, the interior nodes unknowns are concurrently computed by solving $p_{\mathcal{I}_i} = A_{\mathcal{I}_i\mathcal{I}_i}^{-1}(s_{\mathcal{I}_i} - A_{\mathcal{I}_i\Gamma}\bar{p}_\Gamma)$.

More details on the hybrid solver can be found in Haidar (2008); Sourbier et al. (2008b,a).

3.2.3.1 Complexity analysis

Memory complexity The hybrid solver allows for a significant memory saving compared to the direct one. This is quantified from the theoretical memory complexity of LU factorization of a sparse matrix and the ratio between the size of the full domain and that of the subdomains. This heuristic complexity is specific to the FD discretization used in Operto et al. (2007) and nested-dissection reordering. The local memory allocated in the hybrid method includes the memory allocated for concurrent sequential factorizations, for the storage of the local Schur complements and of the preconditioner of the iterative solver on each domain. If we consider a cubic domain of size $n \times n \times n$ and a decomposition of $k \times k \times k$ subdomains, the LU decomposition memory complexity is $\mathcal{O}(n^4/k^2 Log_2(n/k))$. The order of the matrix S_i is the subdomain interface nodes number, the complexity of which is $\mathcal{O}(n^2/k^2)$, therefore the memory complexity for the storage of the dense S_i is $\mathcal{O}(n^4/k^4)$. The size of the local preconditioner is the same as local Schur complement S_i, thus its memory complexity is $\mathcal{O}(n^4/k^4)$. The total memory is simply obtained by multiplying the local memory by the total number of subdomains k^3 and summarized in Table 3.2.

The memory requirement increases with the number of MPI processes with the direct approach because of overheads while it decreases with the hybrid method with the number of subdomains. The initial objective to overcome the burden memory cost of the direct solver is then reached thanks to the hybrid approach based on domain decomposition method.

Table 3.2: Memory complexity of the direct and hybrid solvers.

	Direct Solver	Hybrid Solver
3-D	$ov \times 4 \times n^4 Log_2(n)$	$2 \times n^4/k + (4 \times n^4 Log_2(n/k))/k$

n – Dimension size of a 3-D n^3 grid
k – Number of subdomain along each direction
ov – Memory overhead coefficient ($ov \approx 2$)

Time complexity The time complexity of an LU factorization of a sparse matrix for a 3-D FD problem is $\mathcal{O}(n^6)$ (Section 3.1.2). Therefore, the time complexity of the local LU factorizations is $\mathcal{O}((n/k)^6)$ and dramatically decreases as the number of subdomains increases. The time complexity of the LU factorization of the dense preconditioner is $\mathcal{O}((n/k)^4)$ and also dramatically decreases when k increases. The time complexity of the iterative solver is $\mathcal{O}((N_{it} N_s (n/k)^4)$ and corresponds to the computational cost of matrix-vector product of dense matrices of dimension $(n/k)^2$ performed $N_{it} \times N_s$ times where N_{it} is the number of GMRES iterations and N_s is the number of shots. Assuming a linear increase of the iteration number with the dimension of the computational domain, $\mathcal{O}(N_{it}) = \mathcal{O}(n)$, and a dense 3-D surface acquisition, $\mathcal{O}(N_s) = \mathcal{O}(n^2)$, the theoretical time complexity reduces to $\mathcal{O}(n^7/k^4)$. If we now assume that the number of subdomains in one direction k proportionnally increase with n, $\mathcal{O}(k) = \mathcal{O}(n)$ such that the workload of each processor is roughly kept constant, then, the theoretical time complexity is $\mathcal{O}(n^3)$. The assumption that the $\mathcal{O}(N_{it}) = \mathcal{O}(n)$ will be tested heuristically with several following numerical experiments with the SEG/EAGE salt and overthrust model.

Numerical analysis on SEG/EAGE salt model First, a numerical complexity analysis is performed with the SEG/EAGE salt model originally discretized on a $676 \times 676 \times 210$ FD grids with a grid interval of 20 m leading to a computational domain of dimensions $13.5 \times 13.5 \times 4.18$ km^3. The minimum and maximum velocities are 1.5 and 4.482 km/s respectively. The salt model is a slowly varying sedimentary medium within which is embedded a salt body leading to a strong velocity contrast between sediments and salt (Figure 3.9-a).

The complexity analysis is performed on the IBM Blue Gene cluster of the IDRIS computing center. Simulations are performed for frequencies 5, 7.5, 10 and 12.5 Hz. The grid interval is adapted to each frequency to satisfy a discretization criterion of four grid nodes per wavelength. The resulting grid intervals are 75 m, 50 m, 37.5 m and 30 m for the frequencies 5, 7.5, 10 and 12.5 Hz, respectively. The corresponding grid dimensions are provided in Table 3.3. The problem size increases from 2.43 to 32 millions of unknowns. The number of processors for each frequency modeling was chosen such that the size of the subdomains is kept roughly constant, namely, such that the processor workload is kept constant for all the frequencies. The size of a subdomain is in average $31 \times 31 \times 31$ which is close to the maximum size that can be tackled with 2 *Gb* of memory per MPI process. The number of processors increases with frequency from 72 to 980. All the simulating parameters are outlined in Table 3.4.

The total elapsed time, the elapsed time for the RHS-dependent phase and the number of GMRES iterations are shown in Figure 3.13 and are outlined in Table 3.4. The number

Table 3.3: SEG/EAGE salt and overthrust applications parameters.

Model	$f(Hz)$	$h(m)$	$n_x \times n_y \times n_z$	$N_u(10^6)$
Salt	5	75	$180 \times 180 \times 56$	2.43
Salt	7.5	50	$269 \times 260 \times 84$	8.25
Salt	10	37.5	$360 \times 360 \times 112$	17.16
Salt	12.5	30	$450 \times 450 \times 140$	32.02
Overthrust	3.6	150	$134 \times 134 \times 32$	1.08
Overthrust	5.4	100	$200 \times 200 \times 46$	3.11
Overthrust	7.2	75	$266 \times 266 \times 62$	6.97
Overthrust	9	60	$334 \times 334 \times 78$	12.44
Overthrust	10.8	50	$400 \times 400 \times 94$	19.38

f – Frequency
h – Spacing interval
n_x – X-dimension nodes number
n_y – Y-dimension nodes number
n_z – Z-dimension nodes number
N_u – Number of unknowns

of GMRES iterations linearly increases with frequency. The increase of the number of GMRES iterations with the frequency was expected since the number of subdomains increases with frequency to keep the processor workload constant. An increasing number of subdomains leads to a degradation of the preconditioner and hence an increase of the number of GMRES iterations. The total elapsed time and the RHS-dependent elapsed time also increase roughly linearly with frequency. To estimate the observed time complexity, I plot the quantity $N_P T/n^c$ as a function of the frequency in Figure 3.13-c). N_P is the number of processors, T is the elapsed time (the total time or the RHS-dependent time), n is the number of grid nodes in the model along the horizontal direction and c is an exponent such that the plotted curve remains centered aroung the value of 1. The expression n^c gives the complexity of the HSM algorithm. I found that the observed complexity is respectively $\mathcal{O}(n^{3.4})$ and $\mathcal{O}(n^{3.8})$ for the total run and RHS-dependent task respectively. The observed complexity of the full algorithm is $\mathcal{O}(n^{3.4})$ and is slightly higher than the theoretical one, $\mathcal{O}(n^3)$, that was found assuming that $\mathcal{O}(k) = \mathcal{O}(n)$. The time complexity on each processor defined by T/n^c is plotted in 3.13-d). I found an observed complexity of $\mathcal{O}(n^{0.3})$ and $\mathcal{O}(n^{0.7})$ respectively, which is better than the theoretical complexity of $\mathcal{O}(n)$ when we assume that $\mathcal{O}(k) = \mathcal{O}(n)$. This shows that the efficiency can be greater than 1.

Table 3.4: SEG/EAGE salt and overthrust applications parameters.

$f(Hz)$	N_P	$k_x \times k_y \times k_z$	$n_x \times n_y \times n_z$	$n_x^a \times n_y^a \times n_z^a$	$T_{tot}(s)$	$T_{RHS}(s)$	N_{it}
5	72	$6 \times 6 \times 2$	$32 \times 32 \times 33$	$6 \times 6 \times 5$	588	61	40
7.5	243	$9 \times 9 \times 3$	$31 \times 31 \times 32$	$5 \times 5 \times 6$	672	91	74
10	576	$12 \times 12 \times 4$	$31 \times 31 \times 31$	$6 \times 6 \times 6$	689	114	96
12.5	980	$14 \times 14 \times 5$	$33 \times 33 \times 30$	$6 \times 6 \times 5$	797	109	111

3.6	50	$5 \times 5 \times 2$	$30 \times 30 \times 24$	$8 \times 8 \times 8$	222	24	32
5.4	128	$8 \times 8 \times 2$	$28 \times 28 \times 31$	$12 \times 12 \times 8$	313	35	66
7.2	243	$9 \times 9 \times 3$	$32 \times 32 \times 28$	$11 \times 11 \times 11$	599	106	119
9	432	$12 \times 12 \times 3$	$30 \times 30 \times 32$	$13 \times 13 \times 9$	699	174	234
10.8	1024	$16 \times 16 \times 4$	$26 \times 26 \times 28$	$8 \times 8 \times 9$	425	175	487

f – Frequency
N_P – Number of MPI processes
n_x – X-dimension subdomains number
n_y – Y-dimension subdomains number
n_z – Z-dimension subdomains number
n_x – X-dimension nodes number
n_y – Y-dimension nodes number
n_z – Z-dimension nodes number
n_x^a – X-dimension PML nodes number
n_y^a – Y-dimension PML nodes number
n_z^a – Z-dimension PML nodes number
N_u – Number of unknowns
T_{tot} – Total elapsed time
T_{RHS} – Total RHS-dependent elapsed time
N_{it} – Number of GMRES iterations

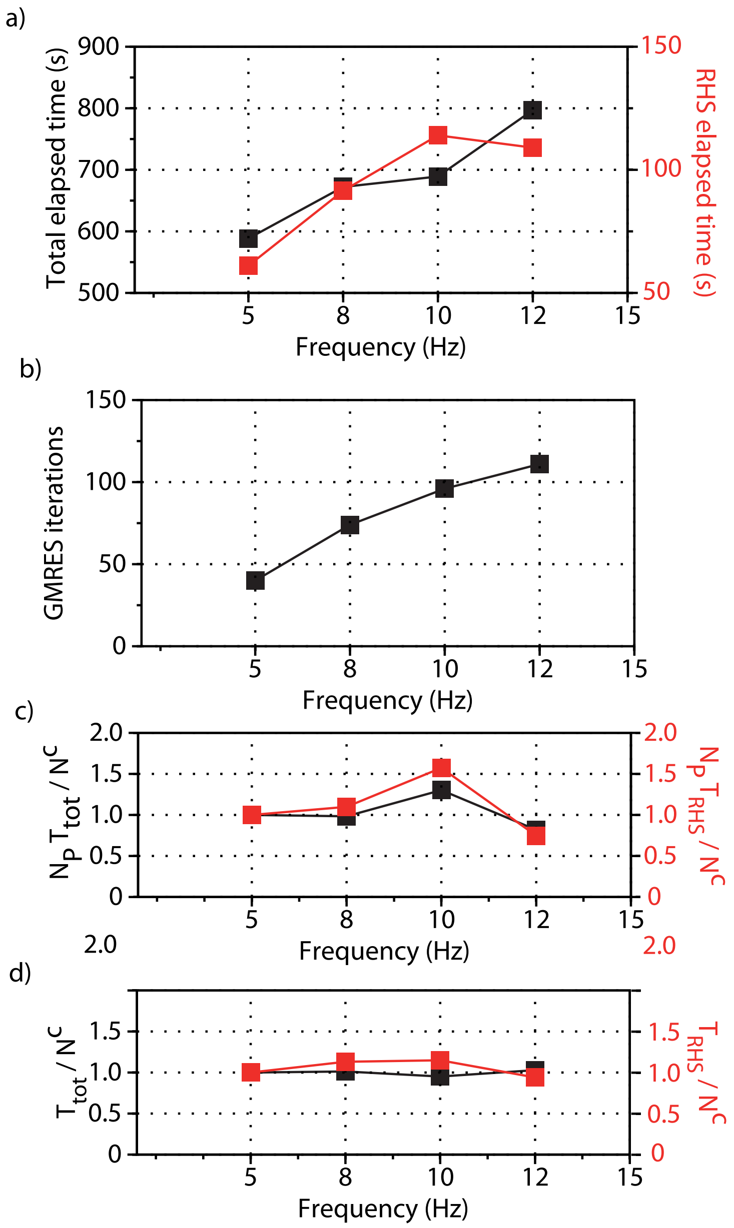

Figure 3.13: Numerical time complexity on the salt model. a) Total and RHS-dependent elapsed time. b) Number of GMRES iterations. c) Normalized numerical complexity. d) One MPI process normalized numerical complexity.

Numerical analysis on SEG/EAGE overthrust model The dimension of the overthrust model is $20 \times 20 \times 4.65\ km^3$ (Figure 3.14). The original model was discretized on a $801 \times 801 \times 187$ FD grid with a grid interval of 25 m. The minimum and maximum velocities are 2.178 km/s and 6 km/s, respectively. The overthrust model represents a thrust area above a decollement level with a weathered layer in the near surface, and therefore, provides a more structurally-complex environnement than the salt model. Then, it provides a complementay case study to assess the robustness of the hybrid solver approach for modeling in heterogeneous medium.

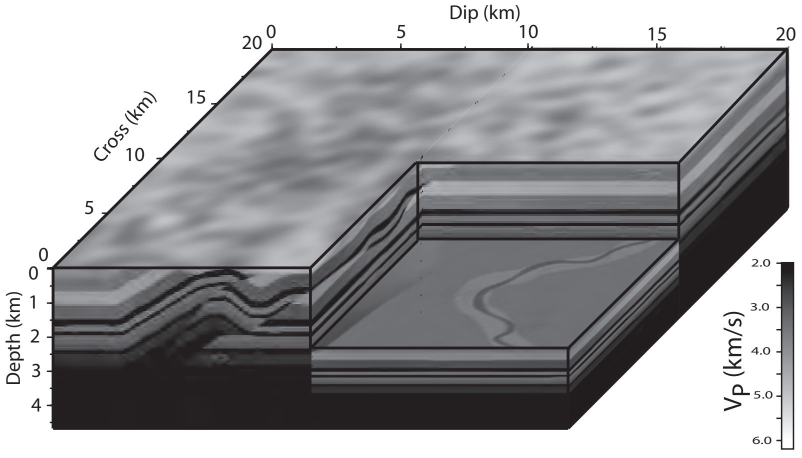

Figure 3.14: SEG/EAGE overthrsut velocity model.

Simulations in the overthrust model were performed for frequencies 3.6, 5.4, 7.2, 9 and 10.8 Hz on the same IBM Blue Gene machine. The grid interval was adapted to the modeled frequency to satisfy the criterion of four grid nodes per wavelength. The grid dimensions and the domain decompositions designed for this complexity analysis are outlined in Tables 3.3 and 3.4. The elapsed times, the number of GMRES iterations and the estimated complexities are shown in Figure 3.15. For the overthrust model, I found that the time complexity is $\mathcal{O}(n^{3.5})$ and $\mathcal{O}(n^{4.2})$ for the full run and for the RHS-dependent task respectively. The corresponding time complexities per processor have a complexity of $\mathcal{O}(N^{1.1})$ and $\mathcal{O}(N^{1.8})$ respectively. The time complexity of the full algorithm is slighlty higher than the theoretical one as for the salt model. The time complexity of the RHS-dependent phase has significantly degraded compared to that obtained with the salt model, which highlights the sensitivity of the iterative component of the hybrid solver to the structural complexity of the subsurface model.

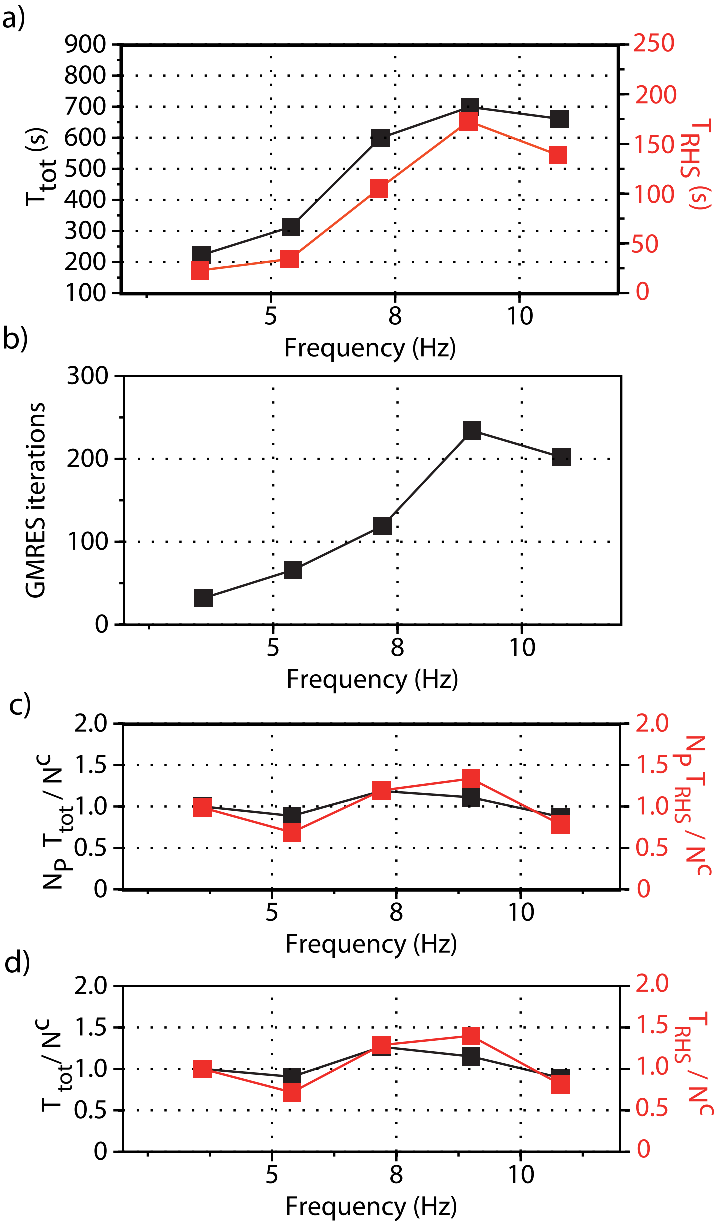

Figure 3.15: Numerical time complexity on the overthrust model. a) Total and RHS-dependent elapsed time. b) Number of GMRES iterations. c) Normalized numerical complexity. d) One MPI process normalized numerical complexity.

3.2.3.2 Analysis of GMRES convergence criterion

Contrary to the direct solvers for which solutions have a high accuracy (machine precision), the hybrid solver allows us to choose our criterion to stop GMRES iterations. The convergence criterion ϵ for the backward error on the GMRES algorithm controls the accuracy of the solution and is defined by $\epsilon = \|Ap - s\|/\|s\|$. Of course, the number of iterations and the CPU time increase in GMRES when the convergence criterion ϵ decreases. Snapshots computed in a 2-D homogeneous velocity model, illustrated in Figure 3.16 for four subdomains, shows clearly that strong reflections at the subdomains edges due to domain decomposition appear at $\epsilon = 10^{-1}$, become weak at $\epsilon = 10^{-2}$ and disappear at $\epsilon = 10^{-3}$. A value of $\epsilon = 10^{-3}$ seems to provide the best tradeoff between accuracy and iteration count. A similar analysis is performed in chapter 7 in order to assess the necessary convergence criterion for the imaging problem.

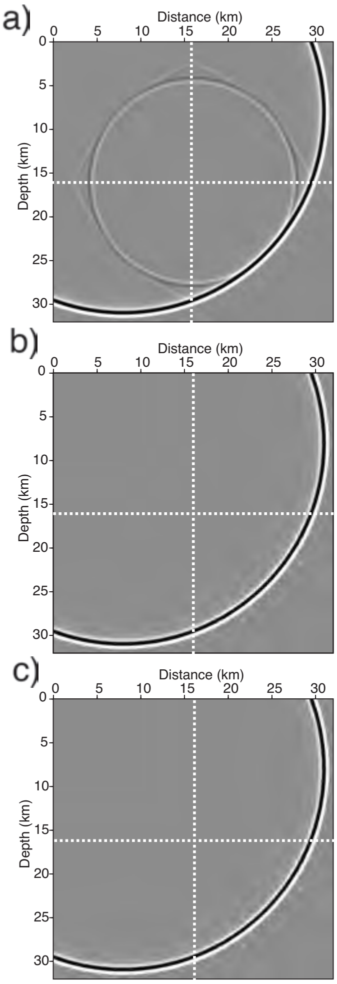

Figure 3.16: Snapshots in 2-D homogeneous velocity model. The considered domain decomposition is 2 × 2. a) $\epsilon = 10^{-1}$. b) $\epsilon = 10^{-2}$. c) $\epsilon = 10^{-3}$. The dash lines delineate the four subdomains.

More details and examples on more complex velocity models can be found Sourbier et al. (2008b).

3.2.3.3 Scalability analysis

In this section, I aim to study the scalability properties of the hybrid solver. Scalability reflects the behavior of a parallel computer program to deal with a given problem size with increasing computational resources. The scalability analysis of the hybrid solver, where the improved preconditioner is used, is performed on the SEG/EAGE overthrust and salt models. Realistic physical attenuation Q-models were considered.

Numerical analysis on SEG/EAGE salt model The scalability analysis was performed on the IBM Blue Gene (Babel) and the IBM Power 6 (Vargas). On the IBM Blue Gene, the modeled frequency is 7.5 Hz and the grid interval is 50 m. This leads to a FD grid of dimensions $270 \times 270 \times 84$ corresponding to 8.18 millions of unknowns. On the IBM Power 6, the modeled frequency is 5 Hz and the grid interval is 75 m. This leads to a FD grid of dimensions $180 \times 180 \times 56$ corresponding to 2.77 millions of unknowns. An increasing number of subdomains was used. The different domain decompositions were designed such that the subdomains geometry is as close as possible to a cube in order to minimize the number of interface nodes. The number of subdomains increases from 300 to 1944 on the IBM Blue Gene and from 50 to 432 on the IBM Power 6. The total elapsed time and the elapsed time of the RHS-dependent phase (this latter time includes the time spent in GMRES and the time required to compute interior solutions), the number of GMRES iterations and the efficiency of the full computation and of the RHS-dependent phase are plotted in Figure 3.17 and are outlined in Table 3.5.

The efficiency is given by

$$E = \frac{T_{ref} N_P^{ref}}{T_{N_P} N_P},$$

where T_{ref} is the elapsed time obtained on the smallest number of processors, N_P^{ref}, and T_{N_P} is the elapsed time obtained on N_P processors.

The first conclusion is that the number of GMRES iterations roughly linearly increases with the number of subdomains. This expected result shows the degradation of the preconditioner accuracy with the number of subdomains. Second, the elapsed time of the RHS-dependent phase remains almost constant when the number of subdomain increases. This means that, when the number of subdomains increases, the computational burden resulting from the increasing number of GMRES iterations is balanced by the computational saving on each processor provided by the decreasing of the size of the subdomains. Third, the computational cost of the RHS-independent phase, measured by the difference between the total time and the RHS-dependent time, rapidly decreases with the number of subdomains which is consistent with the numerical complexity of the LU factorization and the preconditioner building. The efficiency of the full run remains always greater than one, and increases with the number of processor when faster processors are used. In contrast, the efficiency of the RHS-dependent phase decreases with the number of processors since the elapsed time of the RHS-dependent phase remains almost constant whatever is the number of used processors. The behavior of the algorithm is slightly better on the IBM Power 6 than on the IBM Blue Gene as shown on the efficiency curves because of faster processors on the Power 6. This highlights the good scalability of the algorithm resulting from a limited number of communications.

In the case of a large number of RHS, the best strategy is clearly to use a number of processors as small as possible to maintain the efficiency of the RHS-dependent phase close to 1. Groups of shots (i.e., RHS) can be distributed over groups of N_P processors where N_P denotes the number of processors required to perform one domain decomposition such that each group of N_P processors proceed a limited number of shots (if N_P^{tot} is the total number of processors, the number of RHS per groups of N_P processors will be $N_s N_P / N_P^{tot}$). In contrast, in the case of only one RHS as it can be the case when source encoding techniques (chapter 8) are used, the best strategy is to use a significant number of processors to make negligible the computational cost of the RHS-independent preprocessing step.

Table 3.5: Scalability analysis for the 3-D salt and overthrust models.

Computer	Model	N_P	$k_x \times k_y \times k_z$	$n_x \times n_y \times n_z$
Babel	Salt	300	$10 \times 10 \times 3$	$29 \times 29 \times 32$
Babel	Salt	484	$11 \times 11 \times 4$	$26 \times 26 \times 26$
Babel	Salt	980	$14 \times 14 \times 5$	$21 \times 21 \times 20$
Babel	Salt	1620	$18 \times 18 \times 5$	$16 \times 16 \times 20$
Babel	Salt	1944	$18 \times 18 \times 6$	$16 \times 16 \times 16$
Babel	Overthrust	243	$9 \times 9 \times 3$	$32 \times 32 \times 28$
Babel	Overthrust	576	$12 \times 12 \times 4$	$23 \times 23 \times 21$
Babel	Overthrust	980	$14 \times 14 \times 5$	$20 \times 20 \times 16$
Babel	Overthrust	1536	$16 \times 16 \times 6$	$18 \times 18 \times 12$
Babel	Overthrust	1944	$18 \times 18 \times 6$	$16 \times 16 \times 12$
Vargas	Salt	50	$5 \times 5 \times 2$	$38 \times 38 \times 33$
Vargas	Salt	72	$6 \times 6 \times 2$	$32 \times 32 \times 33$
Vargas	Salt	192	$8 \times 8 \times 3$	$24 \times 24 \times 22$
Vargas	Salt	400	$10 \times 10 \times 4$	$19 \times 19 \times 17$
Vargas	Salt	432	$12 \times 12 \times 3$	$16 \times 16 \times 22$
Vargas	Overthrust	72	$6 \times 6 \times 2$	$35 \times 35 \times 28$
Vargas	Overthrust	98	$7 \times 7 \times 2$	$30 \times 30 \times 28$
Vargas	Overthrust	243	$9 \times 9 \times 3$	$24 \times 24 \times 20$
Vargas	Overthrust	363	$11 \times 11 \times 3$	$20 \times 20 \times 20$
Vargas	Overthrust	432	$12 \times 12 \times 3$	$18 \times 18 \times 20$

$n_x^a \times n_y^a \times n_z^a$	$T_{tot}(s)$	$T_{RHS}(s)$	N_{it}	E_{tot}	E_{RHS}
$10 \times 10 \times 6$	520	77	83	1	1
$8 \times 8 \times 10$	255	48	114	1.26	1.00
$12 \times 12 \times 8$	97	30	202	1.64	0.79
$9 \times 9 \times 8$	88	58	472	1.09	0.25
$9 \times 9 \times 6$	56	30	612	1.42	0.39
$11 \times 11 \times 11$	599	106	119	1	1
$5 \times 5 \times 11$	132	43	229	1.92	1.00
$7 \times 7 \times 9$	96	59	534	1.54	0.45
$11 \times 11 \times 5$	110	89	842	0.86	0.19
$11 \times 11 \times 5$	101	85	1124	0.74	0.16
$5 \times 5 \times 5$	224	23	30	1	1
$6 \times 6 \times 5$	129	14	36	1.21	1.18
$6 \times 6 \times 5$	38	7	68	1.53	0.88
$5 \times 5 \times 6$	13	4.6	102	2.15	0.64
$6 \times 6 \times 5$	12	5	129	2.18	0.54
$5 \times 5 \times 5$	125	15.1	49	1	1
$5 \times 5 \times 5$	90	11.2	60	1.	0.99
$8 \times 8 \times 7$	33	9.8	135	1.13	0.46
$10 \times 10 \times 7$	24	10.5	220	1.05	0.29
$8 \times 8 \times 7$	19	10.1	285	1.10	0.25

N_P – Number of MPI processes
n_x – X-dimension subdomains number
n_y – Y-dimension subdomains number
n_z – Z-dimension subdomains number
n_x – X-dimension nodes number
n_y – Y-dimension nodes number
n_z – Z-dimension nodes number
n_x^a – X-dimension PML nodes number
n_y^a – Y-dimension PML nodes number
n_z^a – Z-dimension PML nodes number
N_u – Number of unknowns
T_{tot} – Total elapsed time
T_{RHS} – Total RHS-dependent elapsed time
N_{it} – Number of GMRES iterations
E_{tot} – Efficiency based on the total elapsed time
E_{RHS} – Efficiency based on the RHS-dependent elapsed time

Numerical analysis on SEG/EAGE overthrust model The same scalability analysis is performed on the SEG/EAGE overthrust model. On the IBM Blue Gene, the modeled frequency is 7.5 Hz and the grid interval is 75 m. This leads to a FD grid of dimensions

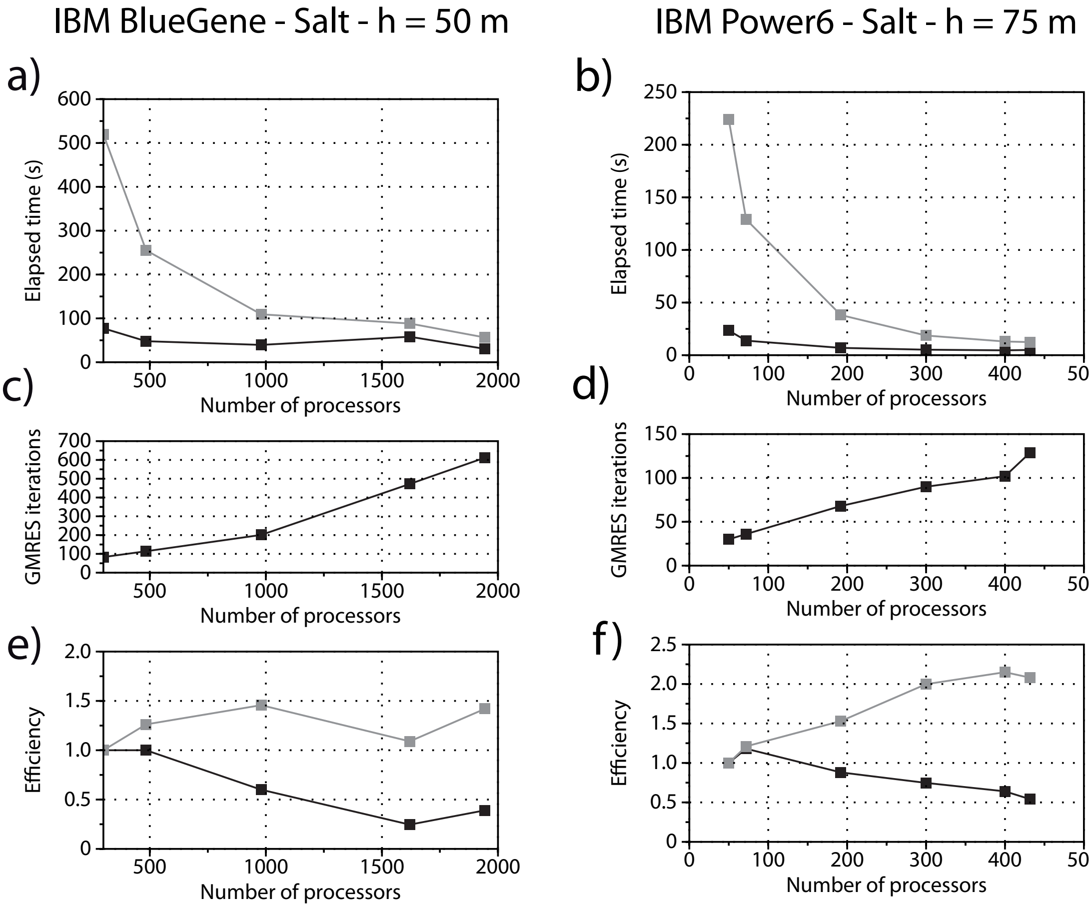

Figure 3.17: Scalability analysis for the 3-D salt model on Babel: a) Elapsed time. b) GMRES iterations. c) Efficiency. and Vargas: d) Elapsed time. e) GMRES iterations. f) Efficiency. *Black line* for the total and *gray line* for the RHS-dependent time/efficiency.

266 × 266 × 62 corresponding to 6.2 millions of unknowns. On the IBM Power 6, the modeled frequency is 5.4 Hz and the grid interval is 100 m. This leads to a FD grid of dimensions 200 × 200 × 46 corresponding to 2.89 millions of unknowns. The number of subdomains increases from 243 to 1944 on the IBM Blue Gene and from 72 to 432 on the IBM Power 6. The total elapsed time and the elapsed time of the RHS-dependent phase, the number of GMRES iterations and the efficiency of the full computation and of the RHS-dependent phase are plotted in Figure 3.18 and are outlined in Table 3.5. A similar behavior of the algorithm than for the salt model is observed: the number of GMRES iteration increases with the number of subdomains and the resulting computational burden is balanced by the decrease of the subdomain dimensions. The higher structural complexity of the overthrust model compared to the salt model is however illustrated by the fact that the elapsed times increase more significantly after a critical number of processor in the overthrust-model case compared to the salt-model case. This is more obvious on the IBM Blue Gene where the processors have a slower clock frequency.

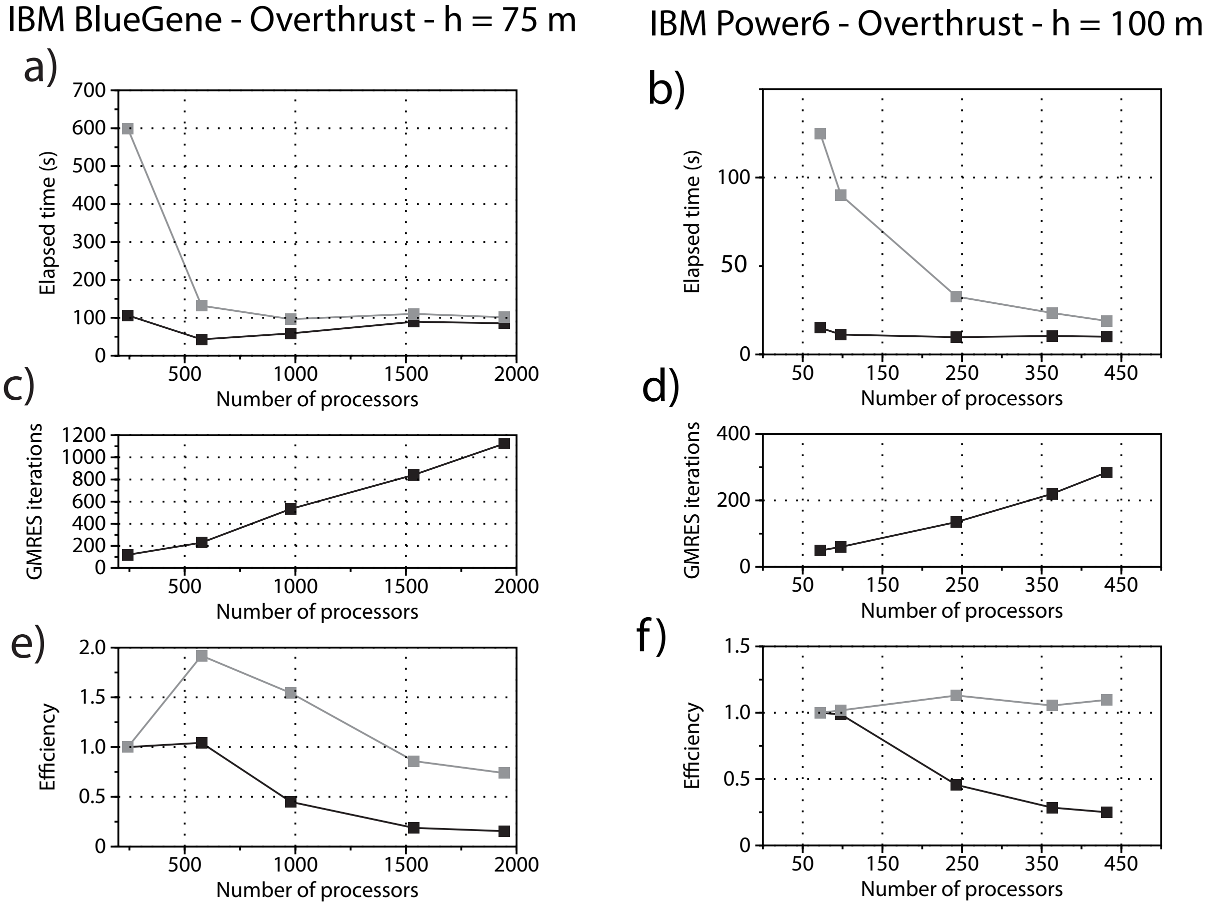

Figure 3.18: Scalability analysis for the 3-D overthrust model on Babel: a) Elapsed time. b) GMRES iterations. c) Efficiency. and Vargas: d) Elapsed time. e) GMRES iterations. f) Efficiency. *Black line* for the total and *gray line* for the RHS-dependent time/efficiency.

3.3 Practical implementation issues

Before going to the numerical examples section in order to validate and compare the two different approaches involved in the seismic wave modeling, i.e. direct and hybrid solver, I will be interested in some practical issues related to particular boundary conditions that allow to simulate 2-D wave propagation with a 3-D software and to the implementation of the source.

3.3.1 Mimicking 2-D wave propagation with the 3-D software

Two-dimensional experiments can be designed considering 2.5-D velocity models (laterally invariant in the y-direction) and an infinite line source in the y direction. The infinite line source in the y direction was implemented on a limited computational domain in the y direction using periodic boundary conditions on the two faces of the model corresponding to $y = 0$ and $y = y_{max}$.

The periodic boundary conditions that were implemented are

$$\left[\frac{\partial P}{\partial y}\right]_{y=-h/2, y_{max}+h/2} = 0.$$

They are applied on two virtual ghost faces located outside the computational domain at positions $y = -h/2$ and $y = y_{max} + h/2$ where h stands for the grid interval.

We applied 3-D and 2-D FWI to a dip section of the overthrust model (Aminzadeh et al., 1997), discretized on a 801×187 grid with a grid spacing $h = 25\ m$. For the 3-D application, the dip section of the overthrust model was duplicated 3 times in the y direction leading to a 3-D $801 \times 3 \times 187$ finite differences grid. A 2-D wavefield computed in this 2.5-D model with the above mentioned boundary conditions is shown in Figure 3.19. PML absorbing boundary conditions are set on the 4 edges of the 2-D model.

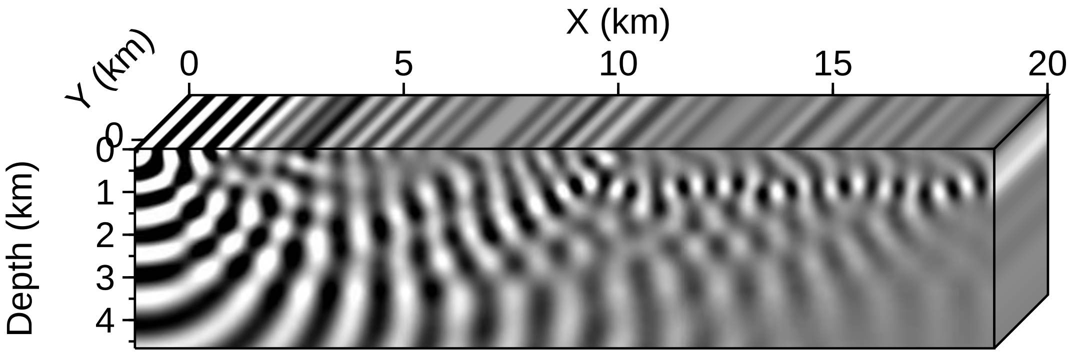

Figure 3.19: A monochromatic wavefield obtained when mimicking 2-D seismic wave propagation with the 3-D code. The wavefield was computed on a velocity section extracted from SEG/EAGE overthrust model at a frequency $f = 5Hz$.

3.3.2 Source implementation at arbitrary position

In this paragraph, I am interested in a usual problem in finite differences seismic modeling. The issue consists in implementing a source if its position does not coincide with a grid node. The simplest solution is to put the source on the nearest grid node. This is efficient and does not introduce big discrepancies if the spacing interval is sufficiently small. However, for an efficient frequency domain full waveform inversion implementation, the computational grid is adapted to frequency which leads for small frequencies to coarse grids. A better approach is necessary to reach an acceptable solution. This consists in spreading the source term on the surrounding grid nodes. Hicks (2002) has proposed an efficient distribution approach. Consider the elementary Dirac (explosive) source. The source term is replaced by its truncated approximation, the cardinal sine. The cardinal sine is located at the source position and the appropriate weights are put in the surrounding nodes. In addition, if the source is positioned on a grid node, the cardinal sine verifies that zeros coincide with the grid nodes except for the source position. Note that the length of the spreading is limited to a few grid nodes using a taper function. Otherwise, the spreading is computationally too expensive, especially if many sources are involved such as in imaging. Hicks (2002) has also detailed how to implement the dipole (force) source and the ghost effect at the free surface. Note that the same strategy can be used to extract the data at the receiver positions when these latter do not coincide with grid nodes.

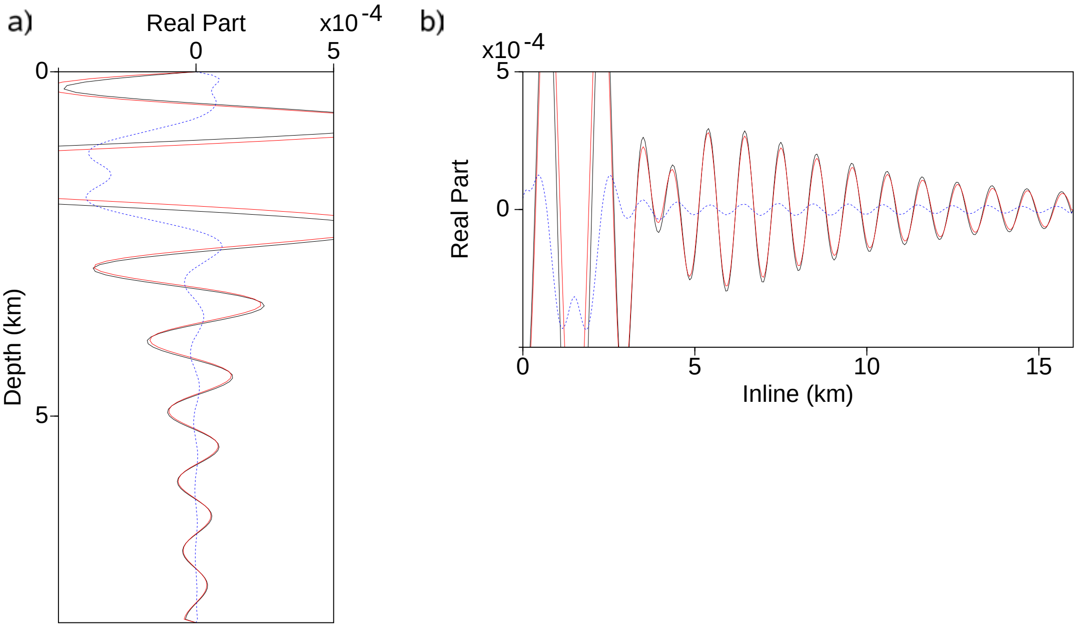

Figure 3.20: Validation of the distributed source implementation on a homogeneous medium. The numerical solution in the fine grid is in *solid black* and in the coarse grid in *solid red*. Differences are in *dashed blue*. a) Dip profile. b) Inline profile.

The implementation of the distributed source as in (Hicks, 2002) has been validated on a homogeneous velocity model taking account of the free surface effects. Figure 3.20 illustrates the comparison between the solution computed in a fine FD grid where the source is positioned on a grid node (reference solution) and in a coarse grid where the source is spread. The fit between the two numerical solutions is quite good except near the source position.

In order to validate the two seismic wave modeling approaches and compare between their memory and CPU time requirements, numerical examples on either homogeneous or heterogeneous (SEG/EAGE overthrust and salt models) velocity models have been performed.

The validation of the 3-D FD stencil developed in the chapter 2 is detailed in Operto et al. (2007).

3.4 Numerical examples

The numerical examples based on hybrid solver were performed on the IDRIS IBM Power6 cluster. Only 3.5 *Gbytes* of memory per MPI process are allowed. Those based on the direct solver were performed on a distributed-memory architecture cluster located at Geoazur Institute and composed of 18 nodes, each of them includes two quad-core 2.7-GHz Opteron processors with 64 *Gbytes* of shared memory. This allows to reduce the number of processors and therefore to minimize the overhead memory.

3.4.1 Validation against the analytical solution in homogeneous medium

In order to validate the modeling engine, the numerical direct and hybrid solver solutions are compared with the analytical Green's function and with each other. The analytical Green's function $G(r,t)$ is given by

$$G(r,t) = \frac{1}{4\pi c^2}\frac{\delta(t-\frac{r}{c})}{r}, \tag{3.13}$$

where t denotes the time, c the wave velocity and r the propagated distance (Aki and Richards, 1980, chapter 4).

Note that two convergence criteria were tested for the hybrid solver, $\epsilon = 10^{-2}$ and 10^{-3}. The P-wave velocity is 4 km/s and the density is 2000 kg/m^3. The frequency is 20 Hz. The grid size is $41 \times 41 \times 201$ and the grid interval is 50 m. The PML layer is discretized with ten grid nodes and hence spans over two and a half wavelengths. For this simulation, I used 32 2.4-GHz-cores with 2 *Gbytes* of memory per core, either for direct or hybrid solver.

Examples of mono-frequency pressure wavefields and residuals between different solutions are shown in Figures 3.21 to 3.23. Some horizontal profiles extracted from the pressure wavefields (Figures 3.24 and 3.25) are compared after correction for geometrical spreading ($\frac{1}{r}$ in Equation (3.13)). The fit between Green's function, direct solver and hybrid solver solution at $\epsilon = 10^{-3}$ is quite good (Figures 3.24-a), -c) and 3.25-b)). However, there are some discrepancies with hybrid solver solution at $\epsilon = 10^{-2}$ (Figures 3.24-b) and 3.25-a)).

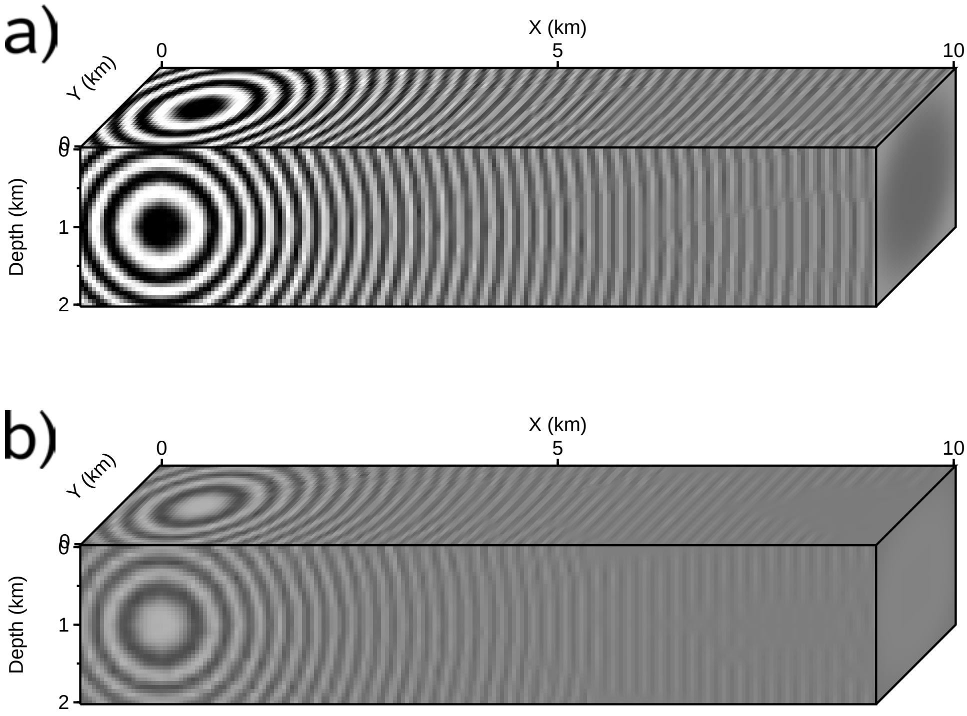

Figure 3.21: Validation on homogeneous medium. a) Analytic Green's function. b) Differences with numerical Green's function obtained with direct solver.

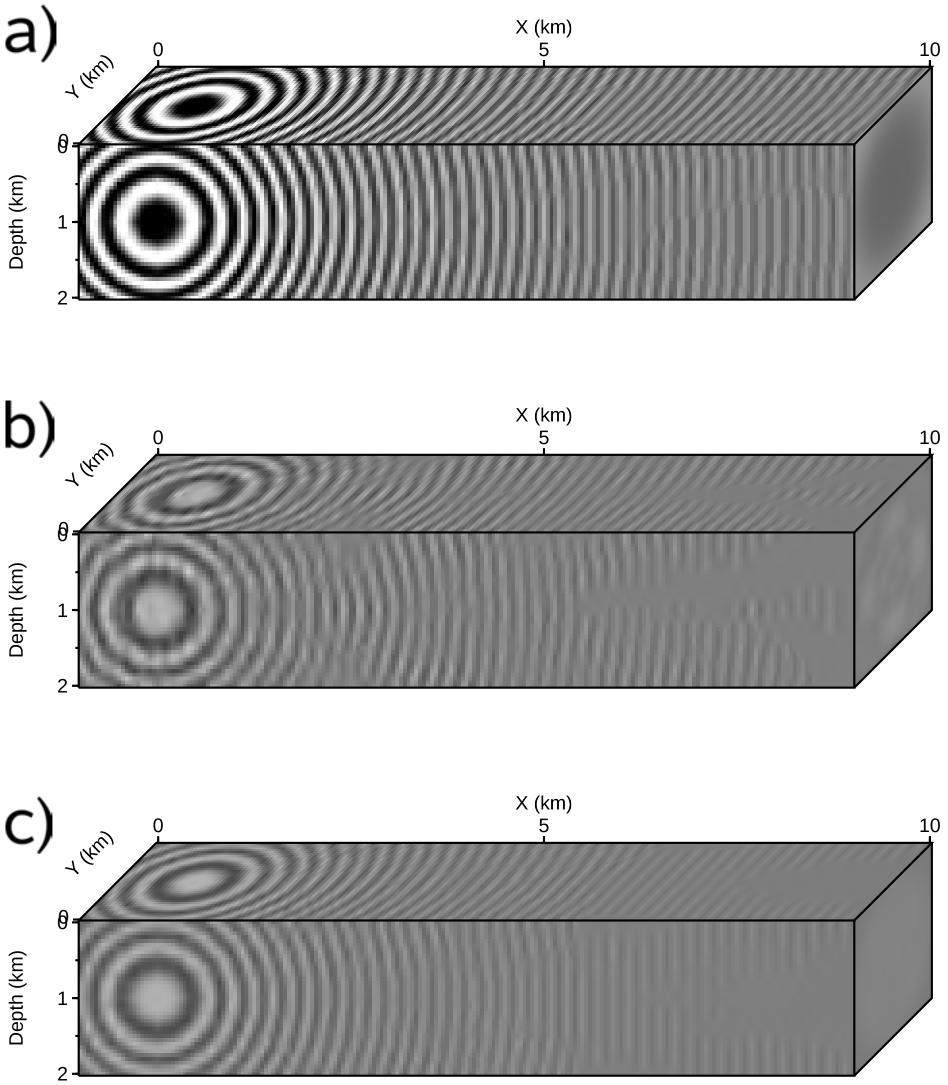

Figure 3.22: Validation on homogeneous medium. a) Analytic Green's function. b) Differences with numerical Green's function obtained with hybrid solver at a convergence criterion $\epsilon = 10^{-2}$. c) Differences with numerical Green's function obtained with hybrid solver at a convergence criterion $\epsilon = 10^{-3}$.

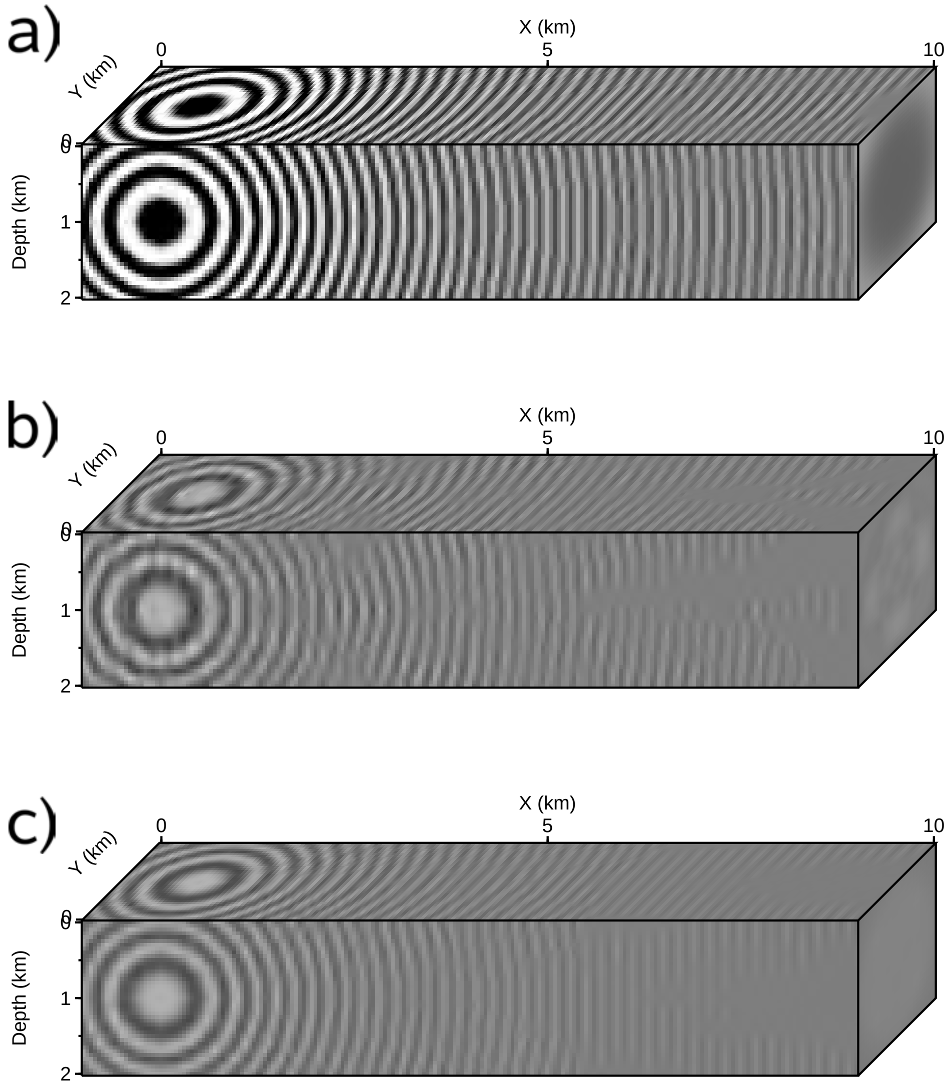

Figure 3.23: Validation on homogeneous medium. a) Numerical Green's function obtained with direct solver. b) Differences with numerical Green's function obtained with hybrid solver at a convergence criterion $\epsilon = 10^{-2}$. c) Differences with numerical Green's function obtained with hybrid solver at a convergence criterion $\epsilon = 10^{-3}$.

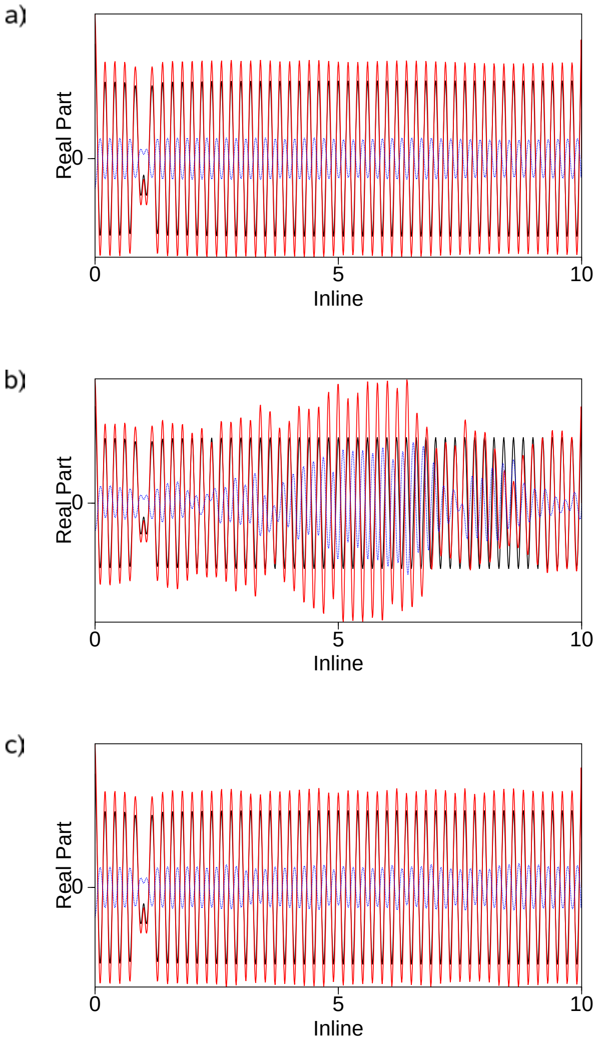

Figure 3.24: Validation on homogeneous medium. Quantitative comparison between dip profiles extracted at $(X = 2km, Y = 2km)$. a) Analytic (*solid black*) and direct solver (*solid red*). b) Analytic (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-2}$ (*solid red*). c) Analytic (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-3}$ (*solid red*). Differences are in *dashed blue.*

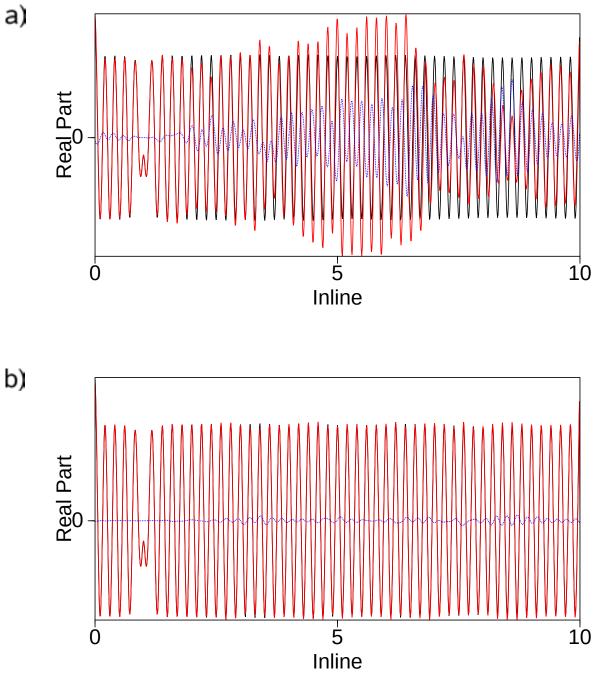

Figure 3.25: Validation on homogeneous medium. Quantitative comparison between dip profiles extracted at $(X = 2km, Y = 2km)$. a) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-2}$ (*solid red*). b) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-3}$ (*solid red*). Differences are in *dashed blue.*

3.4.2 SEG/EAGE overthrust velocity model

The second set of tests were done on SEG/EAGE overthrust velocity model. The numerical direct and hybrid solver at $\epsilon = 10^{-2}$ and 10^{-3} solutions are compared with each other. The frequency is 5 Hz. The grid size is $61 \times 265 \times 265$ and the grid interval is 75 m leading to 6 grid nodes per wavelength. The PML layer is discretized with five grid nodes and hence spans over almost one wavelength. The problem involves 5.3 millions unknowns.

Mono-frequency pressure wavefields and residuals between different solutions are shown in Figures 3.26. Some horizontal and vertical profiles extracted from the pressure wavefields (Figures 3.27 and 3.28) are compared. The fit between direct solver and hybrid solver solution at $\epsilon = 10^{-3}$ is quite good (Figures 3.27-b) and 3.28-b)). However, there are some discrepancies with hybrid solver solution at $\epsilon = 10^{-2}$ (Figures 3.27-a) and 3.28-a)).

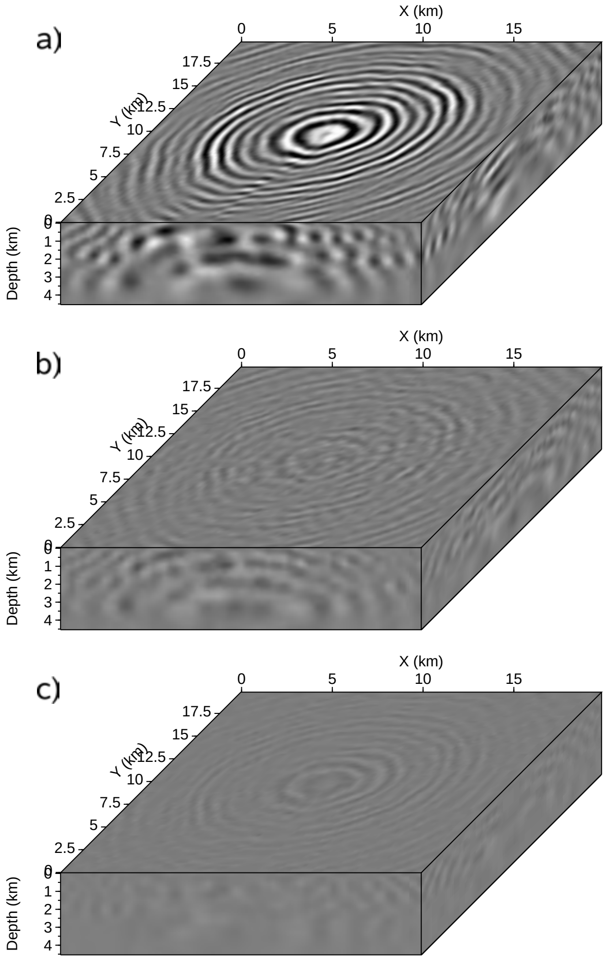

Figure 3.26: Validation on SEG/EAGE overthrust velocity model. a) Numerical solution with direct solver. b) Differences with numerical solution with hybrid solver at a convergence criterion $\epsilon = 10^{-2}$. d) Differences with numerical solution with hybrid solver at a convergence criterion $\epsilon = 10^{-3}$.

The Table 3.6 gathers the CPU time and memory storage required for the simulations based on either the direct or the hybrid solver. Results show that as expected the hybrid

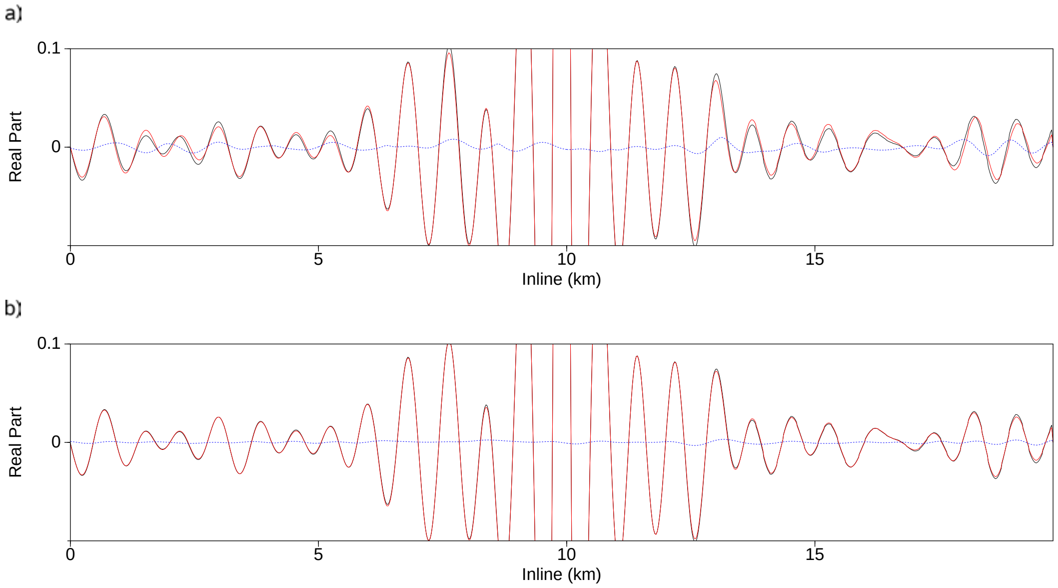

Figure 3.27: Validation on SEG/EAGE overthrust velocity model. Quantitative comparison between Xline profiles extracted at $(Z = 2.25km, X = 9.9km)$. a) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-2}$ (*solid red*). b) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-3}$ (*solid red*). Differences are in *dashed blue*.

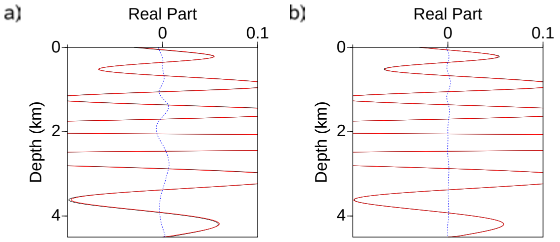

Figure 3.28: Validation on SEG/EAGE overthrust velocity model. Quantitative comparison between dip profiles extracted at $(X = 9.9km, Y = 9.9km)$. a) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-2}$ (*solid red*). b) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-3}$ (*solid red*). Differences are in *dashed blue*.

solver allows mitigating the memory requirement of LU decomposition and the elapsed time in the preprocessing phase (LU decomposition and LU decomposition on subdomains and Schur complement and preconditioner building for the direct and hybrid solvers respectively) while the direct solver is much more efficient in the solving phase and therefore for multi-RHS simulations.

Table 3.6: Comparison between the time and memory requirements of the simulations on the overthrust model based on either direct or hybrid solver.

	Hybrid Solver ($\epsilon = 10^{-3}$)	Direct Solver
N_p	243	96
$Mem(Gb)$	170	245
$T_p(s)$	38	1900
$T_s(s)$	33	3

N_p – Number of MPI processes
Mem – Total memory
T_p – Elapsed time for preprocessing (LU decomposition and LU decomposition on subdomains and Schur complement and preconditioner building for the direct and hybrid solvers respectively)
T_s – Elapsed time for solving one RHS

3.4.3 SEG/EAGE salt velocity model

The final set of tests were done on SEG/EAGE salt velocity model. The numerical direct and hybrid solver at $\epsilon = 10^{-2}$ and 10^{-3} solutions are compared with each other. The frequency is 5 Hz. The grid size is 69 × 225 × 225 and the grid interval is 60 m leading to 6 grid nodes per wavelength. The PML layer is discretized with five grid nodes and hence spans over almost one wavelength. The problem involves 4.3 million unknowns.

Mono-frequency pressure wavefields and residuals between different solutions are shown in Figures 3.29. Some horizontal and vertical profiles extracted from the pressure wavefields (Figures 3.30 and 3.31) are compared. The fit between direct solver and hybrid solver solution at $\epsilon = 10^{-3}$ is quite good (Figures 3.30-b) and 3.31-b)). However, there are some discrepancies with hybrid solver solution at $\epsilon = 10^{-2}$ (Figures 3.30-a) and 3.31-a)).

As for the previous example, CPU time and memory required for the simulations are given in Table 3.7. The same discussion apply for the salt model and highlight the previous conclusions.

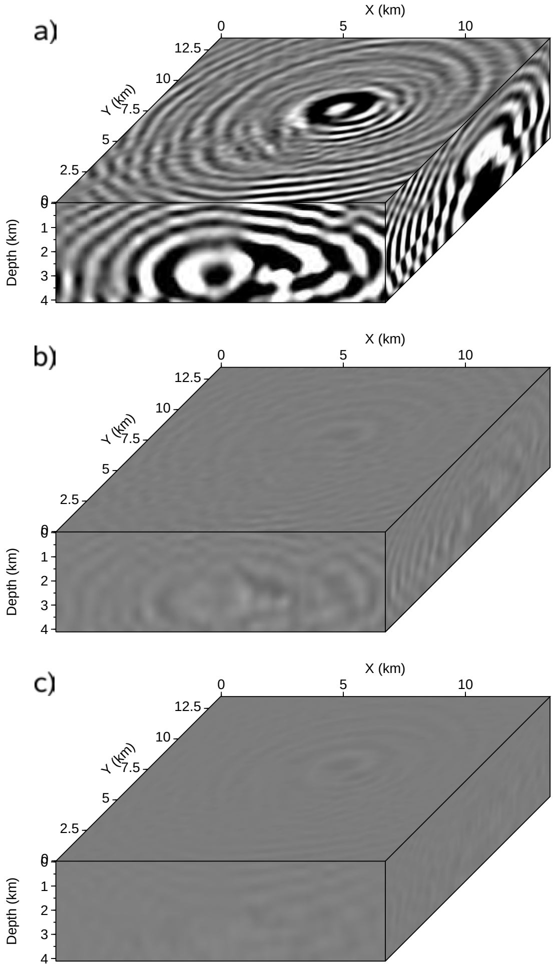

Figure 3.29: Validation on SEG/EAGE salt velocity model. a) Numerical solution with direct solver. b) Differences with numerical solution with hybrid solver at a convergence criterion $\epsilon = 10^{-2}$. d) Differences with numerical solution with hybrid solver at a convergence criterion $\epsilon = 10^{-3}$.

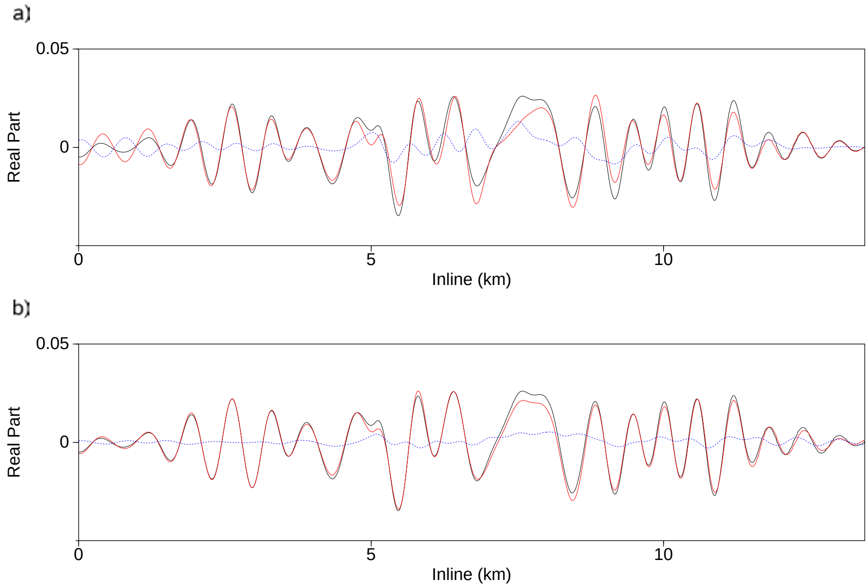

Figure 3.30: Validation on SEG/EAGE salt velocity model. Quantitative comparison between Xline profiles extracted at $(Z = 2.04km, X = 6.72km)$. a) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-2}$ (*solid red*). b) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-3}$ (*solid red*). Differences are in *dashed blue*.

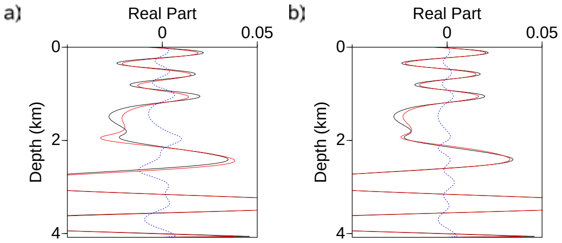

Figure 3.31: Validation on SEG/EAGE salt velocity model. Quantitative comparison between dip profiles extracted at $(X = 6.72km, Y = 6.72km)$. a) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-2}$ (*solid red*). b) Direct solver (*solid black*) and hybrid solver at a convergence criterion $\epsilon = 10^{-3}$ (*solid red*). Differences are in *dashed blue*.

Table 3.7: Comparison between the time and memory requirements of the simulations on the salt model based on either direct or hybrid solver.

	Hybrid Solver ($\epsilon = 10^{-3}$)	Direct Solver
N_p	243	96
$Mem(Gb)$	55	213
$T_p(s)$	31	2573
$T_s(s)$	24	6

N_p – Number of MPI processes
Mem – Total memory
T_p – Elapsed time for preprocessing (LU decomposition and LU decomposition on subdomains and Schur complement and preconditioner building for the direct and hybrid solvers respectively)
T_s – Elapsed time for solving one RHS

3.5 Discussion on seismic modeling for seismic imaging

The final discussion is devoted to the comparison of the different approaches considered for seismic wave modeling in the prospect of full waveform inversion where thousands of sources are involved. Different properties or criteria are discussed such as the CPU time complexity, especially for the case of multi-RHS resolution, the memory complexity, the scalability of the approach on large parallel computing platforms and the robustness of the method. A similar discussion was the topic of the publication Virieux et al. (2009).

In addition to direct and hybrid frequency domain seismic modeling approaches, explicit scheme time domain method has been implemented in parallel. The implementation is based on a domain decomposition method, on a forth-order in space $\mathcal{O}(x^4)$ and second-order in time $\mathcal{O}(t^2)$ finite differences scheme (Levander, 1988) and C-PML boundary conditions (Komatitsch and Martin, 2007). The iterative method alternative as well is discussed basing my discussion on the works of Erlangga and collaborators (Vuik et al., 2003; Erlangga, 2005; Riyanti et al., 2006; Plessix, 2007; Erlangga and Herrmann, 2008).

Table 3.8 gathers the theoretical memory and CPU time complexities of the different discussed approaches. Note that the memory complexity of either the time domain or the iterative solver corresponds to the memory necessary to store the multiple source solutions while the direct and hybrid solvers present an additional memory cost resulting from the LU decomposition (storing L and U matrices in the core memory), the Schur complement and its preconditioner building (preprocessing phase).

Table 3.8: Memory and CPU time complexities of the different methods used in seismic wave modeling.

	DSM	HSM	TDM	ISM
M_C	$\mathcal{O}(n^4Log_2(n)) + \mathcal{O}(n^3N_s)$	$\mathcal{O}(n^4/k) + \mathcal{O}(n^3N_s)$	$\mathcal{O}(n^3N_s)$	$\mathcal{O}(n^3N_s)$
T_C	$\mathcal{O}(n^6)/S_{LU} + \mathcal{O}(n^4N_s)/S_S$	$\mathcal{O}((n/k)^4N_sN_{it})$	$\mathcal{O}((n^3N_tN_s)/k^3)$	$\mathcal{O}(n^3N_{it}N_s)$

DSM – Direct solver method
HSM – Hybrid solver method
TDM – Time solver method
ISM – Iterative solver method
M_C – Memory complexity
T_C – CPU time complexity
k – Number of subdomains in one direction, i.e. associated decomposition is $k \times k \times k$
N_s – Number of shots
N_{it} – Number of iterations ($\approx \mathcal{O}(n)$)
N_t – Number of time samples/steps ($\approx \mathcal{O}(n^2)$)
S_{LU} – Speed up of parallel LU decomposition
S_S – Speed up of parallel LU solve

The memory complexities suggest that the iterative solver and time domain methods are much more less memory demanding than the direct and hybrid solvers for single source modeling. This explains that for multi-RHS simulations these methods are generally parallelized on the shots, i.e. one shot is allocated to one MPI process, since the memory requirements are affordable on nowadays computers for realistic seismic targets and low frequencies (5-10 Hz). This parallelization approach is referred to coarse grain approach. This clearly implies that these approaches are very scalable and have almost perfect speed up and efficiency.

In contrast, the memory requirements of either the direct or hybrid solver impose a fine grain parallelization approach. This means that the parallelization is performed through a domain decomposition independent of the RHS, i.e. parallel LU decomposition for the direct solver and domain decomposition method for the hybrid solver. As mentioned in the respective sections, the direct solver presents poor scalability in parallel platforms due to the LU decomposition. The hybrid solver is better scalable but does not reach the time domain or iterative solver scalability. The two parallelization approaches of course can be mixed.

The CPU time complexity of the iterative and the hybrid solvers depend on the convergence rate of the iterative solving. These complexities strongly depend on the preconditioning efficiency and therefore the dependency of the number of iterations N_{it} with respect to the size of the problem and the frequency. Recently, Erlangga and Herrmann (2008) have proposed an iterative approach based on a multi-grid shifted Laplace preconditioning strategy to remedy to this limitation. On the other hand, the direct solver is closely depending on the LU decomposition which suffers from limitations related to matrix conditioning, which degrades with increasing problem size, and poor scalability. The explicit time-marching scheme time complexity is well settled and depends only on the time and space discretization rules.

The iterative and the time domain solvers have theoretically the best CPU time performances. Nevertheless, it is still not clear which solver is the best choice, especially for multi-RHS resolution, on small and medium size problems (applications at low frequencies). To better assess the different discussed aspects related to CPU time, simulations on the 7-Hz overthrust model ($71 \times 275 \times 275$) were performed with the direct solver MUMPS, the hybrid and the time domain solvers. The direct and hybrid solver were performed on 192 MPI processes while the time domain simulation was performed in sequential mode, i.e. one MPI process. To compare between the different approaches, I consider that the time domain solver is perfectly scalable and divide the sequential time by the number of MPI processes used for the other simulations which is 192 MPI processes. CPU time requirements are gathered in Table 3.9.

For one source, the best choice is the time domain solver (9 s). The direct solver leads to the biggest CPU computation time (≈ 2550 s). This is due to the LU decomposition complexity since the time for solving is only 3 s. The hybrid solver has a mitigated computation time (67 s). The elapsed time in the preprocessing phase (LU decomposition and Schur complement/preconditioner building) is 55 s and 12 s in the solving phase.

For multi-RHS resolutions, the behavior is not completely the same. Figure 3.32 plots the respective computation time curves for the different solvers with respect to the number of sources. The slope of the curves is related to the RHS-dependant solving phase of each approach, i.e. 3 s for the direct, 12 s for the hybrid and 9 s for the time domain solver. The Y-intercept or origin ordinate of curves represents the preprocessing phase computation time. The time domain solver is better that the hybrid solver independently of the number of the sources. However, the direct solver curve intercepts the hybrid and time domain solvers curves at $N_s = 277$ and $N_s = 425$ respectively, where N_s is the number of

Table 3.9: CPU time for preprocessing and solving phases for the 7-Hz overthrust model.

	DSM	HSM	TDM
N_p	192	192	1
$T_P(s)$	2550	55	-
$T_S(s)$	3	12	1670

DSM – Direct solver method
HSM – Hybrid solver method
TDM – Time solver method
N_p – Number of MPI processes
T_P – CPU time for preprocessing phase
T_S – CPU time for solving phase

sources. This clearly proves that the direct solver is very efficient for multi-RHS resolution applications of small dimension such as in seismic imaging based on full waveform inversion at low frequencies.

The Figure 3.33 plots the same quantities for a different number of MPI processes. This latter decreases the slope of the three curves by the same ratio and does not change the Y-intercept. The figure shows that even if the time domain solver is better than the hybrid one, the difference between respective slopes is not so high (9 s and 12 s respectively) and implies that any improvement in the convergence of the hybrid solver can lead to a competitive approach. The direct solver is the most appropriate choice for medium size applications due to its high multi-RHS resolution efficiency.

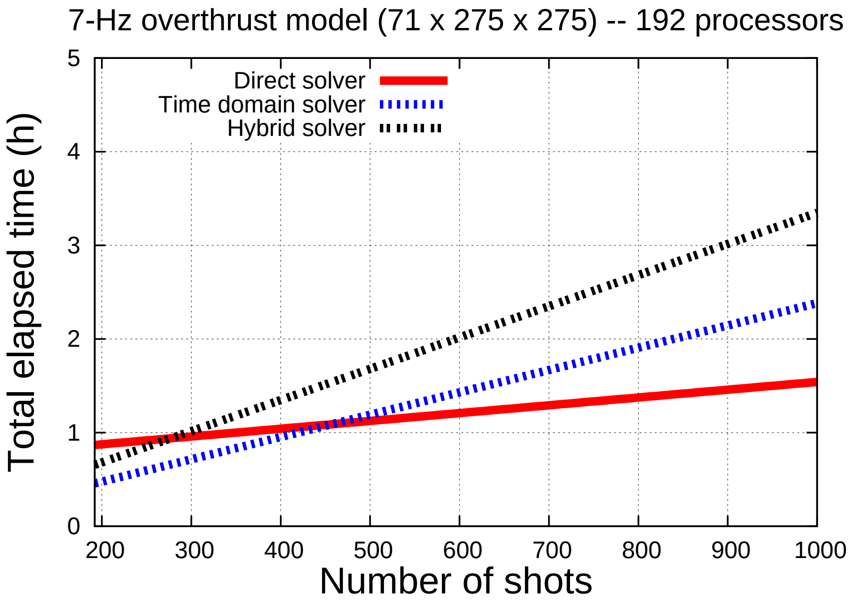

Figure 3.32: Comparison between direct, hybrid and time domain approaches on the 7-Hz overthrust model.

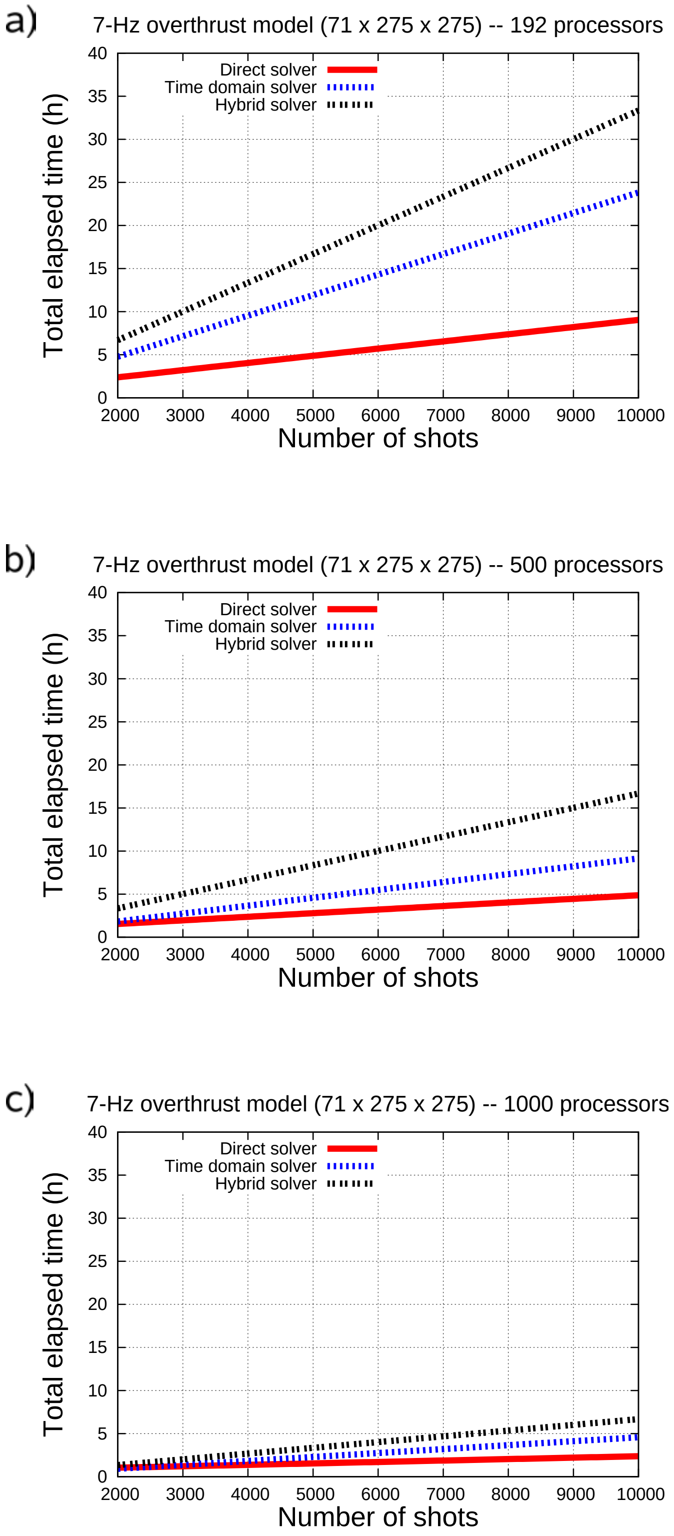

Figure 3.33: Comparison between multi-RHS resolution with direct, hybrid and time domain approaches on the 7-Hz overthrust model.

Conclusion

In this chapter, I have introduced the wave propagation modeling strategy adopted in the prospect of seismic imaging based on full waveform inversion. One of the main important features necessary in FWI is the multiple sources modeling efficiency. The modeling is based on either a massively parallel direct or hybrid direct-iterative solver. For each of them, theoretical aspects as well as complexity and scalability analysis have been introduced in order to support our choice.

The direct solver is very efficient with regards to multi-RHS resolution while the LU factorization can be a bottleneck due to its high order complexity. Moreover, the parallel direct solver suffers from a poor scalability. On contrary, the hybrid solver has got better scalability. The memory complexity is mitigated by the domain decomposition method since the LU are henceforth performed on smaller subdomains. However, the hybrid solver is less efficient with regards to multi-RHS resolution due to the iterative (GMRES) part necessary to perform for each source. In addition, the hybrid solver requires a tradeoff analysis between the memory storage and the computation time through the search of the optimal domain decomposition.

The iterative solver is a very promising approach in terms of memory and CPU time complexities, even if its robustness is still questioned. The time domain solver is a competitive approach thanks to its good scalability and robustness.

Part II

Inverse Problem

Chapter 4

Theory of local optimization

The aim of optimization is to find the variable/model that optimizes a given objective function. Without loss of generality, I will use “minimize” instead of “optimize”. In fact, maximize a function is equivalent to minimize its opposite (“minus” the function). There are typically two main approaches for optimization:

- Global optimization aim to reach the global minimum of an objective function. The algorithms are generally computationally demanding. There two main classes global optimization: deterministic approaches (branch and bound, etc.) explore the search space using for example a divide and conquer scheme and probabilistic approaches (simulated annealing, Monte-Carlo, genetic algorithms,) seek to optimize an appropriate probability function.

- Local optimization aims to find a local minimum in the vicinity of an initial “guess” model. Algorithms in this case are more affordable to deal with huge problems such as 3-D problems. If the initial guess is sufficiently close to the global minimum, then the local optimization will converge to the global minimum.

Local optimization algorithms are iterative. They start with an initial “apriori” variable/model and generate a sequence of improved estimates until convergence to the minimum. To update the model, most algorithms use values of the objective function and possibly its first and second derivatives. Typically, two fundamental strategies to move from an iterate to the next one are considered:

- “Line search” algorithms choose a direction and search along it for a new iterate that minimizes the objective function. The “distance to move” or “step length” can be a fixed value obtained with “apriori” information or computed through solving an other optimization problem.

- “Trust region” algorithms restrict the search of the minimizer in a region around some “apriori” model. If the trust region is too large, the search region is shrinked and the minimization is redone.

In some way, the two strategies differ in the order in which they choose the “direction” and the “step length” to move to the next iterate.

For Line search algorithms, the most obvious direction is the “steepest descent” direction given by minus the gradient (first-order derivative) of the objective function (Taylor’s theorem). The other well known search direction is the "Newton" direction. This search direction needs to compute the inverse of the Hessian (second-order derivative) of the objective function. This task may be computationally demanding. In this case, the Hessian is approximated and a class of search direction is defined and known as "Quasi-Newton" direction. "Conjugate gradient methods" use the knowledge of the previous iterate search direction and build a new direction such as they are "conjugate".

In this chapter, I will focus on the fundamentals of local optimization, specifically the "line search" methods. I will next expand the different ways to compute the "search direction" and "step length".

4.1 Line search methods

Let's consider a function f which depends on the variable m. The objective of local optimization is to find m^* which minimize f,

$$\min_{m} f(m). \tag{4.1}$$

As mentioned in the introduction, most of local optimization algorithms only consider the function and its first-order and second-order derivatives. It implies the strong "apriori" that the function f is very close to its quadratic approximation.

Consider the second-order Taylor series approximation of the function f around the point m_k,

$$\begin{aligned} f(m_k+p) &= f(m_k) + p^T \nabla f(m_k) + \frac{1}{2} p^T \nabla^2 f(m_k+tp)p \\ &= f(m_k) + p^T \nabla f(m_k) + \frac{1}{2} p^T \nabla^2 f(m_k)p + \mathcal{O}(||p||^3), \end{aligned} \tag{4.2}$$

where t belongs to $(0,1)$.

To find the minimizer, line search local optimization algorithms try to build an iterating chain that has this general form

$$m_{k+1} = m_k + \alpha_k p_k, \tag{4.3}$$

where α_k is called the "step length" and p_k the "descent direction" or "search direction" (Nocedal and Wright, 1999). This means that for any descent direction p and step length α,

$$\begin{aligned} f(m_k+\alpha p) &= f(m_k) + \alpha p^T \nabla f(m_k) + \frac{1}{2} \alpha^2 p^T \nabla^2 f(m_k+tp)p \\ &= f(m_k) + \alpha p^T \nabla f(m_k) + \frac{1}{2} \alpha^2 p^T \nabla^2 f(m_k)p + \mathcal{O}(||\alpha p||^3), \end{aligned} \tag{4.4}$$

where t belongs to $(0,\alpha)$.

4.1.1 First-order optimality condition

If m^* is a local minimizer of a continuously differentiable function f in an open neighborhood of m^*, then $\nabla f(m^*) = 0$ (Proof in (Nocedal and Wright, 1999)).

4.1.2 Second-order optimality condition

If m^* is a local minimizer of a function f and $\nabla^2 f$ exists and is continuous in an open neighborhood of m^*, then $\nabla f(m^*) = 0$ and $\nabla^2 f(m^*)$ is positive semidefinite (Proof in (Nocedal and Wright, 1999)).

4.2 Descent direction

4.2.1 Steepest descent method

The most natural search direction is the "steepest descent" defined as

$$p_k^{SD} = -\nabla f(m_k). \tag{4.5}$$

It is the direction along which the function decreases the most rapidly. The steepest descent direction involves only the first-order derivative or gradient of the function f.

To demonstrate this assumption, consider the first-order Taylor series approximation

$$f(m_k + \alpha p) \quad = \quad f(m_k) + \alpha p^T \nabla f(m_k) + \mathcal{O}(||\alpha p||^2). \tag{4.6}$$

The direction p that minimizes the function f satisfies

$$\min_p p^T \nabla f(m_k), \;\; \textit{subject to} \;\; ||p|| = 1. \tag{4.7}$$

The solution is intuitive,

$$p = -\frac{\nabla f(m_k)}{||\nabla f(m_k)||}. \tag{4.8}$$

More details can be found in Nocedal and Wright (1999). The convergence of the steepest descent method is linear.

The main advantage of the steepest descent method is that it only involves the gradient of the function contrary to Newton or quasi-Newton methods as it will be shown later. This does not mean that the computation of the gradient is easy. In fact, it depends on how we compute it. People have been used to explicitly compute and store the *Jacobian* or Fréchet derivatives matrix and use it when necessary. However, this matrix can need a huge amount of memory, thus need to be stored in the hard disk. This leads to some Input/Output (I/O) issues. An elegant way to compute the gradient without computing the *Jacobian* matrix is more and more used in the seismic community and is known as the adjoint-state method (Tromp et al., 2005; Plessix, 2006). This issue will be the object of a next section.

4.2.2 Newton method

To obtain the Newton direction, consider the Taylor series approximation (4.4)

$$f(m_k + \alpha p) \approx f(m_k) + \alpha p^T \nabla f(m_k) + \frac{1}{2}\alpha^2 p^T \nabla^2 f(m_k + tp)p := F_k(p), \tag{4.9}$$

where $F_k(p)$ is the quadratic approximation of $f(m_k + \alpha p)$.

The variable p that minimizes $F_k(p)$ is called the Newton direction and is expressed as

$$p_k^N = -(\nabla^2 f(m_k))^{-1} \nabla f(m_k). \tag{4.10}$$

The Newton direction is reliable when the function f and its quadratic approximation F are very close. The convergence of the Newton method is quadratic.

The main disadvantage of the Newton method is the need to evaluate the Hessian $\nabla^2 f(m_k)$. It is cumbersome, error-prone and prohibitive.

4.2.3 Quasi-Newton method

Quasi-Newton methods represent an alternative to Newton method in that way they do not require the computation of the Hessian. They only need an approximation of it. The quasi-Newton direction can write as

$$p_k^{QN} = -Hq_k^{-1} \nabla f(m_k). \tag{4.11}$$

where Hq is an approximation of the Hessian. In the following, Hq is called quasi-Hessian.

Hq is built such that it satisfies the equation known as the "secant equation" (Nocedal and Wright (1999) for details),

$$Hq_{k+1} s_k = y_k. \tag{4.12}$$

where $s_k = m_{k+1} - m_k$ and $y_k = \nabla f_{k+1} - \nabla f_k$. Note that the secant equation corresponds simply to a finite difference approximation of the second-order derivative, i.e the Hessian.

Two of the most popular formulae for updating Hq are the "symmetric-rank-one" (SR1) formula defined by

$$Hq_{k+1} = Hq_k + \frac{(y_k - Hq_k s_k)(y_k - Hq_k s_k)^T}{(y_k - Hq_k s_k)^T s_k}, \tag{4.13}$$

and the BFGS (Broyden-Fletcher-Goldfarb-Shanno) formula defined by

$$Hq_{k+1} = Hq_k - \frac{Hq_k s_k^T s_k Hq_k}{s_k^T Hq_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}. \tag{4.14}$$

Practically, implementations avoid the factorization of Hq at each iteration. Instead to update Hq, they update the inverse of Hq. In fact, it can be demonstrated that the equivalent BFGS formulae for the inverse of Hq is given by

$$Hq_{k+1}^{-1} = \left(I - \frac{s_k y_k^T}{y_k^T s_k}\right) Hq_k^{-1} \left(I - \frac{y_k s_k^T}{y_k^T s_k}\right) + \frac{s_k s_k^T}{y_k^T s_k}, \tag{4.15}$$

where I is the identity matrix.

The convergence of quasi-Newton methods is superlinear.

4.2.4 Conjugate gradient methods

The conjugate gradient direction has the general form

$$p_k = -\nabla f_k + \beta_k p_{k-1}, \tag{4.16}$$

where β_k is a scalar that ensures that p_k and p_{k-1} are conjugate.

Conjugate gradient methods were originally designed to solve a linear system of equations $Ax = b$, where A is symmetric and positive definite. This problem is equivalent to the problem of minimizing the quadratic function defined as $\frac{1}{2}x^T Ax - b^T x$. Conjugate gradient methods were next extended to nonlinear optimization problems.

Fletcher and Reeves (1964) have shown how to extend the linear conjugate gradient to nonlinear problems and presented the Fletcher-Reeves method. Many variants of nonlinear conjugate gradient methods were then proposed, the most efficient is the Polak-Ribière method (Polak and Ribière, 1969).

The nonlinear conjugate gradient methods are generally more efficient than the steepest descent method but do not attain Newton method efficacy.

4.3 Step length

The ideal choice of the step length is the minimizer of the function Φ defined as

$$\Phi(\alpha) = f(m_k + \alpha p_k). \tag{4.17}$$

We face a tradeoff in computing the step length. In fact, we would like to get α that minimizes Φ, but this can be too prohibitive. Practical methods perform an inexact line search to identify a step length that achieves an adequate reduction of f. Typically, these algorithms try out a sequence of candidates. A first step, bracketing phase, consists in determining an interval that contains an adequate step length. The second step, selection phase, consists in performing an interpolation or bisection phase within this interval in order to get the best step length.

4.3.1 Wolfe conditions

The Wolfe conditions (Nocedal and Wright, 1999) read

$$\begin{aligned} f(m_k + \alpha_k p_k) &\leq f(m_k) + c_1 \alpha_k \nabla f_k^T p_k \\ \nabla f(m_k + \alpha_k p_k)^T p_k &\geq c_2 \nabla f_k^T p_k, \end{aligned} \tag{4.18}$$

where $0 < c_1 < c_2 < 1$. The first inequality of system (4.18) is called *Armijo condition*. It stipulates that α_k give a sufficient decrease in the objective function. The second inequality is called the *curvature condition*.

A step length may satisfy the Wolfe conditions without being particularly close to the minimizer of the function Φ. The *strong* Wolfe conditions assume that

$$\begin{aligned} f(m_k + \alpha_k p_k) &\leq f(m_k) + c_1 \alpha_k \nabla f_k^T p_k \\ \left|\nabla f(m_k + \alpha_k p_k)^T p_k\right| &\geq c_2 \left|\nabla f_k^T p_k\right|. \end{aligned} \tag{4.19}$$

4.3.2 Goldstein conditions

The Goldstein conditions read

$$f(m_k) + (1-c)\alpha_k \nabla f_k^T p_k \leq f(m_k + \alpha_k p_k) \leq f(m_k) + c\alpha_k \nabla f_k^T p_k, \tag{4.20}$$

where $0 < c < \frac{1}{2}$. A disadvantage of the Goldstein conditions conpared to the Wolfe conditions is that the first inequality of system (4.20) may exclude all minimizers of the function Φ.

4.3.3 Examples of step length selection algorithms

4.3.3.1 Backtracking line search with interpolation

This algorithm takes account only of the Armijo condition. This latter can be written as

$$\Phi(\alpha_k) \leq \Phi(0) + c_1 \alpha_k \Phi'(0). \tag{4.21}$$

If the initial step length α_0 satisfies the condition (4.21), we finish the search. Otherwise, the interval $[0, \alpha_0]$ contains acceptable step lengths. Through a quadratic approximation of the function Φ that passes through the points $(0, \Phi(0))$, $(0, \Phi'(0))$ and $(\alpha_0, \Phi(\alpha_0))$ in the interval $[0, \alpha_0]$, we can define a new trial value as

$$\alpha_1 = \frac{\Phi'(0)\alpha_0^2}{2[\Phi(\alpha_0) - \Phi(0) - \Phi'(0)\alpha_0]}. \tag{4.22}$$

If α_1 satisfies (4.21), we terminate the search. Otherwise, a cubic approximation of Φ gives a new step length (Nocedal and Wright, 1999). We do the same thing with the two last step lengths until we satisfy the condition (4.21).

4.3.3.2 Line search algorithm for Wolfe conditions

The algorithm is detailed in the Algorithm (1) (Nocedal and Wright, 1999). The procedure is based on the knowledge that the interval (α_{i-1}, α_i) contains step lengths that satisfy strong Wolfe conditions. To perform the last step of the algorithm, interpolation procedure can be used. More details on the function **zoom** (Algorithm (2)) are in Nocedal and Wright (1999).

Algorithm 1 Line search algorithm for Wolfe conditions (Nocedal and Wright, 1999)

```
Set $\alpha \leftarrow 0$, choose $\alpha_{max}$ and $\alpha_1 \in (0, \alpha_{max})$
$i \leftarrow 1$
loop
    Evaluate $\Phi(\alpha_i)$
    if $\Phi(\alpha_i) > \Phi(0) + c_1\alpha_k\Phi'(0)$ OR $[\Phi(\alpha_i) \geq \Phi(\alpha_{i-1})$ AND $i > 1]$ then
        $\alpha^* \leftarrow$ zoom$(\alpha_{i-1}, \alpha_i)$ and stop
    end if
    Evaluate $\Phi'(\alpha_i)$
    if $|\Phi'(\alpha_i)| \leq -c_2\Phi'(0)$ then
        $\alpha^* \leftarrow \alpha_i$ and stop
    end if
    if $\Phi'(\alpha_i) \geq 0$ then
        $\alpha^* \leftarrow$ zoom$(\alpha_i, \alpha_{i-1})$ and stop
    end if
    Choose $\alpha_{i+1} \in (\alpha_i, \alpha_{max})$
    $i \leftarrow i + 1$
end loop
```

4.3.3.3 Discussion

Algorithms based on Wolfe conditions or strong Wolfe conditions behave generally better than backtracking algorithms. However, the curvature condition (second condition) is very restrictive when dealing with huge problems. Indeed, we need to evaluate the gradient of the function f at the point $m_k + \alpha_k p_k$ besides the point m_k necessary for the Armijo condition. This task can be prohibitive and the step length search is limited to backtracking techniques.

Conclusion

The main topic brought up during this chapter is local optimization methods. I have been focused on line search algorithms since our full waveform inversion approach is based on it. I was interested in introducing the main features of such algorithms. Typical line search algorithms are based on a quadratic approximation of the function to optimize (generally minimize) through a second-order Taylor expansion. This implies that the function to minimize is close to its approximation in the vicinity of the initial variable/model.

The minimization process is based on the construction of a perturbation chain starting from an initial model to reach a local minimum, due to local optimization process, which

Algorithm 2 zoom (Nocedal and Wright, 1999)

```
loop
    Interpolate (quadratic, cubic or bisection) to find a trial step length $\alpha_j \in (\alpha_{lo}, \alpha_{hi})$
    Evaluate $\Phi(\alpha_i)$
    if $\Phi(\alpha_i) > \Phi(0) + c_1\alpha_k\Phi'(0)$ OR $\Phi(\alpha_i) \geq \Phi(\alpha_{lo})$ then
        $\alpha_{hi} \leftarrow \alpha_i$
    else
        Evaluate $\Phi'(\alpha_i)$
        if $|\Phi'(\alpha_i)| \leq -c_2\Phi'(0)$ then
            $\alpha^* \leftarrow \alpha_j$ and stop
        end if
        if $\Phi'(\alpha_i)(\alpha_{hi} - \alpha_{lo}) \geq 0$ then
            $\alpha_{hi} \leftarrow \alpha_{lo}$
        end if
        $\alpha_{lo} \leftarrow \alpha_j$
    end if
end loop
```

can correspond to the global minimum if the initial model (well chosen) is in the vicinity of the global minimum. Such line search algorithms divide the process into two steps: the first step consists in finding an appropriate search direction such as the steepest descent, Newton, quasi-Newton, etc., and the second step in applying the optimal step length (amplitude).

Chapter 5

Full waveform inversion

In the eighties, Tarantola (1984a) and Lailly (1984) have formulated the seismic imaging problem as a least squares nonlinear inverse problem generally called waveform inversion or full waveform inversion (FWI). The aim of waveform inversion is to extract the complete information provided by the recorded data through a complete seismic modeling of the wave propagation including multiple reflections and diving waves in order to bypass imaging limitations of the standard imaging procedure in complex media such as salt body dipping flanks imaging. The optimal expected spatial resolution of FWI is half the wavelength of the wave provided that wide aperture acquisitions and wide frequency bandwidth sources are used. Moreover, FWI is a quantitative imaging method that provides, in addition to an image of the subsurface, a characterization of the medium physical properties.

The objective is to minimize the differences between predicted and observed seismograms. Unfortunately, the approach have shown to suffer from the presence of many local minima when applied to short or medium offsets reflection data (Mora, 1989). Fortunately, this property of least squares misfit function is less arduous at low frequencies than at high frequencies. Indeed, the width of the valley of the global minimum is proportional to the inverse of the frequency (Bunks et al., 1995). Therefore, a multi scale imaging strategy can be settled through successively inverting small to high frequencies.

In the nineties, Pratt (Pratt and Worthington, 1990; Pratt, 1990) have reformulated waveform inversion in the frequency domain. Pratt and collaborators apply waveform inversion to crosshole (Pratt, 1999) and wide aperture (large offsets) data (Pratt et al., 1996a; Brenders and Pratt, 2006, 2007) taking advantage of diving waves. In the frequency domain, the inversion procedure intrinsically defines a multi scale/resolution strategy where we first invert low frequency components and include progressively higher frequencies. This multi scale strategy allows mitigating the nonlinearity of the inverse problem. In addition, only few discrete frequencies are necessary to provide an acceptable image thanks to wavenumber redundancy (Sirgue and Pratt, 2004), thus favor frequency domain formulation to be efficient.

During the two last decades, many researchers have performed synthetic and real data case studies that have clearly shown the benefit of this approach (Pratt et al., 1996a; Ravaut et al., 2004; Dessa et al., 2004; Operto et al., 2006b,a; Jaiswal et al., 2008; Sears et al., 2008; Choi and Shin, 2008). Nevertheless, different investigations have shown that it is imperative to have a good starting pair (initial velocity model, lowest frequency ($\leq 3-5Hz$)) (Brenders and Pratt, 2006, 2007) to manage the nonlinearity of FWI. Data frequency content is closely related to the source frequency content. Many efforts have been carried out to improve low frequency data quality. Regarding the initial velocity model, investigations have been interested to evaluate the usefulness of models built with standard techniques such as first arrival traveltime (FATT) or reflection tomography (Ravaut et al., 2004; Dessa et al., 2004; Brenders and Pratt, 2006, 2007; Jaiswal et al., 2008). Results were mitigated according to large offsets and low frequencies availability and quality. Other techniques were proposed to remedy to this lack of efficacy such as stereotomography (Lambaré, 2008; Prieux et al., 2009), phase inversion (Shin and Min, 2006; Min and Shin, 2006) and Laplace domain inversion (Shin and Cha, 2008; Shin and Ha, 2008).

During the last years, the efforts have been concentrated on the 3-D FWI. The applications have been restricted to the acoustic approximation of the seismic wave propagation since FWI is based on complete seismic wave propagation modeling which is extremely computational demanding in 3-D. This partially explains why efforts have been particularly concentrated on the development of an efficient 3-D seismic wave modeling engine. Based on the efficacy of the use of direct solver in 2-D thanks to its multi-RHS resolution efficiency, I have proposed the same approach in 3-D (Ben-Hadj-Ali et al., 2008). Despite the limitations related to the LU decomposition CPU time and memory complexities, I have shown that the method still perform efficiently for small to medium problem sizes of few millions of unknowns, in other words for small frequencies ($\leq 7Hz$). Warner et al. (2007) and Erlangga and Herrmann (2008); Plessix (2007) have recommended to use an iterative method to solve the Helmholtz system. Plessix and Perkins (2009); Plessix (2009) have presented an application of anisotropic FWI on deep-water OBS data recorded in the Golf of Mexico. Sirgue et al. (2007b) use a time domain modeling and compute monochromatic responses through a discrete Fourier transform. Sirgue et al. (2009) have applied frequency domain FWI on Valhall (Munns, 1985; Kommedal et al., 2004) real data set and have shown the imaging resolution improvement with regards to reflection tomography velocity model which was used as an initial model for FWI.

In this chapter, I first expose the least squares formulation of FWI and Newton-like methods to solve this inverse problem, and focus on the frequency domain formulation. Then, I discuss the properties, advantages and drawbacks of waveform inversion.

5.1 Formulation

In this section, I aim to expound the mathematical formulation of full waveform inversion. The problem is set up as a local optimization problem that aims to minimize the misfit $\delta \mathbf{d}$ defined as the differences at the receivers between the observed/recorded data $\mathbf{d}_o$ and the computed/modeled data $\mathbf{d}_c = \mathbf{d}_c(m)$ where m is the model. The misfit vector $\delta \mathbf{d}$ reads then $\delta \mathbf{d} = \mathbf{d}_o - \mathbf{d}_c(m)$. The model m represents the physical parameters of the subsurface. It depends on considered physical approximation of the wave equation: acoustic/elastic, homogeneous/heterogeneous, viscosity, isotropic/anisotropic, etc.

Generally, the problem is set up as a least squares problem. In fact, the $\mathcal{L}2$-norm minimizes the probability of maximum likelihood (Tarantola, 1987).

The upcoming mathematical developments are independent of the domain in which we consider the data: time domain (Tarantola, 1984a,b; Lailly, 1984), frequency domain (Pratt and Worthington, 1990; Pratt, 1990), Laplace domain (Shin and Cha, 2008), etc.

5.1.1 Least squares criterion

The unconstrained/unweighted least squares misfit function $\mathcal{C}(\mathbf{m})$ reads

$$\mathcal{C}(\mathbf{m}) = \frac{1}{2}\mathcal{L}^2(\delta\mathbf{d}) = \frac{1}{2}\delta\mathbf{d}^T\delta\mathbf{d}^*, \tag{5.1}$$

where ^{T} is the transpose symbol and * defines the conjugate symbol. As mentioned in the chapter 8, local optimization methods search for a local optimum in the vicinity of a starting model $\mathbf{m}_0$. The first step is to linearize the misfit function around $\mathbf{m}_0$. This suggests that only small perturbations hold. In the framework of the Born approximation (Born and Wolf, 1993; Beydoun and Tarantola, 1988), $\mathbf{m}$ can be written as the sum of the starting model $\mathbf{m}_0$ and a perturbation model $\delta\mathbf{m}$ and reads $\mathbf{m} = \mathbf{m}_0 + \delta\mathbf{m}$.

The second step consists of expanding the second-order Taylor series of the misfit function $\mathcal{C}(\mathbf{m})$,

$$\begin{aligned} \mathcal{C}(\mathbf{m}_0 + \delta\mathbf{m}) &= \mathcal{C}(\mathbf{m}_0) + \sum_{j=1}^{M} \left.\frac{\partial\mathcal{C}(\mathbf{m})}{\partial m_j}\right|_{\mathbf{m}=\mathbf{m}_0} \delta m_j + \frac{1}{2}\sum_{j=1}^{M}\sum_{k=1}^{M} \left.\frac{\partial^2\mathcal{C}(\mathbf{m})}{\partial m_j \partial m_k}\right|_{\mathbf{m}=\mathbf{m}_0} \delta m_j \delta m_k \\ &+ \mathcal{O}(||\mathbf{m}||^3), \end{aligned} \tag{5.2}$$

where M denotes the dimension of the vector $\mathbf{m}$.

The misfit function reaches its minimum when its first-order derivative vanishes (first-order optimality condition, chapter 8). The first-order derivative with respect to a model parameter m_l reads

$$\frac{\partial\mathcal{C}(\mathbf{m})}{\partial m_l} = \left.\frac{\partial\mathcal{C}(\mathbf{m})}{\partial m_l}\right|_{\mathbf{m}=\mathbf{m}_0} + \sum_{j=1}^{M} \left.\frac{\partial^2\mathcal{C}(\mathbf{m})}{\partial m_j \partial m_l}\right|_{\mathbf{m}=\mathbf{m}_0} \delta m_j, \tag{5.3}$$

and implies the compact expression with respect to $\mathbf{m}$,

$$\frac{\partial\mathcal{C}(\mathbf{m})}{\partial \mathbf{m}} = \left.\frac{\partial\mathcal{C}(\mathbf{m})}{\partial \mathbf{m}}\right|_{\mathbf{m}=\mathbf{m}_0} + \left.\frac{\partial^2\mathcal{C}(\mathbf{m})}{\partial \mathbf{m}^2}\right|_{\mathbf{m}=\mathbf{m}_0} \delta\mathbf{m}. \tag{5.4}$$

$\frac{\partial\mathcal{C}(\mathbf{m}_0)}{\partial\mathbf{m}}$ is called the gradient and $\frac{\partial^2\mathcal{C}(\mathbf{m}_0)}{\partial\mathbf{m}^2}$ the Hessian. Setting $\frac{\partial\mathcal{C}(\mathbf{m})}{\partial\mathbf{m}}$ to zero gives the expression of the descent direction,

$$\delta\mathbf{m} = -\left[\left.\frac{\partial^2\mathcal{C}(\mathbf{m})}{\partial \mathbf{m}^2}\right|_{\mathbf{m}=\mathbf{m}_0}\right]^{-1} \left.\frac{\partial\mathcal{C}(\mathbf{m})}{\partial \mathbf{m}}\right|_{\mathbf{m}=\mathbf{m}_0}. \tag{5.5}$$

This descent direction equation corresponds to Newton direction (Chapter 8).

5.1.2 Newton, Gauss-Newton & steepest descent directions

The objective of this section is to expand the expressions of the gradient and the Hessian and retrieve the descent directions presented in the chapter 8. I only consider the frequency domain formulation of the waveform inversion. In this case, data are complex valued. For more details, see Pratt et al. (1998); Ben-Hadj-Ali et al. (2008).

5.1.2.1 Gradient expression

The first-order derivative of the misfit function with respect to m_l reads

$$\begin{aligned} \frac{\partial \mathcal{C}(\mathbf{m})}{\partial m_l} &= -\frac{1}{2}\sum_{i=1}^{N}\left[\frac{\partial d_{c_i}}{\partial m_l}(d_{o_i}-d_{c_i})^* + (d_{o_i}-d_{c_i})\frac{\partial d_{c_i}^*}{\partial m_l}\right] \\ &= -\sum_{i=1}^{N}\Re\left[\left(\frac{\partial d_{c_i}}{\partial m_l}\right)^*(d_{o_i}-d_{c_i})\right], \end{aligned} \tag{5.6}$$

where N represents the number of terms in the data vector and is directly related to the acquisition, i.e. shots and receivers number. $\Re$ defines the real part of a complex value. In a more compact form, it reads

$$\nabla\mathcal{C}_{\mathbf{m}} = \frac{\partial \mathcal{C}(\mathbf{m})}{\partial \mathbf{m}} = -\Re\left[\left(\frac{\partial \mathbf{d}_c(\mathbf{m})}{\partial \mathbf{m}}\right)^T (\mathbf{d}_o - \mathbf{d}_c(\mathbf{m}))^*\right]. \tag{5.7}$$

For the model $\mathbf{m}_0$, the gradient reads

$$\nabla\mathcal{C}_{\mathbf{m}_0} = \left.\frac{\partial \mathcal{C}(\mathbf{m})}{\partial \mathbf{m}}\right|_{\mathbf{m}=\mathbf{m}_0} = -\Re\left[\left.\left(\frac{\partial \mathbf{d}_c(\mathbf{m})}{\partial \mathbf{m}}\right)^T\right|_{\mathbf{m}=\mathbf{m}_0} (\mathbf{d}_o - \mathbf{d}_c(\mathbf{m}_0))^*\right] = -\Re\left[\mathbf{J}_0^T\delta\mathbf{d}_0^*\right], \tag{5.8}$$

where $\mathbf{J}_0$ is the Fréchet derivatives matrix.

5.1.2.2 Hessian expression

Following the same approach as in for the gradient, the Hessian reads

$$\left.\frac{\partial^2 \mathcal{C}(\mathbf{m})}{\partial \mathbf{m}^2}\right|_{\mathbf{m}=\mathbf{m}_0} = \Re\left[\mathbf{J}_0^T\mathbf{J}_0^*\right] + \Re\left[\left(\frac{\partial \mathbf{J}_0}{\partial \mathbf{m}}\right)^T (\delta\mathbf{d}_0^* \cdots \delta\mathbf{d}_0^*)\right]. \tag{5.9}$$

The first term $\Re\left[\mathbf{J}_0^T\mathbf{J}_0^*\right]$ is called the approximate Hessian. It is the zero-lag correlation between the partial derivative of wavefields with respect to different parameters. Therefore, it represents the spatial correlation between the images of different point scatterers. It can be viewed as a resolution operator resulting from limited bandwidth of the source and the acquisition geometry. Indeed, applying the inverse of the Hessian is equivalent to applying a spiking deconvolution of the gradient of the misfit function.

The term $\Re\left[\mathbf{J}_0^T\mathbf{J}_0^*\right]$ is diagonal dominant since the diagonal terms are defined by zero-lag auto-correlation. This diagonal term reduces the effects of the geometrical spreading. Therefore, in the frame of surface acquisition, it helps to scale the deep perturbations (large offsets / small amplitudes) with respect to the shallow perturbations (near offsets / high amplitudes).

The second term $\Re\left[\left(\frac{\partial \mathbf{J}_0}{\partial \mathbf{m}}\right)^T(\delta\mathbf{d}_0^*\cdots\delta\mathbf{d}_0^*)\right]$ is the zero-lag correlation between the second-order partial derivative of the wavefields with data residuals. Since first-order partial derivative is related to single scattering, it can be expected that second-order partial derivative is related to double or multiple scattering.

5.1.2.3 Descent direction

The descent direction equation (5.5) (Newton direction) reads

$$\delta\mathbf{m} = \left\{\Re\left[\mathbf{J}_0^T\mathbf{J}_0^* + \left(\frac{\partial \mathbf{J}_0}{\partial \mathbf{m}}\right)^T(\delta\mathbf{d}_0^*\cdots\delta\mathbf{d}_0^*)\right]\right\}^{-1}\Re\left[\mathbf{J}_0^T\delta\mathbf{d}_0^*\right]. \tag{5.10}$$

Newton direction has locally a quadratic convergence. Generally, the second term of the Hessian ($\Re\left[\left(\frac{\partial \mathbf{J}_0}{\partial \mathbf{m}}\right)^T(\delta\mathbf{d}_0^*\cdots\delta\mathbf{d}_0^*)\right]$) is neglected since in the framework of the Born approximation multiple scattering are neglected (Pratt et al., 1998). This leads to a quasi-Newton direction called Gauss-Newton and expressed as below,

$$\delta\mathbf{m} = \left\{\Re\left[\mathbf{J}_0^T\mathbf{J}_0^*\right]\right\}^{-1}\Re\left[\mathbf{J}_0^T\delta\mathbf{d}_0^*\right]. \tag{5.11}$$

If the Hessian is replaced by a scalar α, the expression gives the steepest descent direction,

$$\delta\mathbf{m} = \alpha\Re\left[\mathbf{J}_0^T\delta\mathbf{d}_0^*\right]. \tag{5.12}$$

5.2 Numerical issues

5.2.1 Gradient computation through adjoint method

Consider the forward problem $\mathbf{Ap} = \mathbf{s}$ and one source s and one frequency f. The differentiation of this equation with regards to the model parameter m_i leads to

$$\mathbf{A}\frac{\partial \mathbf{p}}{\partial m_i} = -\frac{\partial \mathbf{A}}{\partial m_i}\mathbf{p}. \tag{5.13}$$

$\frac{\partial \mathbf{p}}{\partial m_i}$ is the partial derivative of wavefields. The extraction of the values of the partial derivative of wavefields at the receiver positions allows building the Fréchet derivative matrix $\mathbf{J}$. Indeed, $\mathbf{J} = \left(\frac{\partial(\mathcal{R}\mathbf{p})}{\partial \mathbf{m}}\right)$ where $\mathcal{R}$ is the restriction operator which extracts the solution at the receivers. If we insert the expression of the $\mathbf{J}$ and the relation (5.13) in the gradient expression (5.7), we get

$$\begin{aligned} \nabla\mathcal{C}_{\mathbf{m}} &= -\Re\left[\left(\frac{\partial \mathbf{d}_c(\mathbf{m})}{\partial \mathbf{m}}\right)^T \delta\mathbf{d}^*\right] \\ &= -\Re\left[\left(\frac{\partial (\mathcal{R}\mathbf{p})}{\partial \mathbf{m}}\right)^T \delta\mathbf{d}^*\right] \\ &= \Re\left[\mathbf{p}^T\left(\frac{\partial \mathbf{A}}{\partial \mathbf{m}}\right)^T (\mathbf{A}^{-1})^T(\mathcal{R}^T\delta\mathbf{d}^*)\right]. \end{aligned} \tag{5.14}$$

$\mathcal{R}^T$ is the prolongation operator adjoint to $\mathcal{R}$.

Each column of $\mathbf{A}^{-1}$ represents the Green function for an impulse source located at each node of the model space. In virtue of the reciprocity in space of the Green function, the matrix $\mathbf{A}^{-1}$ is symmetric and $(\mathbf{A}^{-1})^T = \mathbf{A}^{-1}$. Hence, (5.14) reads

$$\begin{aligned} \nabla\mathcal{C}_{\mathbf{m}} &= \Re\left[\mathbf{p}^T\left(\frac{\partial \mathbf{A}}{\partial \mathbf{m}}\right)^T \mathbf{A}^{-1}(\mathcal{R}^T\delta\mathbf{d}^*)\right] \\ &= \Re\left[\mathbf{p}^T\left(\frac{\partial \mathbf{A}}{\partial \mathbf{m}}\right)^T \mathbf{R_b}\right]. \end{aligned} \tag{5.15}$$

$\mathbf{R_b}$ is called the backpropagated residuals wavefield. The conjugate symbol * indicates the backpropagation. Note that the residuals associated with one source are assembled to form one vector. This means that for each source, the computation of the gradient reduces to the simulation of two forward problems, the first one to compute the wave equation solution and extract the data at the receivers, the second one to backpropagate the residuals at once.

This strategy to compute the gradient is known as the adjoint method (Plessix, 2006). It derives from the Lagrange formulation of constrained optimization (Nocedal and Wright, 1999; Akcelik, 2002).

This recalls the imaging principle of wavefield continuation migration methods (Claerbout, 1985; Biondi, 2006), especially reverse time migration.

For multiple sources and frequencies, the expression of the gradient reads

$$\nabla\mathcal{C}_{\mathbf{m}} = \sum_s\sum_f \Re\left[\mathbf{p}^T\left(\frac{\partial \mathbf{A}}{\partial \mathbf{m}}\right)^T \mathbf{R_b}\right]. \tag{5.16}$$

5.2.2 Hessian approximations

Due to the cost of the Hessian computation and storage, only manageable approximations are used. We also need to take into account that we need its inverse. Thus, two classes of approximations are possible: diagonal approximations of the Hessian, hence simple to invert or manageable approximations of the inverse of Hessian.

5.2.2.1 Diagonal of the approximate Hessian

Consider the approximate Hessian $\mathcal{H}_a$ (Pratt et al., 1998) defined by

$$\mathcal{H}_a = \Re\left[\mathbf{J}^T\mathbf{J}^*\right]. \tag{5.17}$$

One element of the sensitivity matrix reads

$$\mathbf{J}_{k(s,r),l} = \mathbf{u}_s^T \left[\frac{\partial \mathbf{A}^T}{\partial m_l}\right] \mathbf{A}^{-1}\delta_r, \tag{5.18}$$

where $k(s,r)$ denotes a (source,receiver) couple, with s and r are respectively shot and receiver positions. δ_r is an impulse source located at the receiver position r. To build the sensitivity matrix, we need to simulate one forward problem for each source and each receiver. This means that the computation cost of the approximate Hessian depends closely on the acquisition geometry.

The storage of the approximate Hessian $\mathcal{H}_a$ is generally cumbersome for huge acquisitions, such as in three dimensions (3-D). A way to circumvent the problem is to consider a coarser acquisition in which we build the approximate Hessian (Operto et al., 2006b,a) and use only its diagonal part. This provides a good scaling of the gradient (Pratt et al., 1998; Ravaut et al., 2004).

5.2.2.2 Diagonal of the pseudo Hessian

Shin et al. (2001) have proposed a computationally efficient approximation of the approximate Hessian. Indeed, they have proposed to use only the virtual sources $f_i = -\frac{\partial \mathbf{A}}{\partial m_i}\mathbf{p}$ (Equation (5.13)) instead of $\mathbf{A}^{-1}f_i$. The pseudo Hessian $\mathcal{H}_p$ reads then

$$\mathcal{H}_p = \Re\left[\mathbf{F}^T\mathbf{F}^*\right], \tag{5.19}$$

where $\mathbf{F}_{(i,j)} = (f_i)_j$ is the virtual sources matrix (Shin et al., 2001). This expression reduces the cost of the approximation of the Hessian to the cost of the gradient. Indeed, computing the pseudo Hessian implies to simulate only forward problems for the sources instead of the sources and receivers in the case of the approximate Hessian. The evaluation of the pseudo Hessian can be done simultaneously with the gradient of the misfit function. This reduces dramatically the computation and storage effort. As for the approximate Hessian, only the diagonal part of the pseudo Hessian is used in practice.

5.2.2.3 Limited memory BFGS (L-BFGS) approximation of the Hessian

Sometimes, it is necessary to consider non diagonal terms of the Hessian to get an acceptable convergence. A smart way to accomplish this objective is the BFGS approach (Section 4.2.3). Nocedal (1980) proposed and efficient variant of BFGS method, called L-BFGS, that limits the memory storage requirements.

Let us recall the BFGS formula for the inverse of the quasi-Hessian Hq,

$$Hq_{k+1}^{-1} = \left(I - \frac{s_k y_k^T}{y_k^T s_k} \right) Hq_k^{-1} \left(I - \frac{y_k s_k^T}{y_k^T s_k} \right) + \frac{s_k s_k^T}{y_k^T s_k}, \tag{5.20}$$

where $s_k = \mathbf{m}_{k+1} - \mathbf{m}_k$ and $y_k = \nabla\mathcal{C}_{k+1} - \nabla\mathcal{C}_k$. This relation is recursive. The inverse of the quasi-Hessian at a given iteration depend only of the models and the gradients of the function at all the previous iterations.

In the L-BFGS algorithm, the models and the gradients of the function are stored only for a limited number of iterations. In addition, the inverse of the quasi-Hessian is not explicitly stored. The product $Hq_k^{-1}\nabla\mathcal{C}(m_k)$ is implicitly evaluated through inner products and vector summations involving $\nabla\mathcal{C}(m_k)$ and the limited stored sequence (s_i, y_i) (Nocedal and Wright, 1999, page 178).

5.2.3 Source estimation

In practice, the source wavelet is unknown. To update the source, we minimize our misfit function (5.1) with regards to the source unknown. The relationship between the seismic wave field and the source is linear, hence,

$$\mathbf{p} = \mathbf{G}\mathbf{s}, \tag{5.21}$$

where $\mathbf{G}$ is the Green function and $\mathbf{s}$ is the source wavelet. The misfit function (5.1) reads

$$\begin{aligned} \mathcal{C}(\mathbf{m}) &= \frac{1}{2}(\mathbf{d}_o - \mathbf{d}_c)^T(\mathbf{d}_o - \mathbf{d}_c)^* \\ &= \frac{1}{2}(\mathbf{d}_o - \mathbf{s}\mathbf{G}_{\mathcal{R}})^T(\mathbf{d}_o - \mathbf{s}\mathbf{G}_{\mathcal{R}})^*, \end{aligned} \tag{5.22}$$

where $\mathbf{G}_{\mathcal{R}}$ is the restriction of Green function $\mathbf{G}$ at the receivers positions. The differentiation of (5.22) with respect to $\mathbf{s}$ leads to the update expression of the source,

$$\mathbf{s} = \frac{\mathbf{d}_o^T \mathbf{G}_{\mathcal{R}}^*}{\mathbf{G}_{\mathcal{R}}^T \mathbf{G}_{\mathcal{R}}^*}. \tag{5.23}$$

The estimation of the source as the estimation of the model perturbation is performed iteratively. At each iteration, the source as well as the model perturbation is updated.

5.3 Full waveform inversion: issues

5.3.1 Preconditioning & regularization

Nonlinear inverse problems are known for their severe ill-posedness. In order to mitigate ill-posedness, some preconditioning and regularization are generally applied. The misfit function (5.1) becomes

$$\begin{aligned}\mathcal{C}(\mathbf{m}) &= \frac{1}{2}\left[\delta\mathbf{d}^T\mathbf{C_d}\delta\mathbf{d}^* + (\mathbf{m}-\mathbf{m}_{apriori})^T\mathbf{C_m}(\mathbf{m}-\mathbf{m}_{apriori})^*\right] \\ &= \frac{1}{2}\left[\delta\mathbf{d}^T\mathbf{W_d}^T\mathbf{W_d}\delta\mathbf{d}^* + (\mathbf{m}-\mathbf{m}_{apriori})^T\mathbf{W_m}^T\mathbf{W_m}(\mathbf{m}-\mathbf{m}_{apriori})^*\right],\end{aligned} \tag{5.24}$$

where $\mathbf{C_d} = \mathbf{W_d}^T\mathbf{W_d}$ is a preconditioning operator and $\mathbf{C_m} = \mathbf{W_m}^T\mathbf{W_m}$ a regularization operator. A widespread choice of $\mathbf{C_d}$ and $\mathbf{C_m}$ is the inverse of model and data covariance operators respectively (Bayes *apriori* (Tarantola, 1987; Scales and Smith, 1994)).

$\mathbf{W_d}$ corresponds to a weighting/filtering operator that applies on the data. $\mathbf{W_d}$ can permit to weight data with respect to offset in order to strengthen the far offset data in the case of crustal scale imaging (Operto et al., 2006b) or in order to define a multi scale imaging strategy based in a progressive introduction of near to far offsets data.

$\mathbf{W_m}$ can be a roughness operator generally obtained by taking a first or second-order finite differences operator. In this case, it aims to impose a smoothness constraint on the model. Such regularization operator is the Tikhonov regularization (Tikhonov and Arsenin, 1977). An alternative choice is total variation regularization (Gholami and Siahkoohi, 2009). It belongs to the edge preserving class. As inferred by the class name, it aims to preserve both the smooth and blocky parts of the model.

The same operations detailed for the previous misfit function (5.1) apply for (5.24) and lead to the model perturbation

$$\begin{aligned}\delta\mathbf{m} &= \left\{\Re\left[\mathbf{J}_0^T\mathbf{C_d}\mathbf{J}_0^*\right] + \mathbf{C_m}\right\}^{-1}\Re\left[\mathbf{J}_0^T\mathbf{C_d}\delta\mathbf{d}_0^*\right] \\ &= \mathbf{C_m}^{-1}\left\{\Re\left[\mathbf{J}_0^T\mathbf{C_m}^{-1}\mathbf{J}_0^*\right] + \mathbf{C_d}^{-1}\right\}^{-1}\Re\left[\mathbf{J}_0^T\delta\mathbf{d}_0^*\right].\end{aligned} \tag{5.25}$$

Practically, the gradient is smoothed with a Gaussian filter (Ravaut et al., 2004; Ben-Hadj-Ali et al., 2008) such that

$$\delta m_i = \alpha\left(diag\mathbf{H}_a + \epsilon I\right)^{-1}\mathcal{G}_\mathbf{m}\Re\left\{\mathbf{p}^T\left[\frac{\partial\mathbf{A}^T}{\partial m_i}\right]\mathbf{A}^{-1}\mathbf{C}_d\delta\mathbf{d}^*\right\}, \tag{5.26}$$

where $diag\ \mathbf{H_a} = diag\Re\{\ \mathbf{J}_0^T\ \mathbf{C}_d\ \mathbf{J}_0^*\}$ denotes the diagonal elements of the weighted approximate Hessian $\mathbf{H_a}$, $\mathbf{J}_0$ denotes the sensitivity matrix and $\mathcal{G}_\mathbf{m}$ is a Gaussian smoothing operator. ϵ denotes a prewhitening factor that prevents singularities.

5.3.2 Resolution power & multi-scale strategy

In this section, I am interested in studying the spatial resolution of full waveform inversion. I will show that the optimal resolution is half the wavelength.

Let us consider a homogeneous background model of velocity c, an incident monochromatic plane wave propagating in the source-scatterer direction, and a scattered monochromatic plane wave in the far-field approximation propagating in the scatterer-receiver direction $\mathbf{r}$ (Figure 5.1). The incident and scattered Green's functions can be approximated

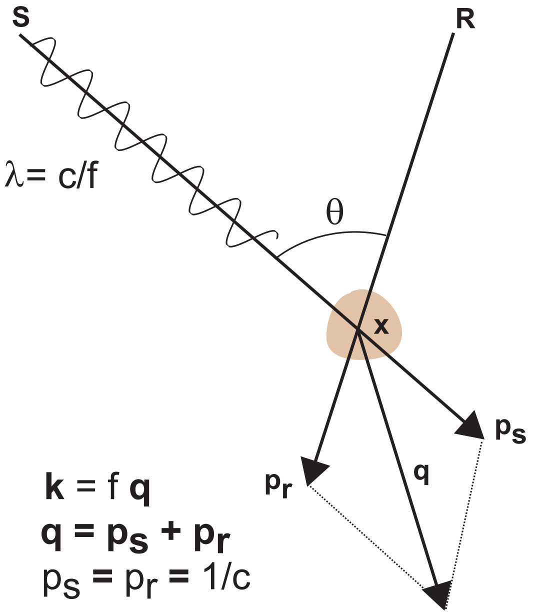

Figure 5.1: 1-D scattering experiment scheme.

as

$$\begin{aligned} G_0(\mathbf{x},\mathbf{s}) &\approx \exp(\iota k \mathbf{s}.\mathbf{x}) \\ G_0(\mathbf{x},\mathbf{r}) &\approx \exp(\iota k \mathbf{r}.\mathbf{x}), \end{aligned} \tag{5.27}$$

where $k_0 = \omega/c$, $\mathbf{s}$, $\mathbf{r}$ and $\mathbf{x}$ are respectively the source, receiver and scatterer position vectors. The gradient of the misfit function reads

$$\begin{aligned} \nabla\mathcal{C}(\mathbf{m}) &\approx -\sum_s \sum_r \Re\left\{\exp(\iota k \mathbf{s}.\mathbf{x})\exp(\iota k \mathbf{r}.\mathbf{x})\delta\mathbf{d}^*\right\} \\ &\approx -\sum_s \sum_r \Re\left\{\exp(\iota k(\mathbf{s}+\mathbf{r}).\mathbf{x})\delta\mathbf{d}^*\right\}. \end{aligned} \tag{5.28}$$

The gradient has the form of a truncated Fourier series where the integration variable is the scattering wavenumber vector $\mathbf{k} = k(\mathbf{s}+\mathbf{r})$ and the coefficients of the series are the data residuals. At a given frequency, the acquisition geometry controls the sampling and the bandwidth of the Fourier series through the summation over the sources and receivers (Figure 5.1). To see more clearly the relationship between spatial resolution and the acquisition geometry, consider the expression of the wavenumber $\mathbf{k}$ written as

$$\mathbf{k} = \frac{2\omega}{c} cos\,(\theta/2)\,\mathbf{n}, \tag{5.29}$$

where $\mathbf{n}$ is a unit vector in the direction of the slowness vector $(\mathbf{s}+\mathbf{r})$. This highlights the relationship between spatial resolution power of FWI and the frequency-offset pair. One frequency and one aperture in the data space map one wavenumber in the model space.

Therefore, they have a redundant control of the wavenumber coverage. This redundancy increases with the aperture bandwidth. Pratt and Worthington (1990), Sirgue and Pratt (2004) and Brenders and Pratt (2007) have proposed to decimate the wavenumber coverage redundancy in the frequency domain FWI by limiting the inversion to a few discrete frequencies. This data reduction leads to a computationally efficient frequency domain FWI and allows the managing of a compact volume of data, two main advantages with comparison to time domain FWI formulation. The rule for the selection of the frequencies to be used in the FWI is that the maximum wavenumber imaged by a frequency matches the minimum vertical wavenumber imaged by the next frequency. A guideline of frequencies selection for 1-D velocity model problems is detailed in Figure 5.2 from Sirgue and Pratt (2004). In a homogeneous medium and for a plane reflector, the guideline of frequencies selection reads

$$\Delta f_{n+1} = f_{n+1} - f_n = (1 - \alpha_{min}) f_{n+1}, \tag{5.30}$$

where α_{min} is a constant that depends on the offset and the reflector depth. According to this guideline, the frequency interval increases with the frequency.

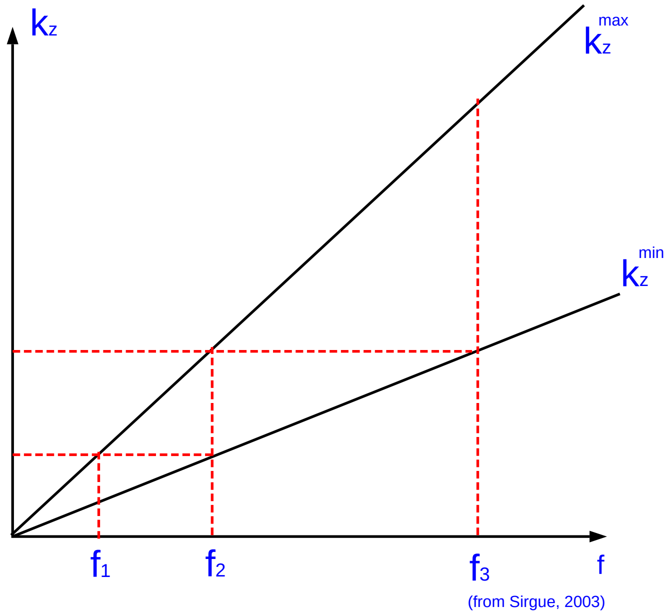

Figure 5.2: Illustration of the frequency discretization strategy. (From Sirgue, 2003)

The low frequencies of the data and the wide apertures contribute to resolve the intermediate and large wavelengths of the medium. At the other side of the wavenumber spectrum, normal incidence wavenumbers lead to a maximum resolution of half a wavelength ($\frac{\lambda}{2}$) if data provide. The ability of the wide apertures to resolve the large wavelengths of the medium has encouraged many studies to consider long-offset acquisition geometries to design better posed FWI problems (Pratt et al., 1996b; Ravaut et al., 2004). Indeed, the equation (5.29) implies that the large wavelengths can be resolved provided that wide-aperture data are recorded by the acquisition geometry.

Conclusion

Full waveform inversion is formulated as a least squares optimization problem where the misfit function between the recorded and predicted data is minimized. The inverse problem is linearized through the Born approximation while the seismic wave modeling is nonlinear. The linearization leads to the Newton, quasi-Newton or steepest descent algorithms to solve this optimization problem, where the gradient and/or the Hessian of the misfit function have to be evaluated. The gradient computation is based on the adjoint-state method where two forward problems are required per source: one to compute the shot wavefield and one to backpropagate the residuals wavefield. The Hessian is approximated since its evaluation and storage are computationally demanding.

Since the problem is highly nonlinear and ill-posed, preconditioning and regularization are performed in order to mitigate nonlinearity and ill-posedness. Preconditioning applies in the data domain while regularization applies in the model domain.

According to the resolution study, the optimal expected spatial resolution is half the wavelength provided that large offset acquisition geometries and wide frequency bandwidth sources are used.

The frequency domain presents many advantages since the frequency sampling is undersampled thanks to wavenumber redundancy which leads to the inversion of a few discrete frequencies and defines a multi resolution/scale imaging strategy through the inversion of small to high frequencies.

Chapter 6

Parallel implementation [1]

In this chapter, I will exclusively be interested in the practical aspects of the implementation of the 3-D frequency domain FWI software. The FWI software interfaces both the direct and hybrid solver approaches, introduced in the chapter 8, to simulate the seismic wave propagation solutions. Therefore, the domain decomposition method, which is used in both modeling approaches, drives the FWI implementation. The practical aspects are mainly related to parallelism and data management. This latter is crucial in 3-D, even if the data are decimated in the frequency domain, since the data storage requirements demand important facilities. In fact, the storage requirements for a dense 3-D acquisition may be critical.

The FWI software is written in Fortran90 and the parallelism is based on *Message Passing Interface* (MPI) communication standard.

First, I will briefly introduce the frequency domain FWI algorithm. Then, I will be interested in the specificities of the implementation of the 3-D FWI software.

---

[1] The reader may skip this chapter if he is not interested in the practical implementation issues.

6.1 Frequency domain FWI algorithm

In this section, I introduce the frequency domain FWI algorithm. The algorithm has been detailed in Sourbier et al. (2009). The multi scale/resolution inversion strategy implies to successively process increasing frequencies or groups of frequencies. Thus, the main loop of the FWI algorithm is the frequency loop.

A detailed frequency domain FWI algorithm is shown in Figure 6.1. Three main parts can be distinguished: the gradient, Hessian approximation and step length. For each one of these parts, multi-source forward problems are simulated. The forward problem simulation can be performed by both the direct and hybrid solver approaches. The gradient and Hessian computation is straightforward. The optimal step length is estimated through parabolic fit.

The parabolic fit approach tries to find the optimal step length and involves evaluating many times the misfit function. A first value α_0 is already evaluated from the gradient calculation. The two other values α_1 and α_2 satisfy that

$$\alpha_0 \leq \alpha_1 \leq \alpha_2,$$
$$\mathcal{C}(\alpha_1) \leq \mathcal{C}(\alpha_0) \; and \; \mathcal{C}(\alpha_1) \leq \mathcal{C}(\alpha_2).$$

Each additional misfit function evaluation needs a multi-source forward problem simulation. Once three valid values are evaluated, the minimum of the parabolic function describing these points is used as the line search step length.

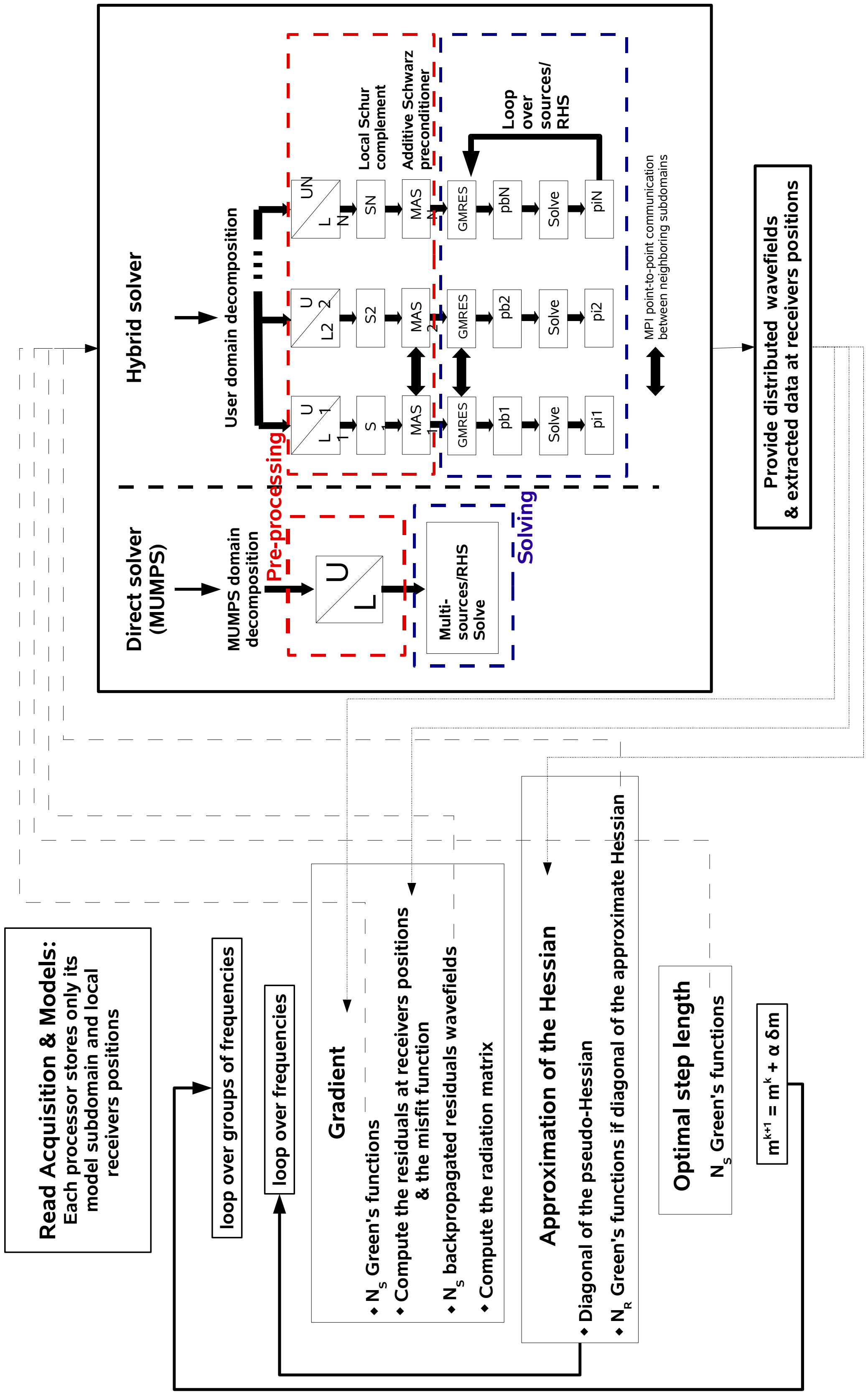

Figure 6.1: Frequency domain FWI flowchart.

6.2 Practical FWI implementation aspects

In this part, the practical aspects of the FWI software implementation are discussed in order to focus on the specificities of the 3-D implementation. As mentioned in the previous section, the forward problem is based on both the direct and hybrid solvers. The parallel implementation of the direct and hybrid solver modeling is based on the domain decomposition method. Therefore, the FWI algorithm is driven by the domain decomposition approach.

6.2.1 Domain decomposition management

For the hybrid solver approach, the domain decomposition is fixed by the user. However, with the direct solver MUMPS, the domain decomposition is driven by the LU factorization and matrix scaling and pivoting issues. Therefore, the domain distribution may change at each iteration since the matrix changes. This issue may prevent to efficiently manage data. To overcome this problem, a second level of domain decomposition is set up by the user. A mapping operation is then performed to redistribute the solutions in accordance to this domain decomposition instead of MUMPS decomposition which is no longer used (Figure 6.2).

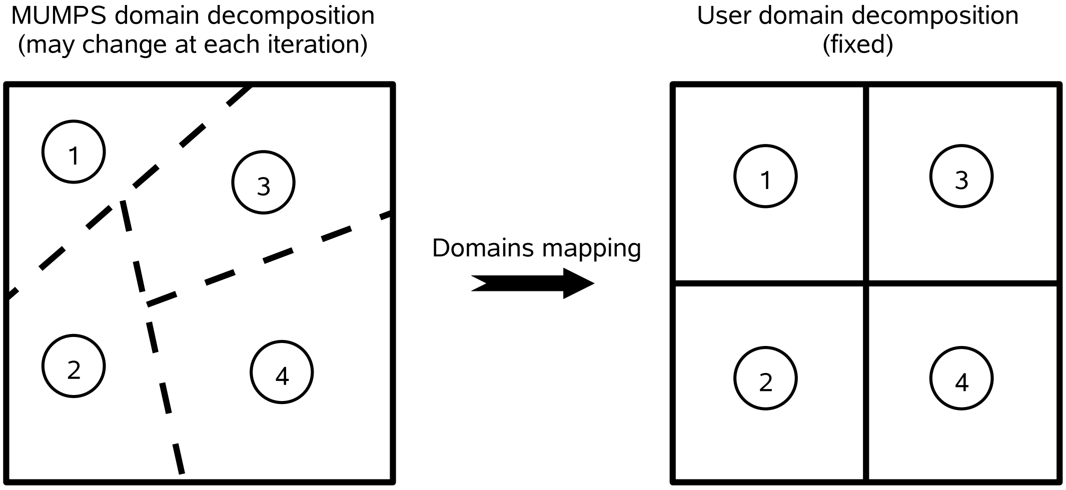

Figure 6.2: Domains mapping operation between MUMPS and user-defined domain decompositions.

6.2.2 Acquisition management

The information related to the acquisition have been used to be stored in 2-D by the host process (Sourbier et al., 2009). In 3-D, this strategy may be not applicable for dense acquisition systems. In fact, for a 3-D acquisition with 4000 shots and 50000 receivers per shot, some arrays related to the acquisition such as the correspondence between shots and their receivers represent about 700 *Mbytes*. In common parallel platforms, only 2 *Gbytes* per core of memory are available. In order to overcome this issue, these arrays are stored in a distributed form. Each process stores only information related to each subdomain. It means that the acquisition data is distributed between the subdomains that contain a part of the acquisition.

6.2.3 Data management

All the forward problem solutions (FPS) remain distributed in-core in the algorithm. No disk swapping is used in the FWI algorithm. If not enough memory is available to store in-core all the FPSs, the computation of the multi-RHS resolution is performed in a sequential loop over partitions of RHS terms. Each partition loads in-core the maximum number of solutions fitting the available memory. At each partition, the gradient, misfit function and Hessian are updated.

The data extracted at receivers positions are stored in a distributed form contrary to 2-D implementation where they are stored by the host process (Sourbier et al., 2009). This prevents a memory bottleneck for dense 3-D seismic acquisition. In addition, the fixed domain decomposition prevents to search the receiver positions in the local distributed wavefields when extracting the solutions at receivers. In fact, the search is done only once and a global-local indices correspondence is performed once for all.

Conclusion

A 3-D parallel visco-acoustic frequency domain FWI software has been developed. Both the direct and hybrid approaches have been implemented as seismic wave modeling engines since both are based on the domain decomposition strategy. In 3-D, data and acquisition information have to be appropriately managed in parallel due to related high memory storage requirements.

Chapter 7

Numerical Examples

In the previous chapters of the Inverse Problem part, I have introduced the theoretical and practical implementation aspects of full waveform inversion. In this chapter, I am interested in validating the FWI software with several synthetic applications on homogeneous and complex models and of different problem sizes. The seismic wave modeling for these FWI applications is based either on the direct (FWID) or hybrid solver (FWIH).

I first introduce the FWID applications, which have been extracted from the paper Ben-Hadj-Ali et al. (2008), in order to show the expected high-resolution images which can be obtained by FWI and highlight the efficiency of FWID on small to medium problem sizes. Then, I present FWIH applications in order to study the convergence criterion, introduced in the chapter 5 for the hybrid solver and studied for the seismic wave modeling, for the seismic imaging problem.

7.1 FWID applications

In this section, I present several synthetic applications with different complexities in order to validate FWID software and show expected FWI imaging results. The sections 7.1.1 to 7.1.4 are extracted from Ben-Hadj-Ali et al. (2008) with minor modifications.

7.1.1 3D full-waveform inversion in 2D configuration

We applied 3D and 2D FWI to a dip section of the overthrust model (Aminzadeh et al., 1997) (Figure 7.1), discretized on a 801×187 grid with a grid spacing $h = 25$ m. For the 3D application, the dip section of the overthrust model was duplicated 3 times in the y direction leading to a 3D $801 \times 3 \times 187$ finite-difference grid.

The 2D acquisition geometry consists of a line of 200 sources and receivers equally-spaced on the surface. The corresponding 2.5D acquisition geometry consists of duplicating 3 times the source and receiver lines in the y direction. The true model was augmented with a 250-m-thick layer on top of it and the source and receiver were deepened accordingly to avoid having sources and receivers just below the PML-model interface. Velocities in this layer are vertically homogeneous and are equal to that on the surface of the original model. We observed strong instabilities during FWI in the near surface velocities when this layer is not added to the model (Figure 1c in Ravaut et al. (2004)). These instabilities can also be removed by setting the true velocities in the first hundred meters of the starting model without augmenting the model with an artificial layer (Operto et al., 2008).

The starting model for inversion is obtained by smoothing the true velocity model with a Gaussian function of horizontal and vertical correlation lengths of 500 meters (Figure 7.1).

We inverted sequentially 7 frequencies ranging from 5 to 20 Hz. For each frequency, we compute 40 iterations. The final velocity models inferred from 2D and 3D FWI are shown in Figure 7.2. Some vertical profiles extracted from these models are compared in Figures 7.3. They are very similar hence, providing a first validation of the 3D FWI algorithm. The agreement between the final FWI models and the true model is also quite good although we noticed some discrepancies between the true and reconstructed velocities around a low velocity layer located at 0.7-1 km depth (Figure 7.3b). This discrepancy is not observed when the first 100 meters of the true model are set in the starting model (Operto et al., 2008). Some high-amplitude perturbations are still slightly underestimated mainly due to an insufficient number of iterations (the profiles in Figure 7.3 can be compared with those obtained from 2D FWI using 80 iterations per frequency in (Operto et al., 2008)). These results also give some insights on the high spatial resolution which can be achieved in the velocity models at relatively low frequencies (i.e., < 15 Hz) by FWI of global offset data thanks to the continuous sampling of the wavenumber spectrum up to a maximum wavenumber of $2/\lambda_{15Hz}\ m^{-1}$ (λ_{15Hz} denotes the wavelength corresponding to a frequency of 15 Hz).

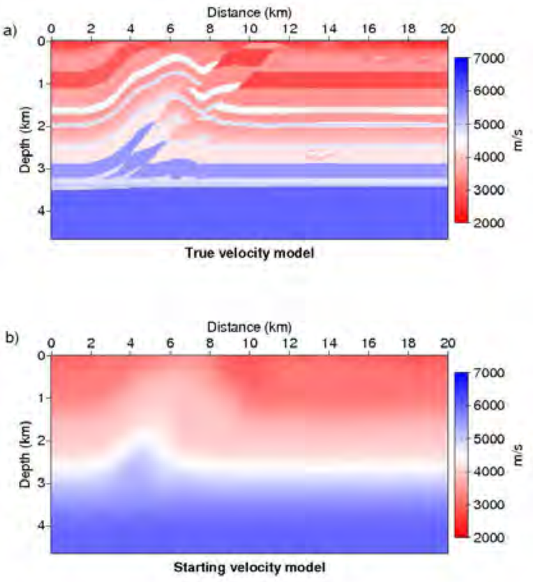

Figure 7.1: Imaging of a dip section of the overthrust model: a) True velocity model. b) Starting velocity model.

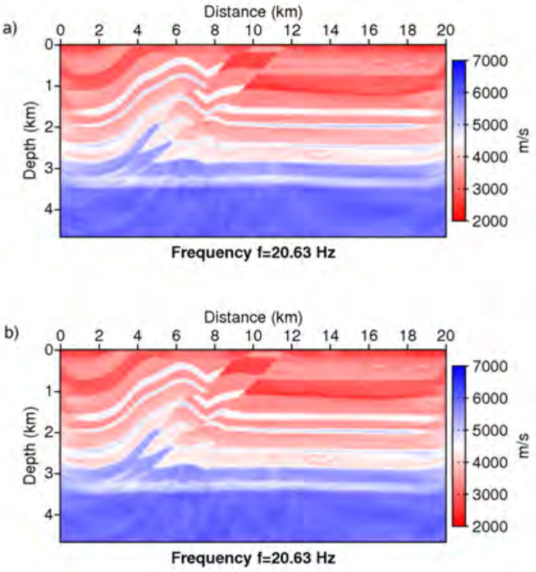

Figure 7.2: Imaging of a dip section of the overthrust model: (a) Final velocity model from 2D FWI. (b) Final velocity model from 3D FWI.

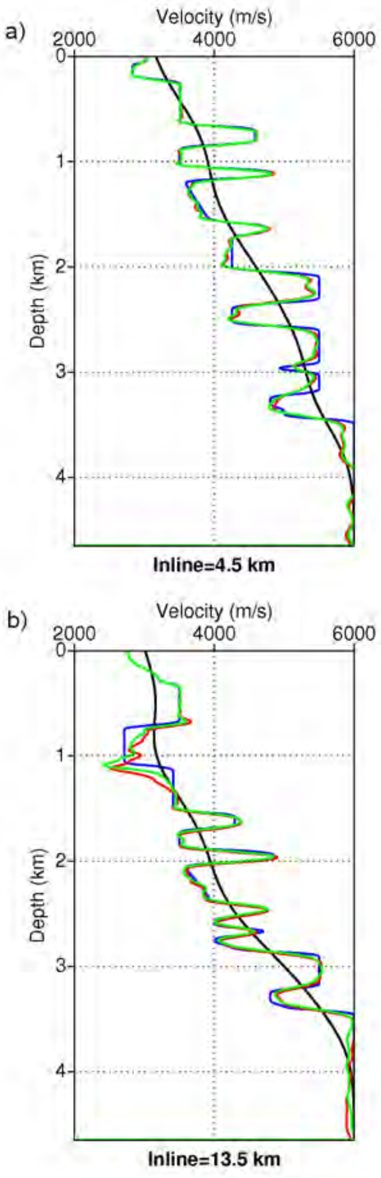

Figure 7.3: Imaging of a dip section of the overthrust model: Comparison between vertical profiles extracted from the true (blue line), the starting (black line) and the 2D and 3D FWI models (red and green lines respectively). The two series of profile are located at 4.5 and 13.5 km of distance. The 2D and 3D FWI profiles are almost identical.

7.1.2 Inclusion models

In this section, we present the application of 3D FWI for simple velocity models composed of homogeneous background with one and two inclusions. The models are discretized on a small 31 × 31 × 31 grid with 50 m and 250 m cubic cells respectively. The velocity in the background medium is 4000 m/s. The inverted frequencies are 3.72, 6.07, 10.00 and 16.27 Hz and 1.75, 2.35, 3.00 and 3.75 Hz for the one-inclusion and two-inclusion models respectively. The criteria to select the inverted frequencies was to remove the wavenumber redundancy in the model space (Sirgue and Pratt, 2004). The frequencies were inferred from the relationship between wavenumber, frequency and aperture angle provided by the theory of diffraction tomography. This relation leads to an increasing frequency interval with frequency due to the linear relation between wavenumber and frequency (Sirgue and Pratt, 2004). One hundred sources (10x10) and 36 receivers (6x6) are uniformly distributed on the top and bottom sides of the 3D model respectively. Source and receiver spacing are 150 m and 250 m respectively for the single-inclusion model and 750 m and 1250 m respectively for the two-inclusion model.

We first consider the case of a velocity model with one inclusion in the homogeneous background. The velocity in the inclusion is 3500 m/s (Figure 7.4). The inclusion is centered on the 3D grid. The 4 frequencies were inverted successively. Some horizontal and vertical sections of the inclusion are shown in Figure 7.4. Note the vertically-elongated shape of the inclusion in the vertical cross-section section and the symmetric shape of the inclusion in the horizontal slice. The vertically-elongated shape of the inclusion is due to the fact the top and bottom parts of the inclusion are mainly sampled by downgoing transmitted wave paths (i.e., forward-scattered wavepaths) which have a limited resolution power while the shape of the inclusion in a horizontal plane is mainly controlled by reflections (i.e., backward-scattered wavepaths) associated with shots and receivers located near a same face of the 3D model. This relationship between the aperture illumination and the resolution of the imaging is also illustrated on the two profiles extracted from a vertical and horizontal section running through the inclusion. The vertical profile exhibits a clear deficit of high wavenumbers due to transmission-like reconstruction while the horizontal profile exhibits slight deficit of small wavenumbers due to reflection-like reconstruction. The symmetry of the image of the inclusion in the horizontal plane which results from the symmetry of the inclusion with respect to the acquisition geometry is an additional validation of the 3D FWI algorithm.

The second example contains two spherical inclusions (3500 m/s and 4500 m/s) corresponding to a positive and negative perturbations in the homogeneous background (Figure 7.5(a)). The center of the two inclusions lies on the same vertical plane in the middle of the grid. The aim of this test is to verify that the 3D inversion handles properly multiple scattering occurring between the two inclusions. The starting model, the frequencies involved in the inversion and the acquisition geometry are the same as those of the previous example. For this case study, the 4 frequencies were inverted both successively and simultaneously (Figures 7.5(b-e) and 7.5(f)). In both cases, the inversion successfully imaged the two inclusions.

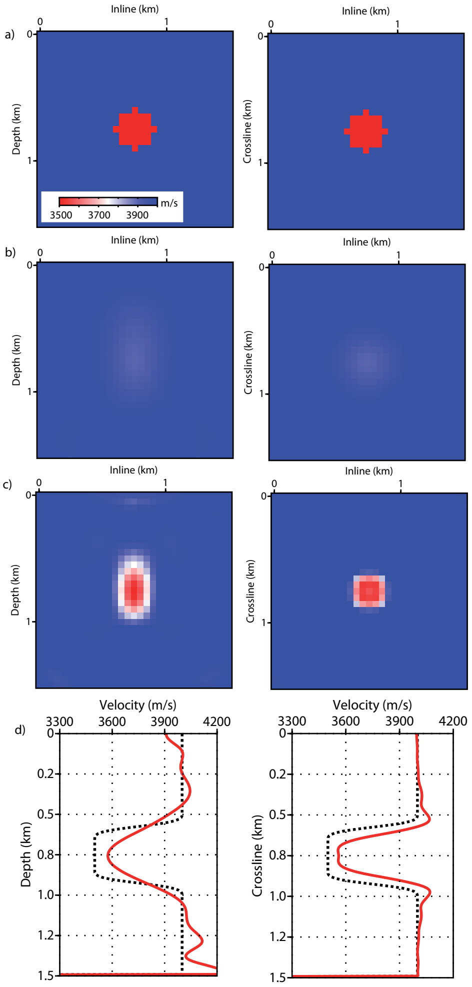

Figure 7.4: Imaging of 1 inclusion by 3D FWI. a) vertical (left) and horizontal (right) sections of the true inclusion. b) vertical (left) and horizontal (right) sections of the inclusion after inversion of the 3.72-Hz frequency. c) vertical (left) and horizontal (right) sections of the inclusion after inversion of the 16.27-Hz frequency. d) vertical (left) and horizontal (right) profiles extracted from models shown in (a) (black lines) and (c) (red lines).

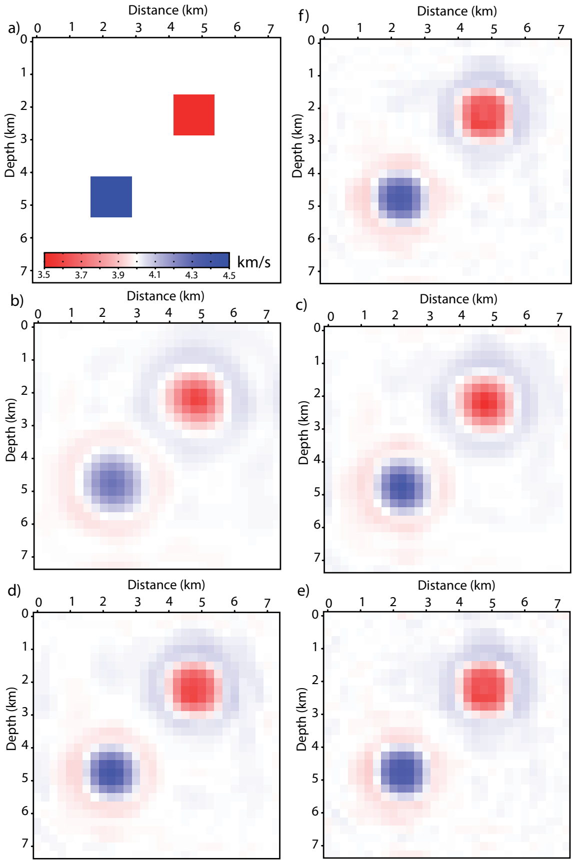

Figure 7.5: Imaging of 2 inclusions by 3D FWI: a) Imaging of 2 inclusions by 3D FWI: true model. (b-e) Vertical cross-sections of the FWI velocity models after successive inversion of frequencies 1.75 (b), 2.35 (c), 3.00 (d) and 3.75 (e) Hz. f) Vertical cross-section of the 3D FWI velocity model after simultaneous inversion of the 4 frequencies 1.75, 2.35, 3.00 and 3.75 Hz.

7.1.3 Inclusion+interface velocity model

A more realistic example consists of a velocity-gradient layer above a homogeneous layer. An high-velocity inclusion corresponding to a velocity perturbation of more than 1km/s was incorporated into the velocity-gradient layer (Figure 7.6). The minimum and maximum velocities are 3.8 and 6.0 km/s respectively. It is discretized on a $100 \times 100 \times 40$ grid with a grid spacing $h = 62.5$ m, which corresponds to a physical domain of 6.25 km x 6.25 km x 2.5 km. The grid spacing h was kept constant over the successive mono-frequency inversions and was set according to the maximum inverted frequency. The starting model for inversion is the velocity-gradient layer extended down to the bottom of the model (Figure 7.6). The acquisition geometry consists of two regular grids of $17 \times 17 = 289$ sources and receivers deployed on the surface. The receiver grid is shifted with respect to the shot one such that each receiver is midway between 4 adjacent shots. This source-receiver configuration was chosen to avoid recording high-amplitude zero-offset data which degrades the inversion conditioning. The distance between either two sources or receivers is 312.5 m. We sequentially inverted 5 frequencies ranging from 1.76 and 12.15 Hz. We computed 20 iterations per frequency. The final FWI velocity model is shown in Figure 7.7(a). A vertical profile across the inclusion extracted from the final FWI perturbation model (i.e., the difference between the final FWI model and the starting model) is shown in Figure 7.7(b). It is compared with that extracted from the true perturbation model after low-pass filtering at the theoretical resolution of FWI at 12 Hz. The bottom layer is well recovered thanks to the large offset coverage allowing to quantitatively image a broad range of the layer wavelengths. The shape of the inclusion is incompletely recovered with respect to the expected resolution of the imaging at 12 Hz although the velocity amplitude in the inclusion is fully recovered (Figure 7.7(b-c)). The spectra of the two profiles reveals that the amplitudes of the low wavenumbers were incompletely recovered. This is probably due to an insufficient number of iterations as suggested by the plot of the objective function as a function of iteration number (Figure 7.7(c)). Moreover, some discrepancies in the shape of the two spectra of Figure 7.7(c) for wavenumbers higher than 0.0017 m^{-1} suggest that the inversion may have converged towards a local minimum. This may have been caused by the high-amplitudes of the model perturbations and the related complex interactions between waves multi-scattered between the bottom of the inclusion and the top of the layer which make the inverse problem more non linear. We speculate that simultaneous inversion of multiple frequencies following the multi scale approach of Bunks et al. (1995) may help to manage this non linearity. Qualitative inspection of the vertical profiles also reveals a slight deficit of small (vertical) wavenumbers in the image of the inclusion and of the bottom layer (this is suggested by the negative velocity perturbations with respect to the true model). This deficit is again explained by the surface acquisition geometry which illuminates the vertical components of the wavenumber vector with reflections only.

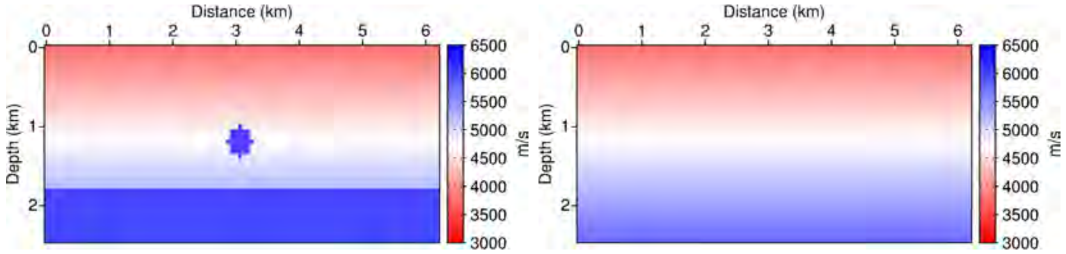

Figure 7.6: Imaging of the inclusion/interface model. (right) True velocity model. (left) Starting velocity model for FWI.

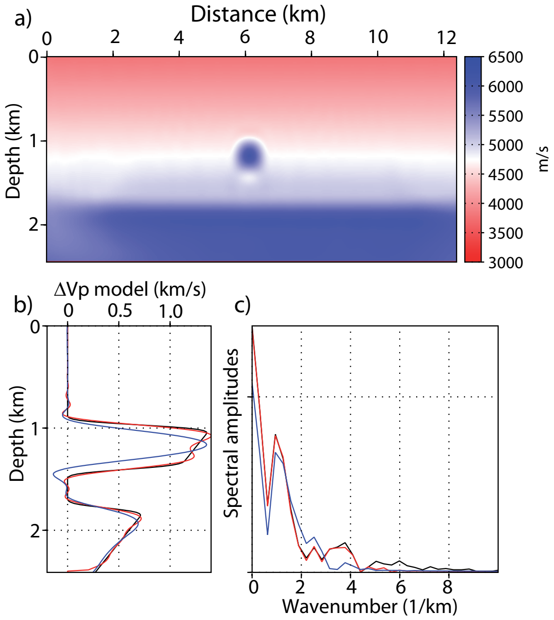

Figure 7.7: Imaging of the inclusion/interface model: (a) Final FWI velocity model. (b) Vertical profile across the inclusion extracted from the true perturbation velocity model (black), from the true perturbation velocity model after low-pass filtering at the theoretical resolution of 12.15-Hz FWI (red) and from the final FWI perturbation velocity model. (c) Spectral amplitudes as a function of wavenumber of the profiles shown in b). These spectra suggest that wavenumbers between 0.003 and 0.004 m^{-1} corresponding to wavelengths of 333 and 250 m respectively were not successfully imaged.

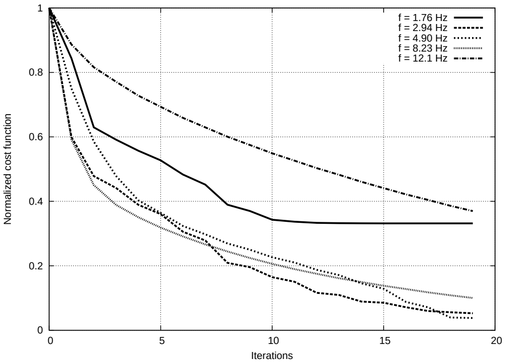

Figure 7.8: Imaging of the inclusion/interface model: objective function versus iteration number for each inverted frequency. Note that the convergence was not achieved which can partially explain underestimated velocities.

7.1.4 SEG/EAGE overthrust model

The 3-D SEG/EAGE overthrust model is a constant density acoustic model covering an area of 20 km x 20 km x 4.65 km (Aminzadeh et al., 1997). It is discretized with 25 m cubic cells, representing an uniform mesh of 801 × 801 × 187 nodes. The minimum and maximum velocities in the overthrust model are 2.2 and 6.0 km/s respectively (Figure 7.9).

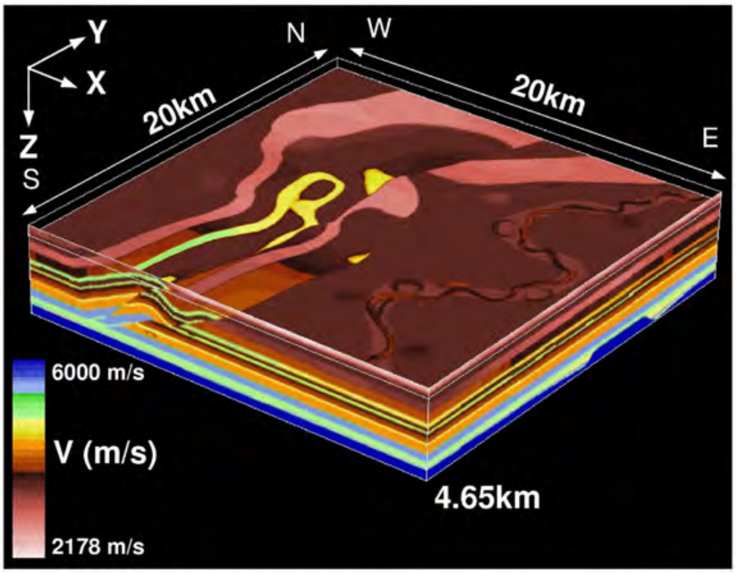

Figure 7.9: The 3D SEG/EAGE overthrust model.

7.1.4.1 Overthrust model: channel target

Due to limited available computer resources, our first application was restricted to a small section of the overthrust model centered on a channel. The maximum frequency involved in the inversion was 15 Hz. A horizontal slice and a vertical section of the model are shown in Figure 7.11. The model covers an area of 7. km x 8.75 km x 2.25 km and is discretized with a grid spacing $h = 50$ m. This leads to a 141 × 176 × 46 grid. The minimum and maximum velocities are 3.3 and 6.0 km/s respectively. The acquisition geometry consists of two regular grids of 44x33 sources and receivers on the surface corresponding to a total of 1452 sources and receivers. The distance between either two sources or receivers is 200 m. The receiver array is shifted according to the source array following the same geometry that of the previous example. We sequentially inverted 5 frequencies ranging from 5 to 15 Hz. For each frequency, we computed 7 iterations. The starting velocity model was obtained by smoothing the true model with a wavenumber filter with a cut-off wavenumber of 1./500 m^{-1} (Figure 7.11). The final FWI model provides a low-pass version of the true model (Figure 7.12). In order to assess the accuracy of the FWI, we low-pass filter the true model in the time domain with a cut-off frequency of 15 Hz to roughly mimic the exact velocity model that would have been inferred by FWI (Figure 7.13). Qualitative comparison between the final FWI velocity model and the low-pass filtered true model shows a good agreement between the two models. Comparison between a vertical profile extracted from the starting model, the low-pass true velocity model and the final FWI model is shown in Figure 7.14.

The agreement is reasonably good with again a slight deficit of small wavenumbers in the FWI profile due to the surface-to-surface illumination. We note also an underestimation of velocities in the deep part of the model (see the high-velocity layer above 2 km depth in Figure 7.14). They may be due to an insufficient number of iterations. Indeed, the deep structures are mainly constrained by later-arriving reflections of smaller amplitude recorded at larger offsets. Misfit reduction may be slower for these arrivals since the value of the objective function is dominated by the residuals of the high-amplitude shallow arrivals during the first iterations. The weighting operator in the data space corresponding to an amplitude gain with offset, eq. (5.25) may help to speed up the reduction of the long-offset residuals at the partial expense of the short-offset ones during late iterations. However, this strategy requires to check that short-offset residuals were sufficiently reduced to avoid propagating errors associated with inaccurate shallow structures deeper in the model. This detailed tuning of 3D FWI still requires further investigations.

To perform this application, we used 60 MPI processes distributed over 15 dual-core biprocessor nodes. Each MPI process used 1.5 Gbytes of RAM (see Table 7.1). Seven iterations of the inversion of one frequency took approximately 45 hours. Table 7.1 gathers information related to running time and memory requirement for LU factorization, multi-shot resolutions (both tasks being devoted to the forward problem), gradient and diagonal Hessian computation. Running time for the solution phase is very small (0.9 s per source) and illustrates the main advantage of frequency-domain modeling methods based on direct solvers for tomographic applications involving few thousands of source. Computation of the gradient is also negligible in the frequency domain (4 s) thanks to the summation without disk swapping over a very compact volume of data limited to few frequency components. Moreover, increasing the number of cores in the inversion would have led to a significant reduction of the computational time at the partial expense of memory saving due to memory overhead during parallel factorization (Operto et al., 2007).

Table 7.1: Computational cost of the imaging of the overthrust model (channel)

$MEM_{FACTO_{ALL}}$ $(GBytes)$	67.
$MEM_{FACTO_{PROC}}$ $(GBytes)$	1.5
$TIME_{FACTO}$ (s)	510.
$TIME_{SOLVE_{ALL}}$ (s)	1270.
$TIME_{SOLVE_{SOURCE}}$ (s)	0.9
$TIME_{GRADIENT}$ (s)	4.
$TIME_{diagHESSIAN_a}$ (s)	3093.
$TIME_{ITERATION}$ (s)	18865.

$MEM_{FACTO_{ALL}}$ – Total memory allocated during factorization.
$MEM_{FACTO_{PROC}}$ – Average allocated memory per working processor during factorization.
$TIME_{FACTO}$ – Elapsed time for factorization.
$TIME_{SOLVE_{ALL}}$ – Total elapsed time for multi-shot resolution.
$TIME_{SOLVE_{SOURCE}}$ – Elapsed time for resolution for 1 source.
$TIME_{GRADIENT}$ – Elapsed time for gradient computation.
$TIME_{diagHESSIAN_a}$ – Elapsed time for diagonal Hessian computation.
$TIME_{ITERATION}$ – Average elapsed time for processing one iteration.

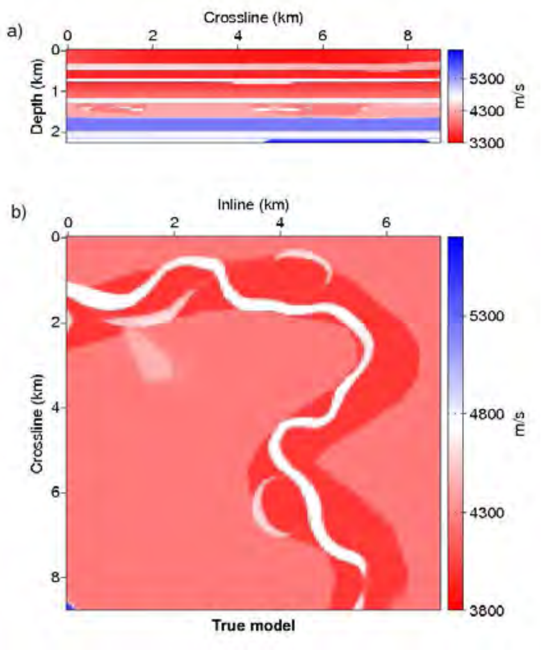

Figure 7.10: Imaging of a channel in the overthrust model: true velocity model. a) Cross-section at x=4 km. b) Horizontal slice at z=1.5 km.

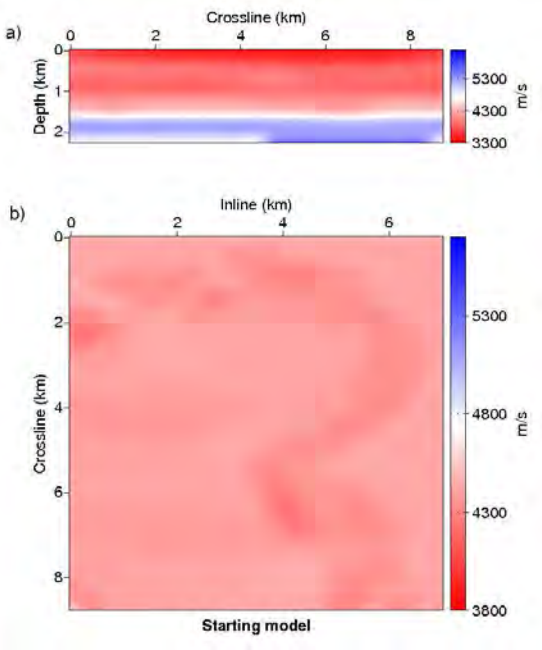

Figure 7.11: Imaging of a channel in the overthrust model: starting velocity model. a) Cross-section at X=4 km. b) Horizontal slice at Z=1.5 km.

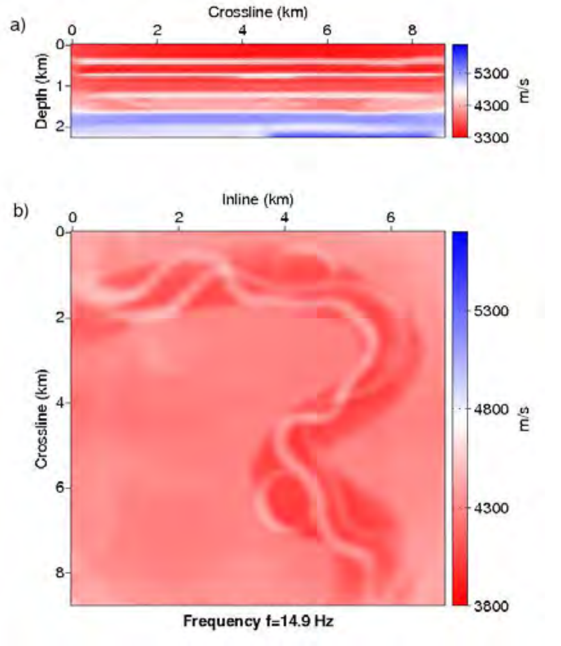

Figure 7.12: Imaging of a channel in the overthrust model: FWI velocity model after successive inversion of the 5 frequencies. a) Cross-section section at x=4 km. b) Horizontal slice at z=1.5 km.

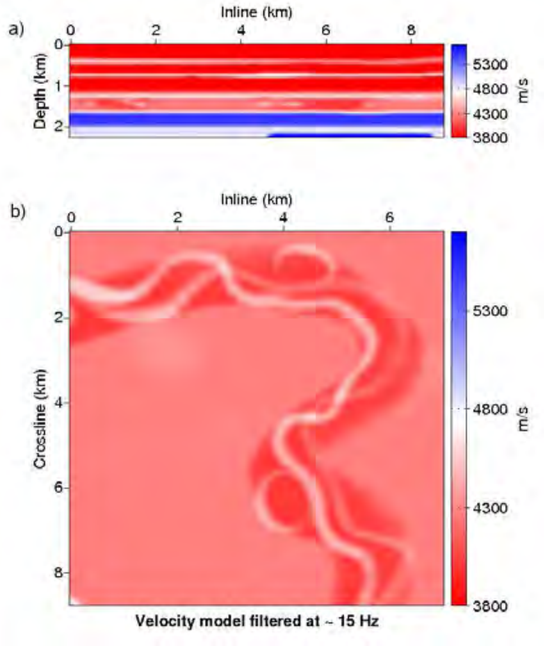

Figure 7.13: Imaging of a channel in the overthrust model: low-pass filtered true velocity model. a) Cross-section at X=4 km. b) Horizontal slice at Z=1.5 km.

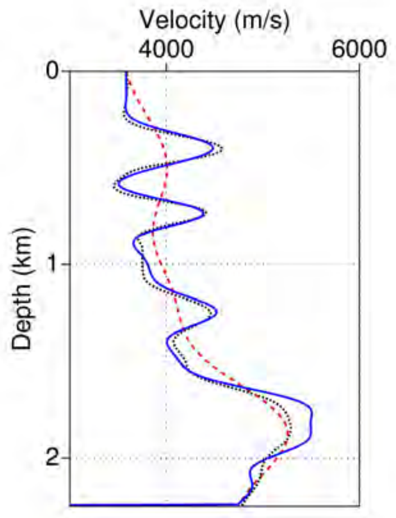

Figure 7.14: Imaging of a channel in the overthrust model: comparison between vertical profiles extracted from the starting model (red dashed line), the low-pass filtered true model (blue solid line) and the final FWI velocity model (black dot line). The profile is located at $(X = 3.5 \text{ km}, Y = 4.4 \text{ km})$.

7.1.4.2 Overthrust model: thrust target

We now consider the imaging of a significant target of the overthrust model which incorporates the main thrusts of the model (Figure 7.15). The minimum and maximum velocities are respectively 2.2 and 6.0 km/s. The model covers an area of 13.425 km x 13.425 km x 4.65 km.

The acquisition geometry consists of two coincident $43 \times 43 = 1849$ grid of sources and receivers deployed on the surface. The distance between either two sources or receivers is 300 m. A receiver spacing of 300 meters is representative of the spacing between 2 adjacent nodes in a dense 3D wide-azimuth node survey (Clarke et al., 2007). We used the same spacing between 2 adjacent shots and receivers although a more representative shot survey could have been designed by using smaller shot and line intervals in the dipline and crossline directions respectively. The increased number of shots should not lead to a dramatic increase of the computational time for the gradient estimation since the residuals recorded at the shot positions (by virtue of shot-receiver reciprocity) can be propagated in one go for each receiver. The extra computational cost caused by denser shot survey would have resulted from the building and storage of denser residual sources (term $\delta \mathbf{d}^*$ in eq. (5.25)) and from the more expensive backward/forward substitutions providing the solutions of $\mathbf{A}^{-1}\mathbf{C}_d\delta\mathbf{d}^*$. On contrary, the CPU time required to compute the diagonal Hessian would dramatically increase since it requires a forward simulation per non redundant shot and receiver positions. However, a good approximation of the diagonal Hessian may be computed on a coarser shot grid with a shot interval of the same order that of the receiver one (Operto et al., 2006b). We sequentially inverted 3 frequencies: 3.5, 5 and 7 Hz. We computed 10 iterations per frequency. For this application, we adapted the grid interval to the inverted frequency. Grid intervals were $h = 150$ m, 100 m and 75 m for the frequencies 3.5, 5 and 7 Hz respectively. Note that the source and receiver positions were chosen such that they coincide with the position of the nodes of the FD grids associated with the 3 inverted frequencies. This allowed us to bypass the problem of accurate implementation of point sources in a coarse FD grid which is a critical issue of 3D frequency-domain FWI when the grid interval is adapted to the frequency (Hicks, 2002).

These discretizations lead respectively to grids of dimension $90 \times 90 \times 32$, $135 \times 135 \times 47$ and $180 \times 180 \times 63$. The starting velocity model was obtained by smoothing the true model with a 3D Gaussian function with a correlation length of 500 meters in the 3 directions (Figure 7.16). The FWI velocity models after inversion of the 3.5-Hz, 5-Hz and 7-Hz frequencies are shown in Figures 7.17, 7.18 and 7.19 respectively. One can note a square pattern superimposed on the horizontal and vertical slices of the FWI velocity models obtained close of inversions of frequencies 5 and 7 Hz (Figures 7.18 and 7.19). Size of the square matches the shot and receiver spacing suggesting that it corresponds to the footprint of the coarse acquisition geometry. This footprint increases from 5 to 7 Hz as the model resolution increases. The acquisition footprint has no preferential orientation due to the fact that both shots and receivers are uniformally deployed all over the surface with a constant spacing in the dip and cross directions. Another illustration of the footprint of the acquisition coarseness on 3D frequency-domain FWI was illustrated by Sirgue et al. (2007a).

For this application, we run 32 processes distributed over 8 dual-core biprocessor nodes (4 MPI processes/node) for the 150-m grid (frequency 3.5 Hz), 60 processes distributed over 20 dual-core biprocessor nodes (3 MPI processes/node) for the 100-m grid (frequency 5 Hz) and 90 processes distributed over 30 dual-core biprocessor nodes (3 MPI processes/node) for the 75-m grid (frequency 7 Hz) respectively. Note that the number of process per dual-core biprocessor node was decreased from 4 to 3 as the size of the problem increases in order to increase the amount of shared memory assigned to each processor for large problems. This allows optimization of the memory use at the partial expense of the running time since the memory overhead decreases with the number of processes.

The 10 iterations took about 24 hours, 72 hours and 120 hours for the 3.5-Hz, the 5-Hz and the 7-Hz frequencies respectively. More detailed information are gathered in the Table 7.2.

Figure 7.20 shows the convergence rate for each processed frequency. This convergence rate can be compared with that shown in Figure 7a of Sirgue et al. (2007a) for the frequency 3.5-Hz keeping in mind that the whole overthrust model was imaged in Sirgue et al. (2007a).

The data fit is illustrated in the frequency domain for the 3.5-Hz, 5-Hz and 7-Hz frequencies for 2 shots in Figures 7.21 and 7.22 respectively. We compare the spectral amplitude and the phase of the monochromatic wavefields at the receiver positions computed in the true velocity model and in the FWI models at the first and last iterations of the three mono-frequency inversions. One shot is located at the upper-left corner of the receiver plane (Figure 7.21) while the second shot is on the middle of the receiver array (Figure 7.22). The misfit reduction between the first and last iterations is more obvious at 3.5 Hz illustrating slower convergence as the frequency increases (Figure 7.20). We note also that this misfit reduction is more effective for the shot located in the middle of the receiver array illustrating again the difficulty to match the lower-amplitude arrivals recorded at larger offsets. As for the channel case study, the objective function is less sensitive to the residuals of these low-amplitude arrivals and more iterations would have been required to cancel them.

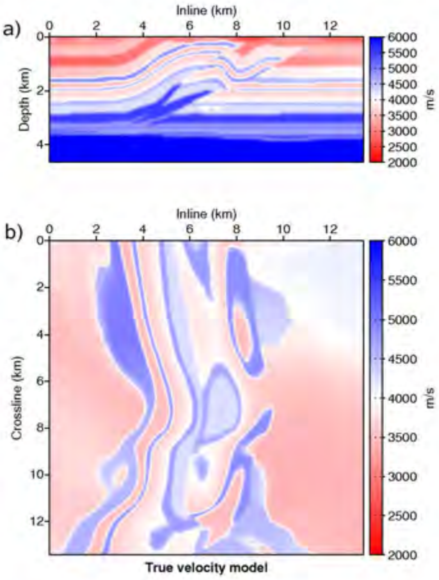

Figure 7.15: Imaging of the thrust system in the overthrust model: true velocity model. a) Cross-section at X=3.3 km. b) Horizontal slice at Z=1.5 km.

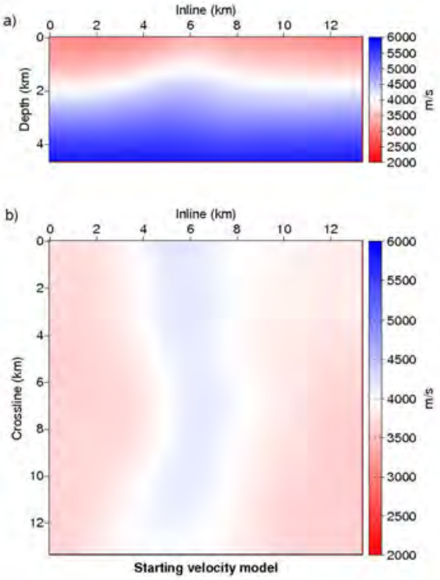

Figure 7.16: Imaging of the thrust system in the overthrust model: starting velocity model. a) Cross-section at X=3.3 km. b) Horizontal slice at Z=1.5 km.

Table 7.2: Computational cost of the imaging of the overthrust model (thrust system)

$FREQUENCY\ (Hz)$	3.5	5	7
$GRID$	$90 \times 90 \times 32$	$135 \times 135 \times 47$	$180 \times 180 \times 63$
$MEM_{FACTO_{ALL}}\ (GBytes)$	16.	64.	175.
$MEM_{FACTO_{PROC}}\ (GBytes)$	0.4	1.1	2.3
$TIME_{FACTO}\ (s)$	72.	340.	1850.
$TIME_{SOLVE_{ALL}}\ (s)$	310.	995.	3450.
$TIME_{SOLVE_{SOURCE}}\ (s)$	0.165	0.535	1.795
$TIME_{GRADIENT}\ (s)$	0.65	1.44	35.
$TIME_{diag_{HESSIAN_a}}\ (s)$	1999.	3432.	4000.
$TIME_{ITERATION}\ (s)$	2940.	13650.	44870.

$FREQUENCY$ – Inverted frequency
$GRID$ – Dimension of the 3D FD grid
$MEM_{FACTO_{ALL}}$ – Total memory allocated during factorization
$MEM_{FACTO_{PROC}}$ – Average allocated memory per working processor during factorization
$TIME_{FACTO}$ – Elapsed time for factorization
$TIME_{SOLVE_{ALL}}$ – Total elapsed time for multi-shot resolution
$TIME_{SOLVE_{SOURCE}}$ – Elapsed time for resolution of 1 source
$TIME_{GRADIENT}$ – Elapsed time for gradient computation
$TIME_{diag_{HESSIAN_a}}$ – Elapsed time for diagonal Hessian computation
$TIME_{ITERATION}$ – Average elapsed time for processing one iteration

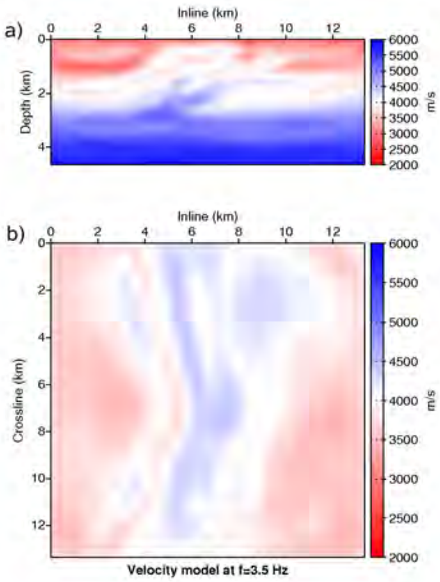

Figure 7.17: Imaging of the thrust system in the overthrust model: 3.5-Hz FWI velocity model. a) Cross-section at X=3.3 km. b) Horizontal slice at Z=1.5 km.

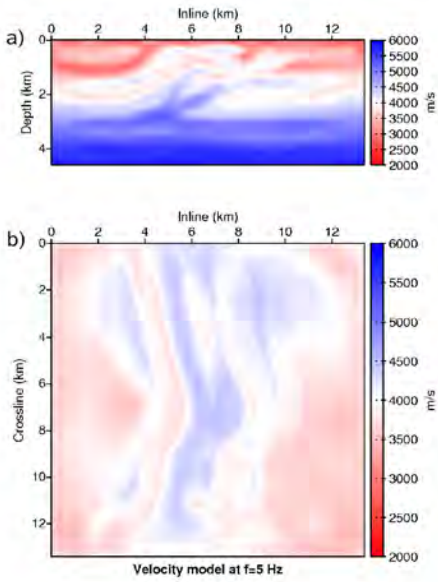

Figure 7.18: Imaging of the thrust system in the overthrust model: 5-Hz FWI velocity model. a) Cross-section at X=3.3 km. b) Horizontal slice at Z=1.5 km.

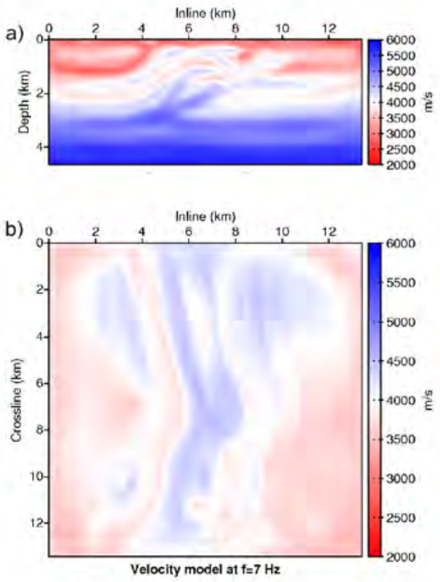

Figure 7.19: Imaging of the thrust system in the overthrust model: 7-Hz FWI velocity model. a) Cross-section at X=3.3 km. b) Horizontal slice at Z=1.5 km.

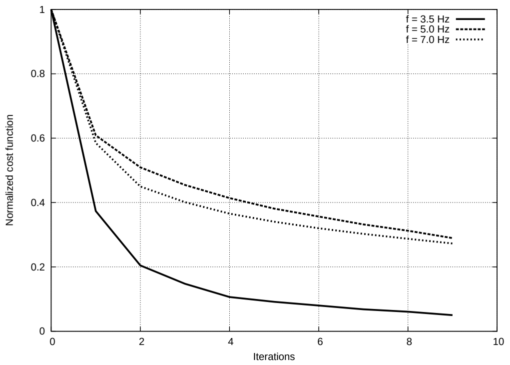

Figure 7.20: Imaging of the thrust system in the overthrust model: objective function versus iteration number for the 3.5-Hz, 5-Hz and 7-Hz frequencies.

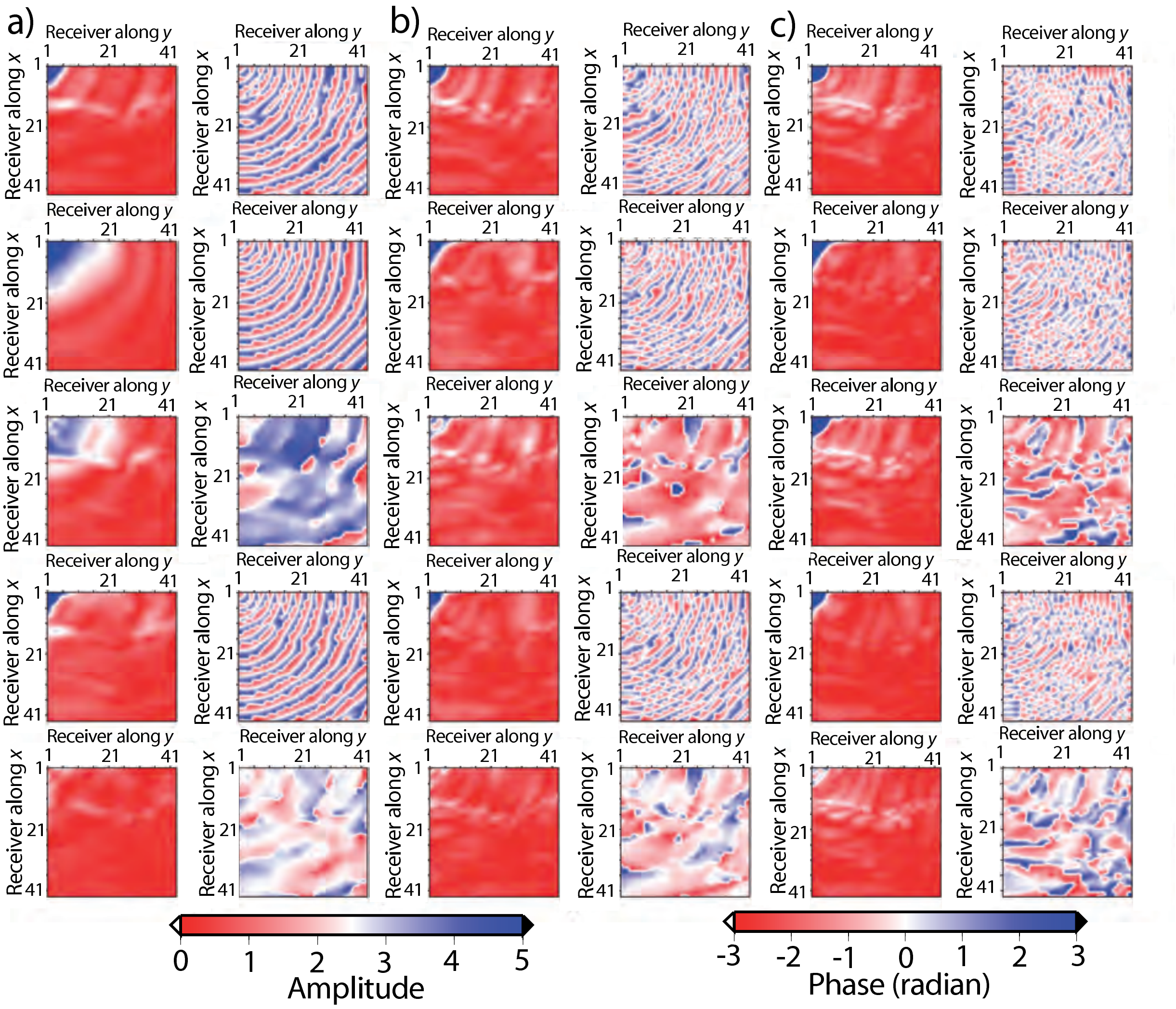

Figure 7.21: Imaging of the thrust system in the overthrust model: a) amplitude (left) and phase (right) of the 3.5-Hz wavefield computed in the true velocity model at the receiver positions. The horizontal and vertical axis label the receiver number in the dip and cross directions respectively. The source is located in the upper-left corner of the receiver array. b) same than (a) but the wavefields were computed in the starting model of the 3.5-Hz inversion. c) Difference between maps shown in a) and b). d) Same than (a) but the wavefields were computed in the final model of the 3.5-Hz inversion. e) Difference between maps shown in (a) and (d). (f-j): same as for (a-e) but for the 5-Hz frequency. (k-o): same as for (a-e) but for the 7-Hz frequency.

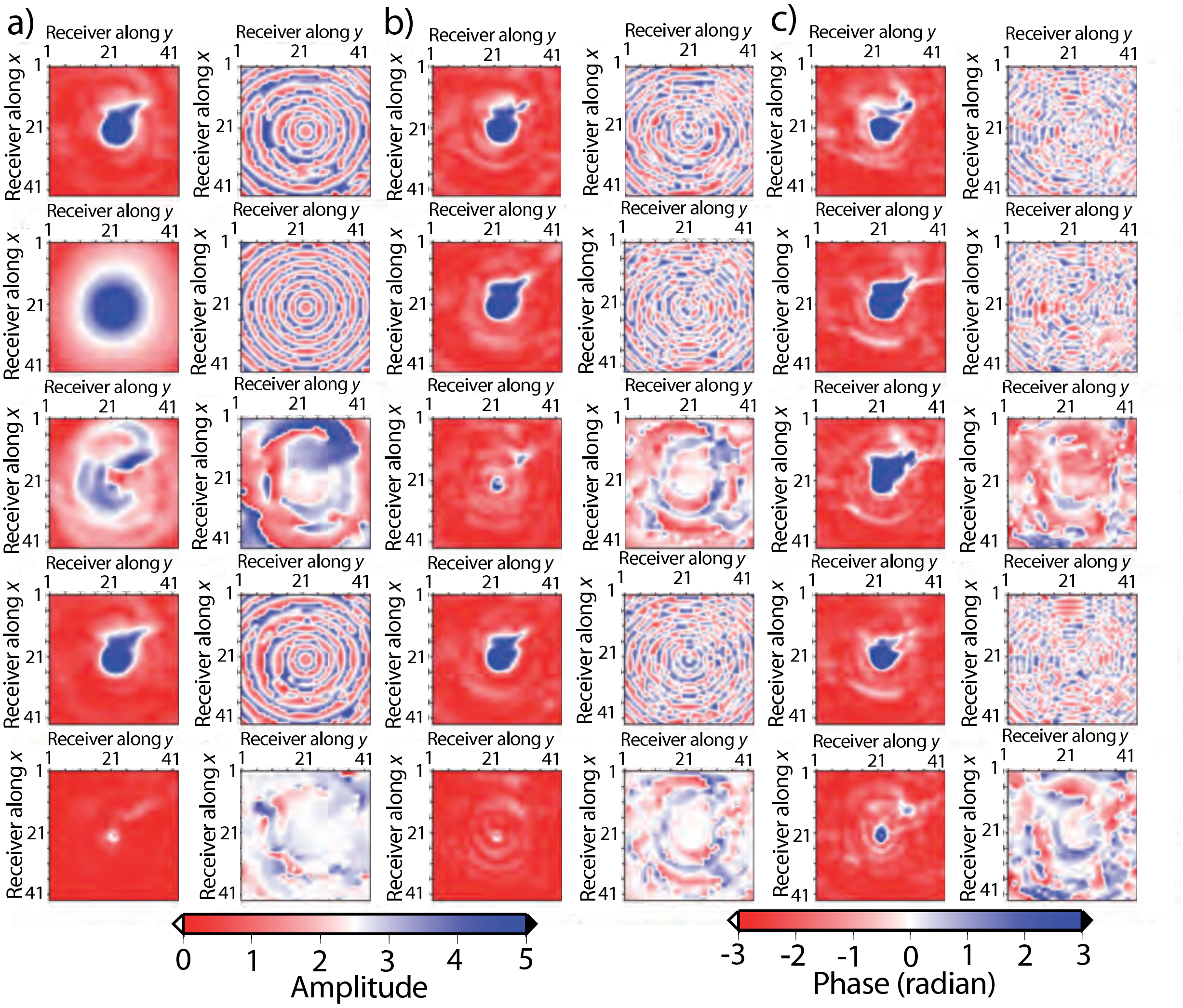

Figure 7.22: Imaging of the thrust system in the overthrust model: Same as for Figure 7.21 but for a source located in the middle of the receiver array.

7.2 FWIH applications

Although the accuracy analysis of the hybrid solver has shown that the optimal ϵ value is 10^{-3} for the seismic wave modeling, many precision values of ϵ were tested ($10^{-3} \leq \epsilon \leq 10^{-1}$) for the FWIH approach to investigate this statement for imaging.

7.2.1 Single anomaly model

Aim of the first example is to image a single sphere anomaly in a homogeneous background model. The model is discretized on a small $31 \times 31 \times 31$ uniform grid with a grid interval of 50 m. The velocity in the background medium is 4000 m/s. The velocity in the anomaly is 3500 m/s. The anomaly is centered on the 3-D grid. Sources and receivers (6×6) are uniformly distributed on the top and bottom, left and right, front and back sides of the 3-D model respectively. Source and receiver spacing is 250 m. The inverted frequencies are 3.72, 6.07, 10.00 and 16.27 Hz. The 4 frequencies are inverted successively. For FWIH simulations, the full domain is splitted into 8 equal subdomains. Two data sets are computed in the true model with the direct and hybrid solver ($10^{-3} <= \epsilon <= 10^{-1}$) respectively. Both data sets were used to perform inversion using the FWID and FWIH algorithms in order to verify the consistency between the solutions computed with the direct and hybrid solver. The results are gathered in the Figures 7.23 to 7.27. The figures show that the precision value $\epsilon = 10^{-1}$ is not sufficient and that imaging results are not acceptable (Figure 7.24) while for $\epsilon \leq 10^{-2}$ (Figure 7.25) the two approaches lead to approximately the same results. This is encouraging in the meaning that this shows that the precision criterion is less limiting in imaging than modeling problem ($\epsilon = 10^{-3}$). This statement should be confirmed on more complex examples.

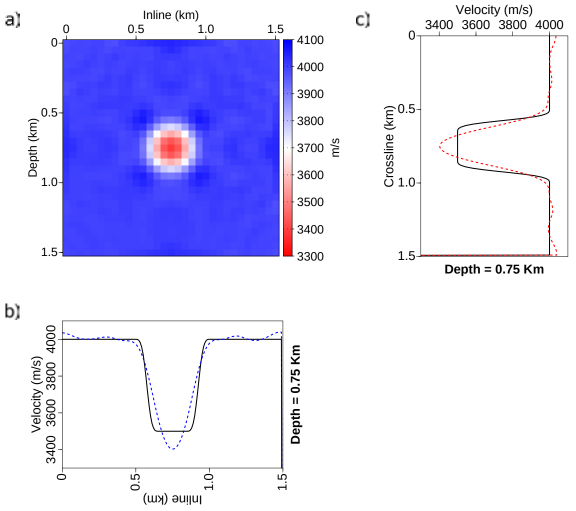

Figure 7.23: Velocity anomaly imaged by FWID in an inverse crime configuration.

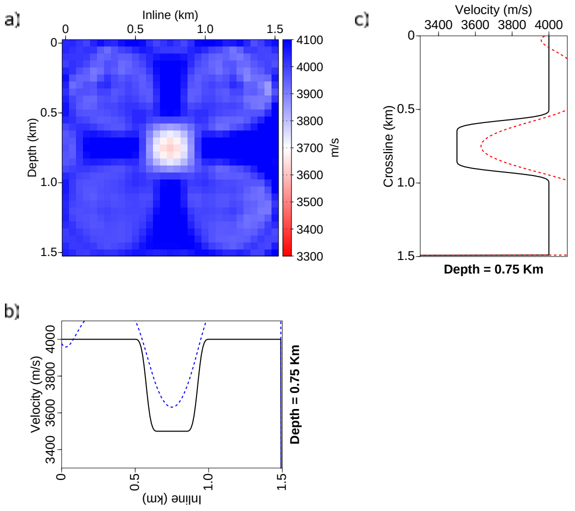

Figure 7.24: Velocity anomaly imaged by FWIH ($\epsilon = 10^{-1}$) from data computed with direct solver.

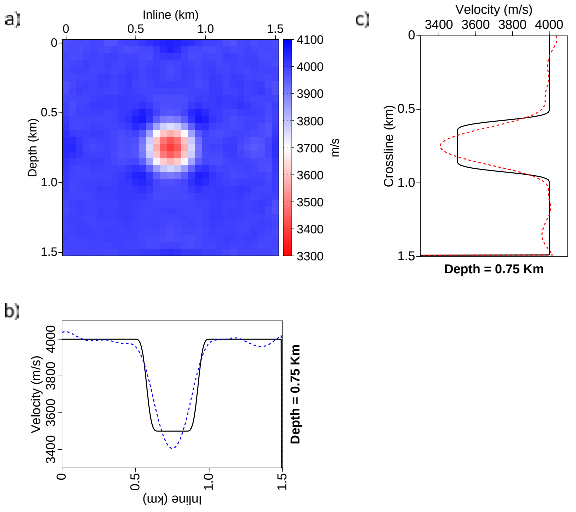

Figure 7.25: Velocity anomaly imaged by FWIH ($\epsilon = 10^{-2}$) from data computed with direct solver.

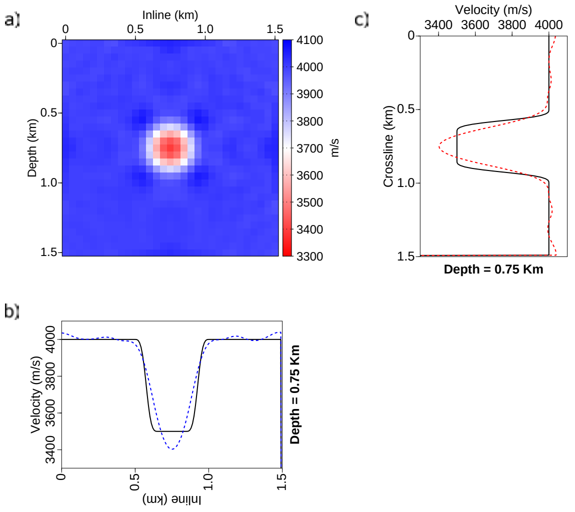

Figure 7.26: Velocity anomaly imaged by FWIH ($\epsilon = 10^{-3}$) from data computed with direct solver.

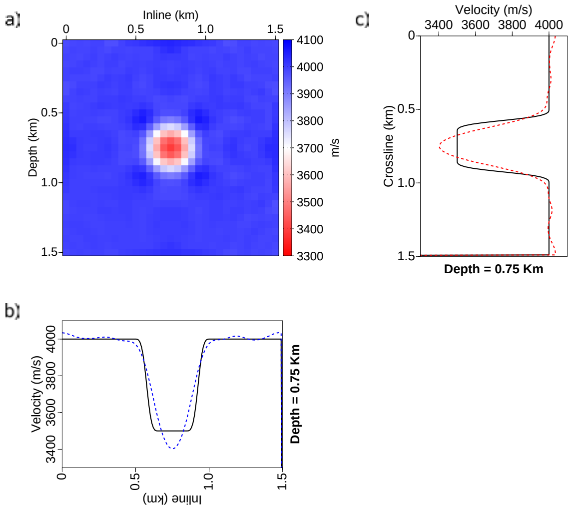

Figure 7.27: Velocity anomaly imaged by FWIH ($\epsilon = 10^{-3}$) in an inverse crime configuration.

7.2.2 SEG/EAGE overthrust velocity model - Channel target

Aim of the second example is to image a complex velocity model which consists in a small target of the 3-D SEG/EAGE Overthrust velocity model focusing on the channel. A horizontal slice and a vertical section of the target are shown in Figure 7.28. The model covers an area of $4.95 \times 4.95 \times 2.25\ km^3$. It is discretized on a $100 \times 100 \times 46$ uniform grid with an interval of 50 m corresponding to 46×10^4 unknowns. The minimum and maximum velocities are 3.3 and 6.0 km/s respectively. The acquisition geometry consists of two regular grids of (89×89) sources and (12×12) receivers on the surface. Source and receiver spacing are 50 m and 400 m respectively. The starting velocity model is obtained by smoothing the true model with a wavenumber filter with a cut-off wavenumber of 1/500 m^{-1} (Figure 7.29). I sequentially invert 5 frequencies ranging from 5 to 14 Hz. For each frequency, I compute 7 iterations. I use 32 processes for both FWID and FWIH. I compare results obtained by FWID and FWIH applied data which are computed in the true model either with the direct or hybrid solver $(10^{-3} \leq \epsilon \leq 10^{-2})$.

Results are shown in the Figures 7.30 to 7.33 respectively. Models obtained with FWID and FWIH are almost identical providing an additional validation of the FWIH code and confirming that a precision $\epsilon = 10^{-2}$ is sufficient to get reliable results. The main computational cost indicators are gathered in Table 7.3. The FWIH code requires 30 % less memory than the FWID one. The limited memory saving provided by FWIH results from the small size of the computational grid (46×10^4) and from the limited number of processors (32). The more the size of the model and the number of processor will increase, the more the memory saving provided by FWIH will be significant according to the theoretical memory complexities provided in Table 3.2. Time to perform FWIH simulations is 3.7 higher that required by FWID. This results from the cost of the iterative solver used in the hybrid method when multiple-shot simulations are performed. Given that the scalability of the iterative solver is better than that of the LU factorization, the time requirement of FWIH should get closer to that of FWID when the size of the computational domain and the number of processors will increase for a given number of sources.

Table 7.3: Computational resources requirements of FWID and FWIH.

	Mem (Gb)	T_f(s)	T_s(s)	T_total(s)
Direct solver	19.2	191	0.5	24700
Hybrid solver	13.5	17	13	91000

Mem – Total allocated memory
T_f – Average elapsed time for factorization
T_s – Elapsed time for the solution step
T_total – Average elapsed time to invert one frequency

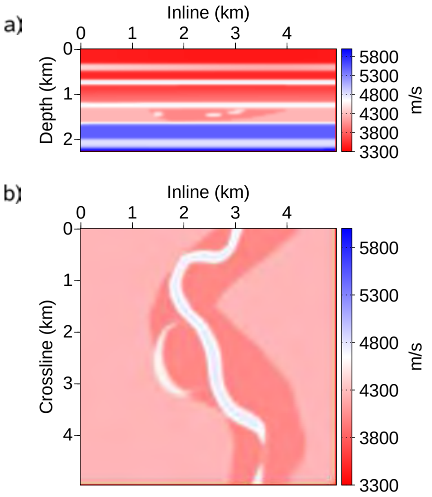

Figure 7.28: Channel velocity model: a) Cross-section at x=2.45 km. b) Horizontal slice at z=1.45 km.

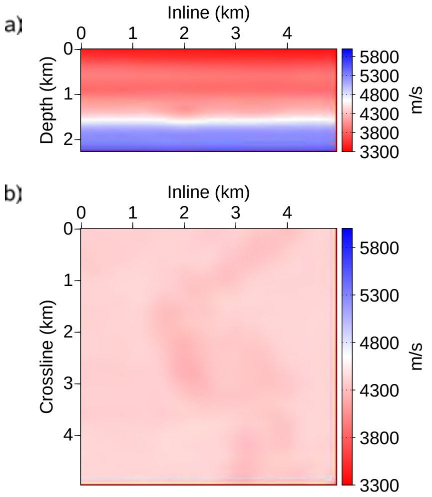

Figure 7.29: Starting channel model for FWI. a) Cross-section at X=2.45 km. b) Horizontal slice at Z=1.45 km.

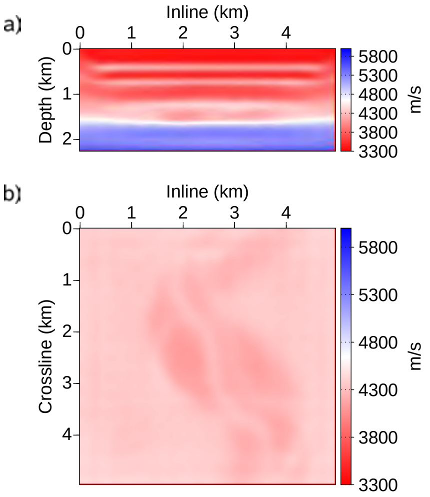

Figure 7.30: Velocity model imaged by FWID in an inverse crime configuration after a successive inversion of the 5 frequencies. a) Cross-section at X=2.45 km. b) Horizontal slice at Z=1.45 km.

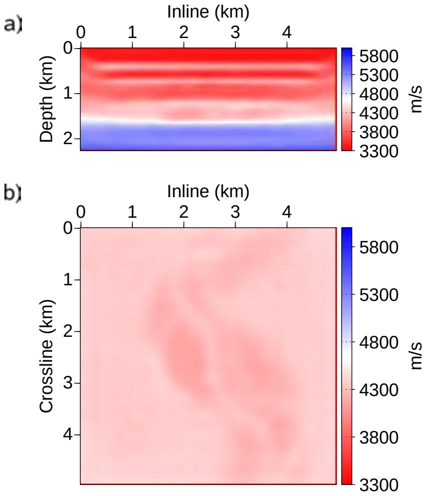

Figure 7.31: Velocity model imaged by FWIH ($\epsilon = 10^{-2}$) in an inverse crime configuration after successive inversion of the 5 frequencies. a) Cross-section at X=2.45 km. b) Horizontal slice at Z=1.45 km.

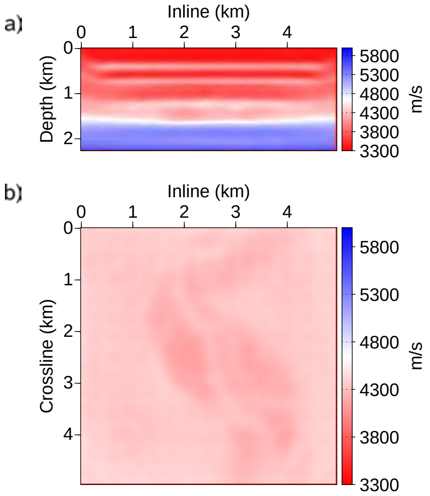

Figure 7.32: Velocity model imaged by FWIH ($\epsilon = 10^{-2}$) from data computed with direct solver after successive inversion of the 5 frequencies. a) Cross-section at X=2.45 km. b) Horizontal slice at Z=1.45 km.

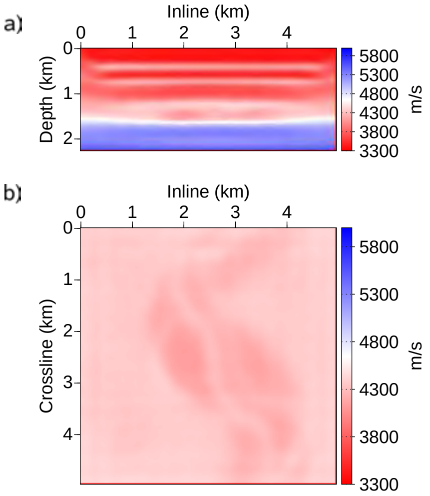

Figure 7.33: Velocity model imaged by FWIH ($\epsilon = 10^{-3}$) in an inverse crime configuration after successive inversion of the 5 frequencies. a) Cross-section at X=2.45 km. b) Horizontal slice at Z=1.45 km.

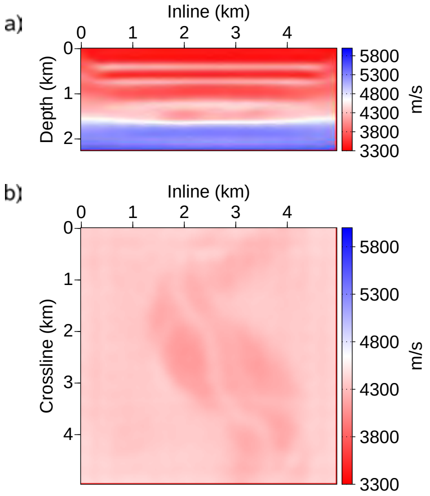

Figure 7.34: Velocity model imaged by FWIH ($\epsilon = 10^{-3}$) from data computed with direct solver after successive inversion of the 5 frequencies. a) Cross-section at X=2.45 km. b) Horizontal slice at Z=1.45 km.

Conclusion

In this chapter, I have validated the FWI software, based either on direct (FWID) or hybrid (FWIH) solver, on homogeneous and complex media. FWID is very efficient on medium size problems thanks to multi-RHS resolution but suffers from LU decomposition time complexity and memory storage requirements. FWIH helps to overcome memory burden but is less efficient with regards to multi-RHS resolution. Hopefully, the convergence criterion is higher for the imaging problem ($\epsilon = 10^{-2}$) than the seismic wave modeling ($\epsilon = 10^{-3}$), which mitigates the computational CPU time.

Part III

Joint Simultaneous-shot & Phase Encoding techniques: an innovative strategy to boost FWI ?

Chapter 8

Simultaneous-shot & Phase Encoding strategy

This chapter is a paper in preparation which will be submitted to the journal "Geophysics".

Because 3-D prestack depth imaging methods such as prestack depth migration and full waveform inversion (FWI) are computationally expensive, the simultaneous-shot technique (Capdeville et al., 2005) can provide an interesting tradeoff between computational efficiency and imaging accuracy. Taking advantage of the linear relationship between the seismic wavefield and the source, the simultaneous-shot technique consists of assembling individual sources to mitigate the number of seismic wave simulations performed during the imaging. The computational cost of one migration or FWI iteration is reduced proportionally to the number of sources gathered in each shot assemblage or super-shot. However, the imaging is altered by artifacts associated with the interferences between the individual sources of a given super-shot, that may require additional iterations of the inversion before convergence towards an acceptable model. These artifacts can be reduced by applying a specific phase code to each source before the assemblage of the super-shot. This technique, called phase encoding (PE) (Morton and Ober, 1998; Jing et al., 2000; Romero et al., 2000), was originally proposed for prestack migration. Romero et al. (2000) have discussed several frequency-dependant and frequency-independent phase encodings and proposed two main approaches for prestack shot-record migration: 1) multiple shots per migration, *versus*, 2) multiple migrations of all the shots. The first approach considers assemblages of phase-coded shots and proceeds migration with these assemblages instead of the single shots. The second approach considers all the phase-coded shots in one assemblage, proceeds many migrations with the single assemblage and stacks all the migration results. For both approaches, phase encoding significantly reduces the computational cost and lead to acceptable images. Liu et al. (2002) have proposed an accurate phase encoding scheme for prestack plane-wave migration and validated their approach on the 2-D Marmousi velocity model.

Since the imaging kernels of prestack migration and frequency domain FWI are basically the same, the technique can be used in FWI algorithms without any particular modification. In fact, the gradient of the objective function, which can be viewed as the imaging kernel of iterative inversion methods, is computed by cross-correlating the shot wavefield with the back-propagated residuals wavefield, such as in migration algorithms based on the wavefield-continuation imaging principle (Claerbout, 1985). Therefore, migration can be performed with one iteration of inversion (Lailly, 1984).

Krebs et al. (2009) have studied the combination of simultaneous-shot and phase encoding techniques on the time domain waveform inversion. They called their strategy Encoded Simultaneous-Source Full Waveform Inversion (ESSFWI). They have demonstrated the efficiency of such imaging strategy compared to the standard approach in waveform inversion on 2D acoustic synthetic data inversions. They have investigated several issues such as the sensitivity of this imaging strategy to the length of the code, starting model and noise. They have concluded that this innovative imaging strategy is robust with regards to noise and initial model and that short codes show greater efficiency which avoids the simulation of longer seismograms than that of the conventional approach which would mean a decrease in the computational efficiency.

We propose to analyze the joint simultaneous-shot and phase encoding strategy for frequency domain FWI. The frequency domain FWI defines hierarchically a multi scale imaging strategy since the iterative inversion process proceeds from the small frequencies to the higher ones to inject higher wavenumbers in the model. The inversion is applied sequentially to few discrete frequencies or groups of frequencies. Due to this hierarchical inversion, different strategies for the application of the phase encoding can be viewed. In the frequency domain in contrast with the time domain FWI, encoding simply consists in applying a phase shift through a complex term multiplication. The encoding may be regenerated without any restriction. Indeed, it may be regenerated at each iteration, for each frequency or group of frequencies and even for each frequency in the group of frequencies. In addition, several types of phase encodings can be compared. The random phase encoding have shown to be the most efficient. As in Romero et al. (2000), we analyze the two strategies of simultaneous-shot, i.e. partial or full source assembling. We assess the performances of the method either when only a limited number of sources are assembled within several super-shots (Ben-Hadj-Ali et al., 2009a) or when all the sources are assembled to form one single super-shot (Ben-Hadj-Ali et al., 2009b). The two approaches show very good efficacy when applied to noise-free data. However, the second strategy is very sensitive to noise.

In this paper, we first readdress the fundamental theory of simultaneous-shot and phase encoding techniques. We present the partial and full strategies of simultaneous-shot approach. We also discuss the importance of the regeneration of the encodings. Then, we quantify the performances of this imaging strategy based on realistic synthetic case studies using the SEG/EAGE overthrust model. Furthermore, we analyze the impact of randomly distributed noise.

8.1 Method

8.1.1 Frequency-domain FWI

FWI is generally recast as an iterative local optimization problem based on the minimization of the least-squares objective function (Tarantola, 1984a; Pratt et al., 1998) given by,

$$C^k(m) = \left(d_o - d_c^k(m)\right)\left(d_o - d_c^k(m)\right)^* \tag{8.1}$$

where m is the model parameter, d_o the recorded/observed data and $d_c^k(m)$ the predicted/computed data at iteration k. The model perturbation, based on the second-order approximation of the objective function and a preconditioned gradient method (Ben-Hadj-Ali et al., 2008), is given by,

$$\delta m^k = -\alpha^k \mathcal{G}_m \left(diagH_a^k + \gamma^k Id\right)^{-1} G^k \tag{8.2}$$

where $diagH_a^k$ is the diagonal of the approximate (Pratt et al., 1998) or pseudo Hessian (Shin et al., 2001) used as a preconditioning of the gradient, $\mathcal{G}_\mathbf{m}$ is a smoothing regularization operator, G^k is the gradient, Id is the identity matrix, α^k is the step length and γ^k is a damping/prewhitening factor that prevents numerical instabilities.

The gradient can be efficiently computed by the adjoint-state method (Plessix, 2006). In the frequency domain, the contribution of one shot to the gradient is given by the product of the incident wavefield emitted by the shot with the backpropagated (i.e., conjugate) residuals wavefield and with a sparse radiation-pattern matrix $\mathbf{W}$, obtained by differentiating the impedance matrix with respect to the model parameter where the gradient is estimated. The gradient is simply formed by summing the contribution of all the shots, that gives,

$$\begin{aligned} G^k &= -\sum_{\textbf{frequencies}} \sum_{\textbf{shots}} P^T \mathbf{W} P_R \\ &= -\sum_{\textbf{frequencies}} \sum_{\textbf{shots}} P^T \mathbf{W} \mathbf{A}^{-1} \left(d_o - d_c^k\right)^* \end{aligned} \tag{8.3}$$

where P is the incident wavefield, P_R the backpropagated residuals wavefield, $\mathbf{A}$ the impedance matrix, the discrete forward modeling operator, $\mathbf{Ap} = \mathbf{s}$, and $\mathbf{W}$ the radiation pattern matrix. ^{T} denotes transpose and * conjugate.

8.1.2 Simultaneous-shot technique

The computational burden in FWI resulting from multi-source simulations can be mitigated by assembling sources in a super-shot. The computational time saving during one FWI iteration is, ideally, proportional to the number of sources assembled into the super-shot (Equation 8.4). However, this shot assemblage introduces artifacts in the gradient and Hessian estimations, that result from cross-talk effects between different sources of a super-shot. Therefore, cross-talks may induce to iterate more and densify frequency sampling in order to improve the wavenumber redundancy and make the inversion process more robust. The expected speed up is defined as

$$\begin{aligned} \textbf{Speed up} &= \frac{\text{Computation time without source assembling}}{\text{Computation time with source assembling}} \\ &= \frac{N_S\ N_S^{its}\ N_S^f}{N_{S-S}\ N_{S-S}^{its}\ N_{S-S}^f}. \end{aligned} \tag{8.4}$$

where N_S, N_S^{its} *and* N_S^f and N_{S-S}, N_{S-S}^{its} *and* N_{S-S}^f are the number of the sources, iterations and frequencies for the single source and simultaneous-shot strategies respectively.

We illustrate our purpose with the gradient G of a super-shot S composed of 2 sources S_1 and S_2. R_1 and R_2 are the backpropagated residual wavefields associated with sources

S_1 and S_2, respectively (Figure 8.1). Note that the same reasoning could apply for the Hessian. The gradient can be schematically written as

$$\begin{aligned} G &= S\mathbf{W}R^* \\ &= [S_1\mathbf{W}R_1^* + S_2\mathbf{W}R_2^*] + [S_1\mathbf{W}R_2^* + S_2\mathbf{W}R_1^*] \end{aligned} \tag{8.5}$$

where $(S = S_1 + S_2;\ R = R_1 + R_2)$ in virtue of the superposition principle.

The first bracketed term corresponds to the standard gradient formed by stacking the contribution of each individual shot while the second term corresponds to cross-talk interferences between sources S_1 and S_2. This extra term alters the imaging result. Minimization of related artifacts is achieved by the so-called phase encoding (PE) technique.

8.1.3 Phase encoding technique for frequency domain FWI

Encoding sources with an arbitrary weight when added into the same super-shot can reduce artifacts coming from interferences. Encoding weights a_1 and a_2 will be expressed with a phase term such that $|a_i| = |\exp(\iota\phi_i)| = 1; i = 1, 2$ and $(S = a_1S_1 + a_2S_2;\ R = a_1R_1 + a_2R_2)$. Equation (8.5) becomes

$$G = [S_1\mathbf{W}R_1^* + S_2\mathbf{W}R_2^*] + [a_1a_2^*S_1\mathbf{W}R_2^* + a_1^*a_2S_2\mathbf{W}R_1^*] \tag{8.6}$$

One may notice that these phase terms act only on the cross-talk terms (Figure 8.1) and the PE strategy can minimize the second bracketed term in equation (8.6) through judicious choice of phases ϕ_i. Several phase encodings have been proposed such as deterministic PE (Jing et al., 2000) and random PE (Morton and Ober, 1998; Romero et al., 2000). Random PE (RPE) generates random phases in the interval $[0, 2\pi]$. Deterministic PE (DPE) assumes that sources in the super-shot are very close and expresses the phase ϕ^k in function of the previous phases $\phi^i; i = 1, k-1$ as given by the following equation,

$$\tan(\phi^k) = -\frac{\sum_{i=1}^{k-1}\cos(\phi^i)}{\sum_{i=1}^{k-1}\sin(\phi^i)} \tag{8.7}$$

An alternative PE (PPE) has been tested as well and consists in taking equidistant phases in the interval $[0, 2\pi]$, i.e. $\phi_i = 2(i-1)\pi/N$ where N equals the number of sources in a super-shot.

In contrast to the time domain FWI where the phase encoding is applied through an addition of time samples to the seismograms, the encoding is performed simply through a multiplication with a complex term $a_i = \exp(\iota\phi_i)$. Therefore, neither the generation nor the application of phase encodings implies significant extra computation cost in the frequency domain. In the hierarchical frequency domain FWI strategy where only few discrete frequencies are inverted, the phases codes may be regenerated for each frequency or group of frequencies and even for each frequency in the group of frequencies and at each iteration of the iterative inversion process. The simultaneous inversion of a group of frequencies have shown to be more robust than the sequential strategy if noise is involved or for elastic FWI (Brossier et al., 2009a).

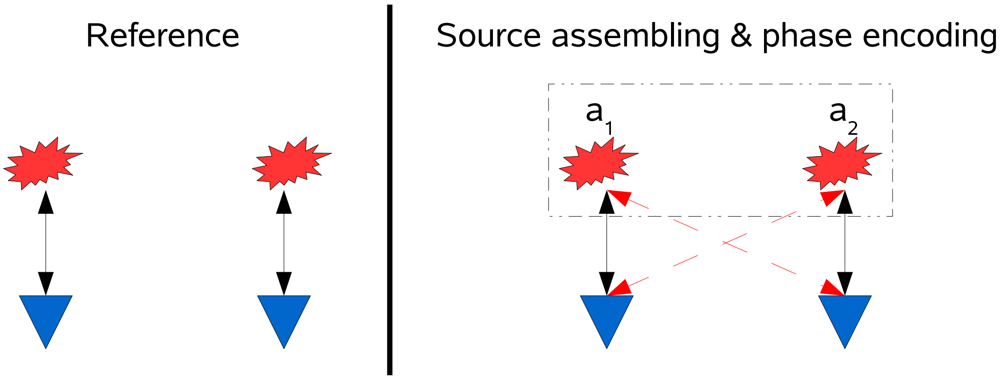

Figure 8.1: Source Assembling & Phase Encoding strategy.

8.2 Numerical Examples

In all the numerical examples, the same code, based on the hybrid solver, is used to either compute observed data (forward problem) or to invert them (inverse problem). Nevertheless, it is not an inverse crime exercise. In fact, data are computed at a GMRES convergence criterion equal to 10^{-3} while they are inverted at 10^{-2} (Sourbier et al., 2008a).

8.2.1 2-D overthrust model case study

In this section, we analyze the behavior of source assembling with/without phase encoding techniques and quantify their performances. Two-dimensional experiments can be designed considering 2.5-D velocity models (laterally invariant in the y-direction) and an infinite line shot in the y direction (Ben-Hadj-Ali et al., 2008) in order to simplify our image analysis although we consider 3-D numerical simulations. We apply 3-D FWI to a dip section extracted from the SEG/EAGE overthrust velocity model (Figure 8.2(a)), discretized on a 801×187 grid with a grid spacing $h = 25$ m. For the 3-D application, the dip section of the overthrust model is duplicated 3 times in the y direction leading to a 3-D $801 \times 3 \times 187$ finite-difference grid. PML absorbing boundary conditions are set on the 4 edges of the 2-D model while periodic conditions are implemented in the y-direction to mimic an infinite medium. The starting model for inversion is obtained by smoothing the true velocity model with a Gaussian function of horizontal and vertical correlation lengths of 500 meters (Figure 8.2(b)). We invert sequentially 7 frequencies ranging from 3.5 to 20 Hz. The sequential approach will be referred to as the sequential single-frequency strategy. For each frequency, we compute 15 iterations. The 2-D acquisition geometry consists of a line of 199 sources and 200 receivers equally-spaced on the surface. The final FWI model obtained without shot assembling is shown in Figure 8.2(c) as reference result.

8.2.1.1 Partial source assembling

First, twenty-five super-shots of 8 sources each are inverted. We build super-shots by gathering either 8 near (cluster super-shot) or distant sources (coarse super-shot) (Figure 8.3). We invert 15 iterations per frequency. The RPE encoding is regenerated at each iteration. Computation time to regenerate codes is negligible compared to other computational tasks.

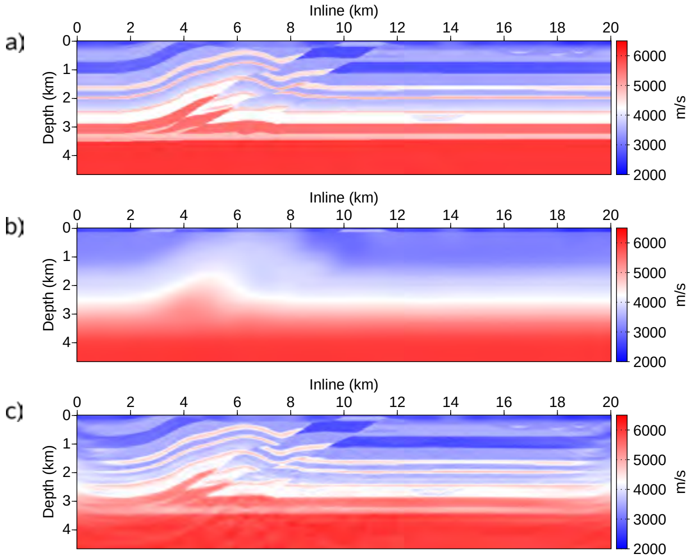

Figure 8.2: 2-D overthrust model application. a) True velocity model. b) Starting velocity model. c) Reference (considering separate sources) FWI velocity model (15 iterations per frequency).

Figure 8.4 shows a comparison between obtained models if codes are regenerated at each iteration or not. The regeneration of the encodings is necessary to significantly reduce cross-talks and lead to good results.

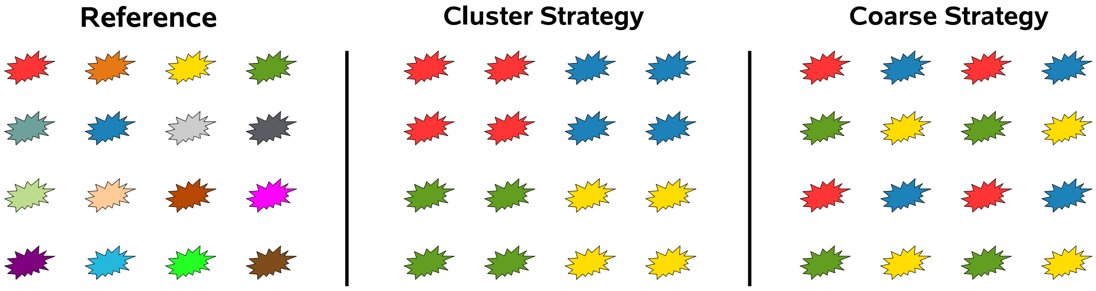

Figure 8.3: Scheme of the Partial Source Assembling strategies: 4 sources per super-shot.

Results obtained with the cluster super-shot approach are outlined in Figure 8.5. The FWI model obtained with shot assembling but without PE is shown in Figure 8.5(a). The footprint (cross-talk artifacts) when considering super-shots is clearly visible especially in the shallower part of the model whose imaging is more sensitive to coarse spacing between sources. The FWI models obtained with the three PE strategies, i.e., RPE, DPE and PPE, are shown in Figures 8.5(b-d) respectively. The three PE techniques succeed to reduce the artifacts in the shallow part of the model. The RPE approach provides the best overall image. This is further confirmed by the L2-norm between the FWI and the true models (Table 8.1).

Results obtained with the coarse super-shot approach are outlined in Figure 8.6. The FWI model obtained with shot assembling but without PE is shown in Figure 8.6(a). The

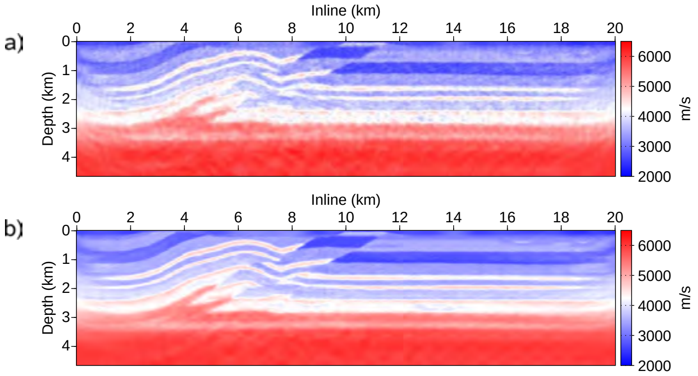

Figure 8.4: 2-D overthrust model application. a) PE changes at each frequency. b) PE changes at each iteration

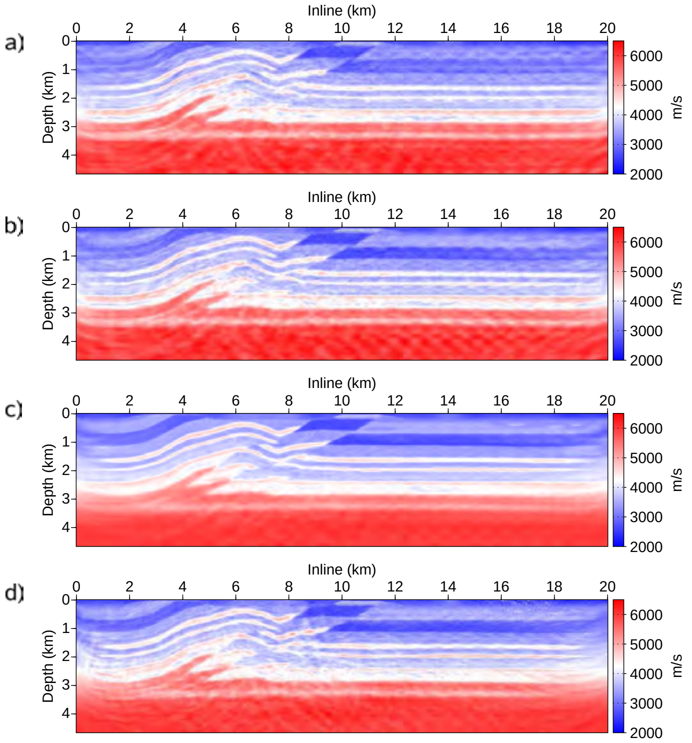

Figure 8.5: 2-D overthrust model application. FWI velocity model with cluster approach (15 iterations per frequency): a) Super-shot no-PE. b) Super-shot & DPE. d) Super-shot & RPE. e) Super-shot & PPE.

footprint (cross-talk artifacts) when considering super-shots is less important than with the cluster approach. The FWI models obtained with the three PE strategies, i.e., RPE, DPE and PPE, are shown in Figures 8.6(b-d) respectively. The three PE techniques succeed to reduce the artifacts in the FWI model. The RPE approach provides the best overall image. DPE gives the worst result since it violates the basic assumption that requires near

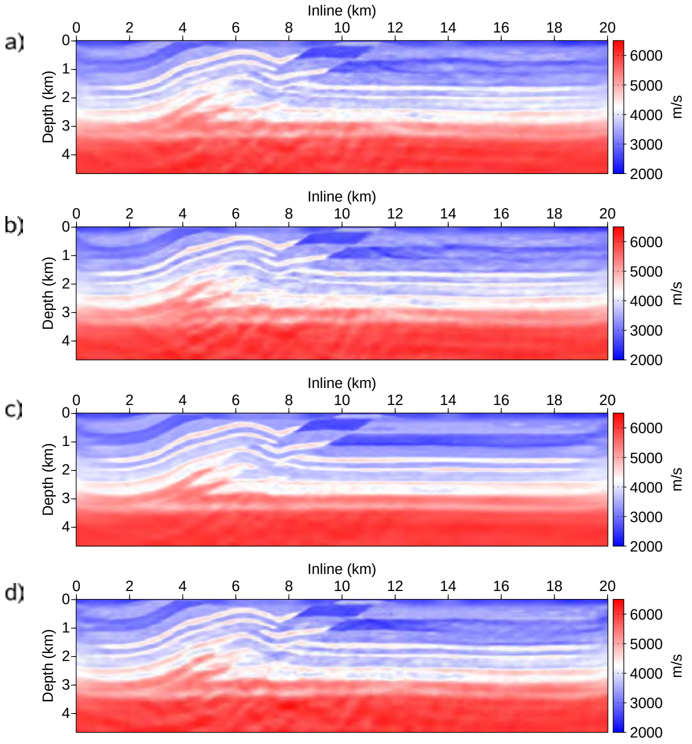

Figure 8.6: 2-D overthrust model application. FWI velocity model with coarse approach (15 iterations per frequency): a) Super-shot no-PE. b) Super-shot & DPE. d) Super-shot & RPE. e) Super-shot & PPE.

sources in a super-shot.

With appropriate PE strategy, i.e. RPE, cluster super-shot approach (Figure 8.5(c)) gives a slightly better result than the coarse approach (Figure 8.6(c)), especially in the shallow part of the model.

Super-shots simulations last nearly 8 times less than the simulation without shot gathering. This is consistent with theory since 8 times less forward problems, which is the most intensive part of the CPU time, have been performed. The results are gathered in Table 8.2.

Table 8.1: Normalized L2-norm residuals (%) between different FWI and true models.

	No-PE	DPE	RPE	PPE
Without shot assembling	3.76	-	-	-
Cluster approach	5.01	4.66	4.47	4.93
Coarse approach	5.99	5.86	4.45	5.75

8.2.1.2 Full source assembling

The same configuration in terms of acquisition and models than that of the previous application is considered. However, all 199 sources are gathered in one super-shot. The final FWI model obtained without PE is shown in Figure 8.7. The number of iterations per frequency inversion was increased to one hundred. The inversion did not converge towards an acceptable velocity model. This failure may be explained by two reasons: first, a super-shot composed with closely-spaced sources is equivalent to a horizontal plane-wave source in virtue of the Huygens principle, that prevents a sufficiently-broad aperture illumination required to obtain well-resolved image of complex structures. Second, the cross-talk terms are not efficiently mitigated if no PE is used.

The FWI model obtained for one super-shot and RPE is shown in Figure 8.8. Two hundreds iterations per frequency inversion were performed. Phase encoding is regenerated at each inversion iteration. The RPE significantly improved the final FWI model, although some artifacts remain in the low velocity layer at 1 km depth in the right hand side of the model. Figure 8.8(a) shows the FWI model after 50 iterations of the 7^{th} (final) frequency and after 200 iterations in 8.8(b). Comparison between the FWI models after 50 and 200 iterations of the 7^{th} frequency shows how the FWI iterations, each of them is performed with a new random phase encoding, help to reduce the cross-talk terms, in addition to the stack of the encoded cross-talk terms performed during each inversion iteration.

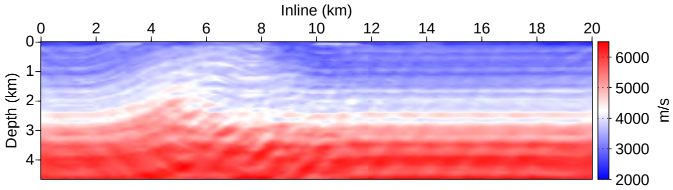

Figure 8.7: 2-D overthrust model application. FWI velocity model with source assembling and without PE (100 iterations per frequency).

Then, we applied FWI to groups of frequencies ([3.54,4.76] , [7.20,9.64] , [13.30,16.97] , 20.63 Hz), where a group defines a group of frequencies that are simultaneously inverted. There is no overlap between successive groups. Only one super-shot with RPE was considered and the random phase encoding was regenerated at each iteration and applied to all the frequencies in the group. The final FWI model is illustrated in (Figure 8.9). Figure 8.9(a) shows the FWI model after 50 iterations of the 7^{th} (final) frequency and after 200 iterations in 8.9(b). The final model is similar to the FWI model obtained without source assembling (Figure 8.2(c)) and closely matches the true model (Figure 8.2(a)).

The comparison between the FWI models of Figures 8.8(b) and 8.9(b) highlights the significant effect of the simultaneous inversion of frequencies. Table 8.2 shows that the model error is very close to that computed for the reference result obtained with single source strategy, i.e without simultaneous-shot technique, and proves the good quality of the obtained model.

In this application, we obtained a speed up of 10, where the speed up is the ratio between the elapsed times required to perform FWI without and with source assembling (Equation 8.4). Ideally, we should have reached a speed up of 15 according to the number of

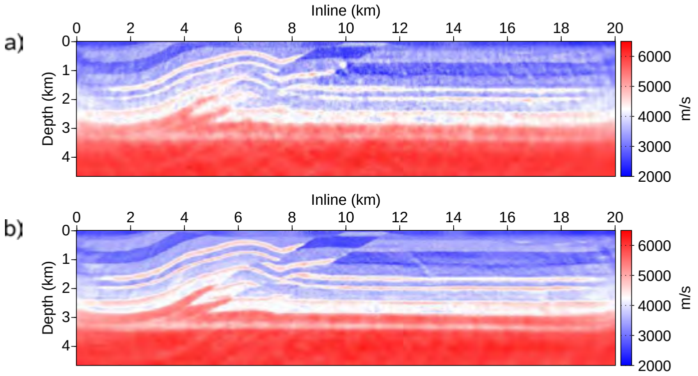

Figure 8.8: 2-D overthrust model application. FWI velocity model with source assembling and RPE for the last-frequency inversion. The seven frequencies were inverted successively. a) 50 iterations per frequency. b) 200 iterations per frequency.

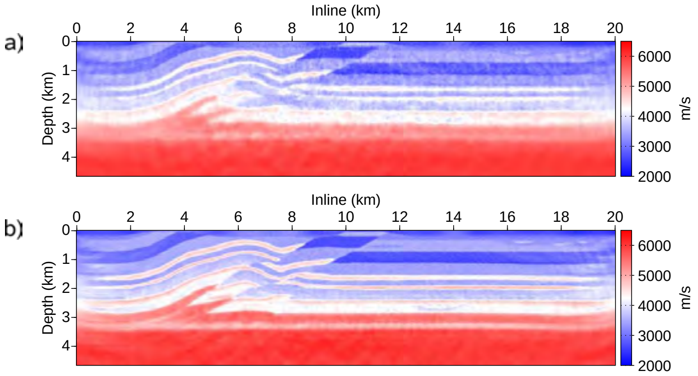

Figure 8.9: 2-D overthrust model application. FWI velocity model with source assembling and RPE for the last-frequency-group inversion. Four frequency groups were inverted. a) 50 iterations per frequency. b) 200 iterations per frequency.

wavefield solutions to be computed (15 iterations and 199 sources *versus* 200 iterations and 1 super-shot) (Table 8.2). However, a significant amount of computations in our frequency domain modeling method based on a hybrid direct-iterative solver (Sourbier et al., 2008a) are independent of the number of shots. A better speed up should be obtained for modeling methods, the computational cost of which linearly increases with the number of shots, such as time domain methods or frequency domain methods based on iterative solver methods.

8.2.2 Noisy data in space

In this section, we introduce noise. The noise is randomly distributed. In the following part, signal-to-noise (S/N) ratio is defined as

$$S/N = \frac{P_S}{P_N} \tag{8.8}$$

where P_S is the signal power and P_N is the noise power. The noise is added separately to each frequency, thus it is a colored random noise since its power depends on the power of frequency signal component. To illustrate the footprint of noise on the data, Figure 8.10 shows monochromatic data with or without noise. Three S/N ratios were respectively considered: 10, 5 and 3.33. The different models obtained without source assembling techniques when four groups of frequencies ([3.54,4.76] , [7.20,9.64] , [13.30,16.97] , 20.63 Hz) are shown in Figure 8.11 and considered as the reference results.

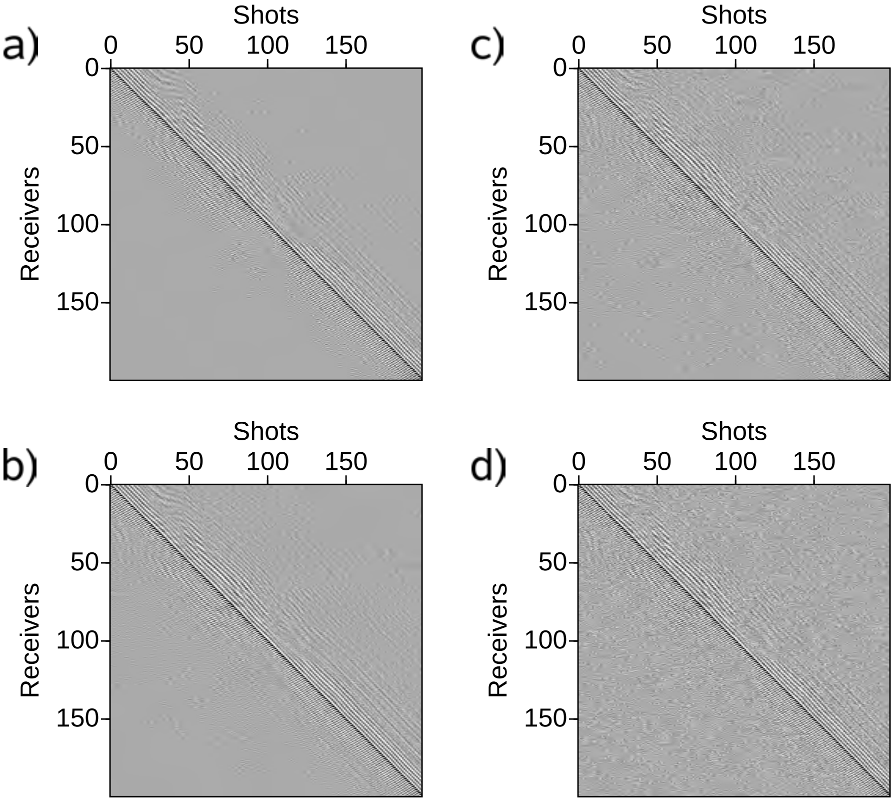

Figure 8.10: 2-D overthrust model application. Frequency data maps. a) without noise. b) S/N ratio=10. c) S/N ratio=5. d) S/N ratio=3.33.

8.2.2.1 Noise impact on full source assembling

The same configuration in terms of acquisition and models than that of the previous application is considered. First, we examine the highest S/N ratio equal to 10. Figure 8.12(a) shows the final inversion result. Compared to the result of the application without noise (Figure 8.9(b)), noise significantly altered the final model. The main geological structures are still well imaged, but channels at about 0.5 km and 2.5 km depth are strongly affected. Since artifacts have mostly high frequency content, we apply the Gaussian smoothing, with horizontal and vertical correlation lengths of 50 m obtained by trial and error, to the gradient. The result is shown in Figure 8.13(a). We notice a slight improvement. The channels at 0.5 km are better imaged.

We perform the same tests with S/N values equals to 5 and 3.33 in order to study the evolution of the results with regards to the noise. At the intermediate S/N ratio equal to 5, the final model is dramatically altered by the noise if gradient smoothing is not applied

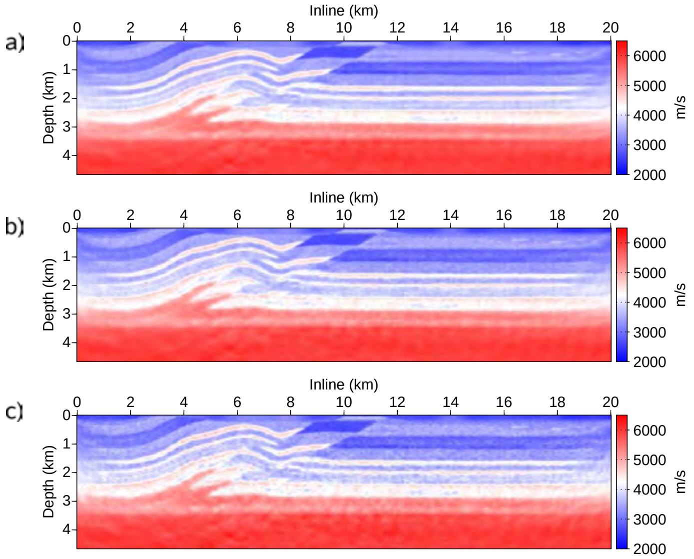

Figure 8.11: 2-D overthrust model application. FWI velocity model without source assembling for the last-frequency-group inversion. Four frequency groups were inverted (15 iterations per frequency). a) S/N ratio=10. b) S/N ratio=5. c) S/N ratio=3.33.

(Figure 8.12(b)) and strong artifacts appear at the subsurface. On the other hand, the result with gradient smoothing can be considered acceptable (Figure 8.13(b)). At the lowest S/N ratio, the obtained model is dramatically altered by noise (Figure 8.12(c)), even if regularization is applied (Figure 8.12(c)). It is clear from these examples that full source assembling strategy is very sensitive to noise. The gradient smoothing is necessary to lead to acceptable inversion results when data are noisy.

To further mitigate the footprint of the noise in the inversion, we now investigate two additional parameters:

- the number of sources in the super-shot: if more sources are added into the super-shot, data are densified and redundancy is strengthened but more cross-talks are introduced;
- the number of inverted frequencies: if frequencies are densified, the redundancy is strengthened.

Impact of the number of sources in the super-shot For this sensitivity study, we consider the intermediate S/N ratio. We compare the reference result obtained with 199 sources in the super-shot (Figure 8.14(b)) with those obtained with either 399 (Figure 8.14(a)) or 99 sources (Figure 8.14(c)). Figure 8.14 shows unambiguously that increasing the number of the sources in the super-shot altered considerably the final model. More sources in the super-shot add more cross-talks. The coupling between these cross-terms and noise can be more important. On the other hand, decreasing the number of the sources has lead to a result slightly worse than the reference one. Based on this study, we need to be aware of the sensitivity in the presence of noise of full source assembling with regards to the number of sources in the super-shot. Many sources in the super-shot may alter results.

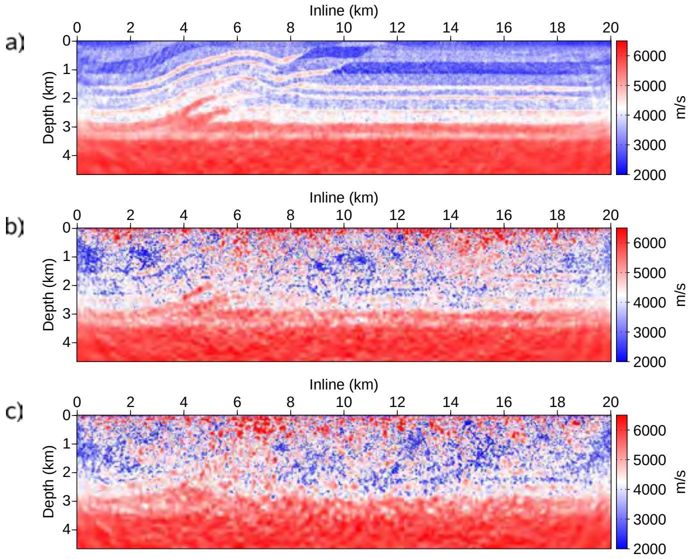

Figure 8.12: 2-D overthrust model application. FWI velocity model with full source assembling and *RPE* for the last-frequency-group inversion. Four frequency groups were inverted (200 iterations per frequency). a) S/N ratio=10. b) S/N ratio=5. c) S/N ratio=3.33.

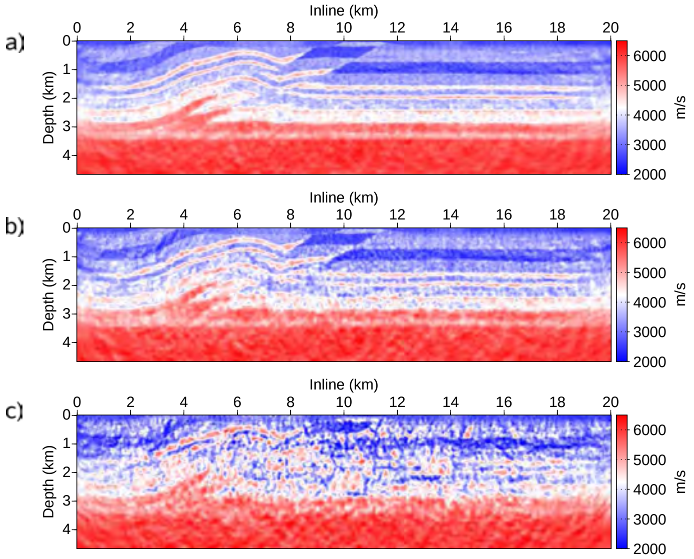

Figure 8.13: 2-D overthrust model application. FWI velocity model with full source assembling and *RPE* for the last-frequency-group inversion. Four frequency groups were inverted (200 iterations per frequency). Smoothing is applied to the gradient. a) S/N ratio=10. b) S/N ratio=5. c) S/N ratio=3.33.

Impact of the number of inverted frequencies We progressively add frequencies in order to assess the behavior of the inversion (Figure 8.15). The number of inverted frequencies ranges from 7 to 29 frequencies distributed in 4 groups. Nevertheless, the

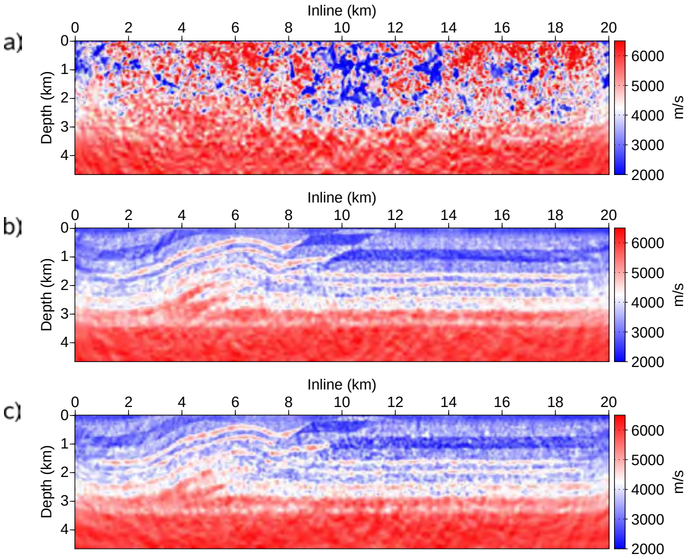

Figure 8.14: 2-D overthrust model application. FWI velocity model with full source assembling and *RPE* for the last-frequency-group inversion. Four frequency groups were inverted (200 iterations per frequency). S/N ratio=5. Smoothing is applied to the gradient. a) 399 sources. a) 199 sources (Figure 10-b)). a) 99 sources.

bandwidths of the different groups of frequencies are approximately the same. The set of simulations proves that the inversion results get better when we inject additional frequencies. The model error shows that the quality of the obtained model is good (Table 8.2).

Nevertheless, the simulations are more computationally demanding since their cost is proportional to the number of inverted frequencies. We have reached a speed up of 2.4 while theoretically we should have got a speed up of 3.6 (Table 8.2).

8.2.2.2 Noise impact on partial source assembling

Once the impact of noise was studied on full source assembling in the previous section, we now assess the behavior of the partial source assembling. The three S/N ratios previously considered are used. We apply FWI to groups of frequencies: [3.54,4.76] , [7.20,9.64] , [13.30,16.97] and 20.63 Hz. We invert 15 iterations for each group of frequencies. The results are shown in Figure 8.16. It clearly illustrates the weak sensitivity of the partial source assembling to noise. These results are compared to those obtained when source assembling technique is not considered, i.e. single sources (Figure 8.11). The obtained models in the two configurations are very close to each other. This highlights the partial source assembling robustness with regards to noise.

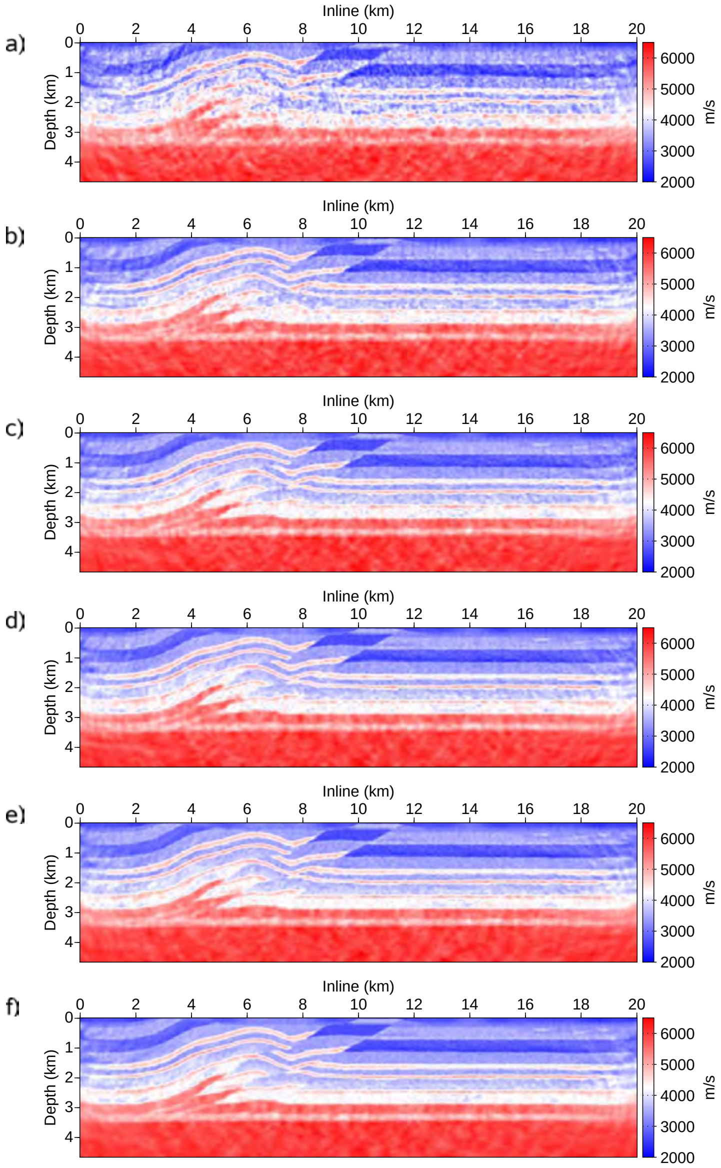

Figure 8.15: 2-D overthrust model application. FWI velocity model with full source assembling and RPE for the last-frequency-group inversion (200 iterations per frequency). Smoothing is applied to the gradient (S/N ratio=5 & 199 sources). a) 7 frequencies. b) 11 frequencies. c) 15 frequencies. d) 18 frequencies. e) 21 frequencies. f) 29 frequencies.

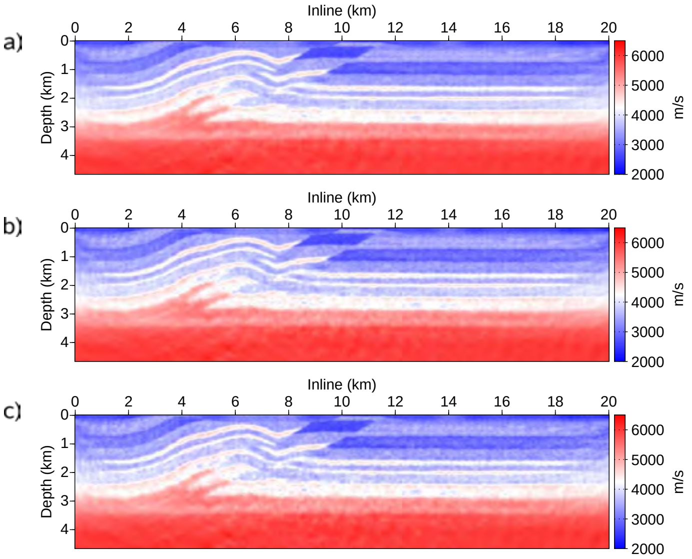

Figure 8.16: 2-D overthrust model application. FWI velocity model with partial source assembling and *RPE* for the last-frequency-group inversion. Four frequency groups were inverted (15 iterations per frequency). a) S/N ratio=10. b) S/N ratio=5. c) S/N ratio=3.33.

Table 8.2: Summary of the different 2-D overthrust model applications.

Figure	S.A	N_S	N_{S-S}	S/N	Model Error %	N_{its}	Speed Up (Theo.)
2-c)	reference	199	199	-	3.76	15	1. (1.)
5	cluster	199	25	-	4.47	15	7.9 (8.)
6	coarse	199	25	-	4.45	15	8.1 (8.)
8	full	199	1	-	4.92	200	10. (15.)
9	full	199	1	-	3.72	200	10.5 (15.)
11-a)	reference	199	199	10	4.88	15	1. (1.)
16-a)	cluster	199	25	10	5.39	15	8.2 (8.)
13-a)	full	199	1	10	4.73	200	10.5 (15.)
11-b)	reference	199	199	5	5.07	15	1. (1.)
16-b)	cluster	199	25	5	5.50	15	8.5 (8.)
14-a)	full	399	1	5	25.1	200	9.2 (15.)
14-b	full	199	1	5	6.31	200	10.2 (15.)
14-c)	full	99	1	5	6.72	200	10. (15.)
15-f)	full	199	1	5	3.82	200	2.4 (3.6)
11-c)	reference	199	199	3.33	5.18	15	1. (1.)
16-c)	cluster	199	25	3.33	5.60	15	8.8 (8.)

S.A – source assembling strategy
N_S – number of sources
N_{S-S} – number of super-shots
S/N – signal-to-noise ratio
Model Error % – model error percentage
N_{its} – number of iterations
Speed Up (Theo.) – evaluated speed up (theoretical speed up)

8.2.3 2-D Marmousi velocity model

The aim of the second 2-D application on Marmousi velocity model is to validate the different conclusions obtained in the previous application and demonstrate that these latter do not depend on the considered model.

The true velocity model is illustrated in Figure 8.17-a). The water zone on the surface has been removed. The computational FD grid is 681 × 3 × 116 and the spacing interval is 25 m. PML absorbing boundary conditions are set on the 4 edges of the 2-D model while periodic conditions are implemented in the y direction to mimic an infinite medium. The initial model (Figure 8.17-b)) for FWI is obtained by 500m-Gaussian smoothing. Either ten single frequencies ot five frequency groups, each group containing two frequencies, were successively inverted. Frequencies range from 3 to 18 Hz. The same S/N ratios as in the previous application have been considered.

The noise-free data FWI final model (considered as reference result) obtained at the last frequency is shown in Figure 8.17-c) while the noisy data FWI models are shown in Figure 8.18.

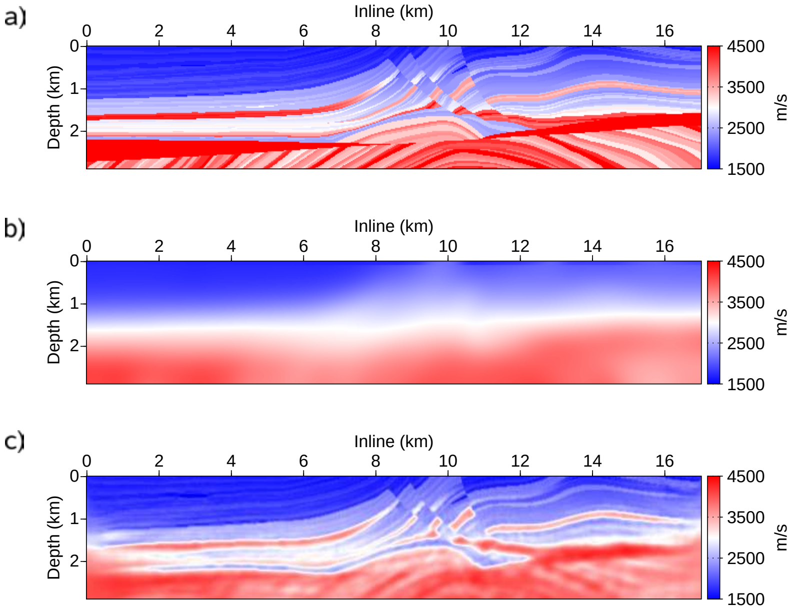

Figure 8.17: 2-D Marmousi model application. a) True velocity model. b) Starting velocity model. c) Reference (considering separate sources) FWI velocity model (15 iterations per frequency).

The joint full simultaneous-shot and random phase encoding leads to a good final result if data are noise free (Figure 8.19-a)). If the data are noisy, the results become not acceptable for the S/N ratios 5 and 3.33 unless a gradient smoothing is applied and frequency groups inversion is adopted (Figure 8.19(b-d) and 8.20(a-c)). If the sources in the single super-shot are densified (in this case doubled), the inversion diverges (Figure 8.21). This proves our conclusion in the previous overthrust application. Too many sources in the super-shot can alter the results.

If partial source assembling with random phase encoding strategy is considered, coarse

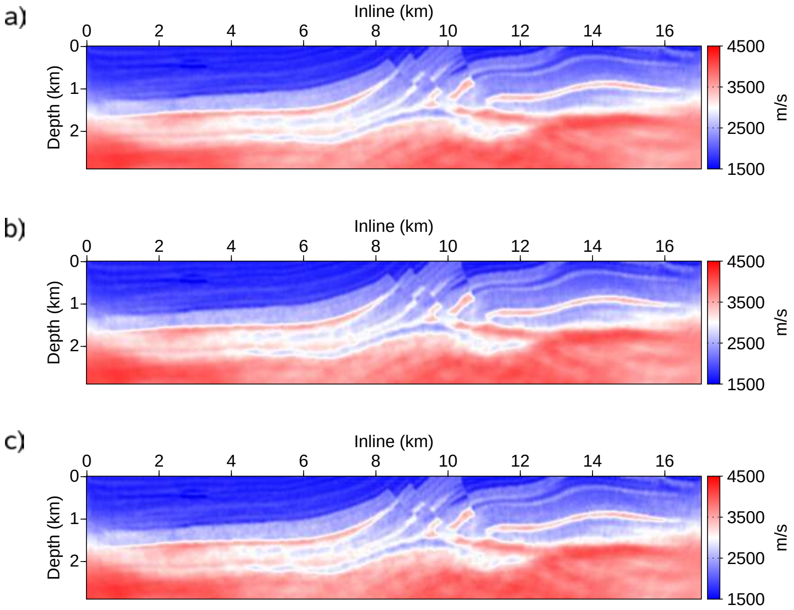

Figure 8.18: 2-D Marmousi model application. FWI velocity model without source assembling for the last frequency group inversion. Five frequency groups were inverted (15 iterations per frequency). a) S/N ratio=10. b) S/N ratio=5. c) S/N ratio=3.33.

or cluster respectively, the results are globally acceptable except near and below the gas charged sand channel at 500 *m* depth and 3 *km* offset (Figure 8.22 and 8.23 respectively). This set of tests also proves that this strategy is less sensitive to noise than full super-shot strategy.

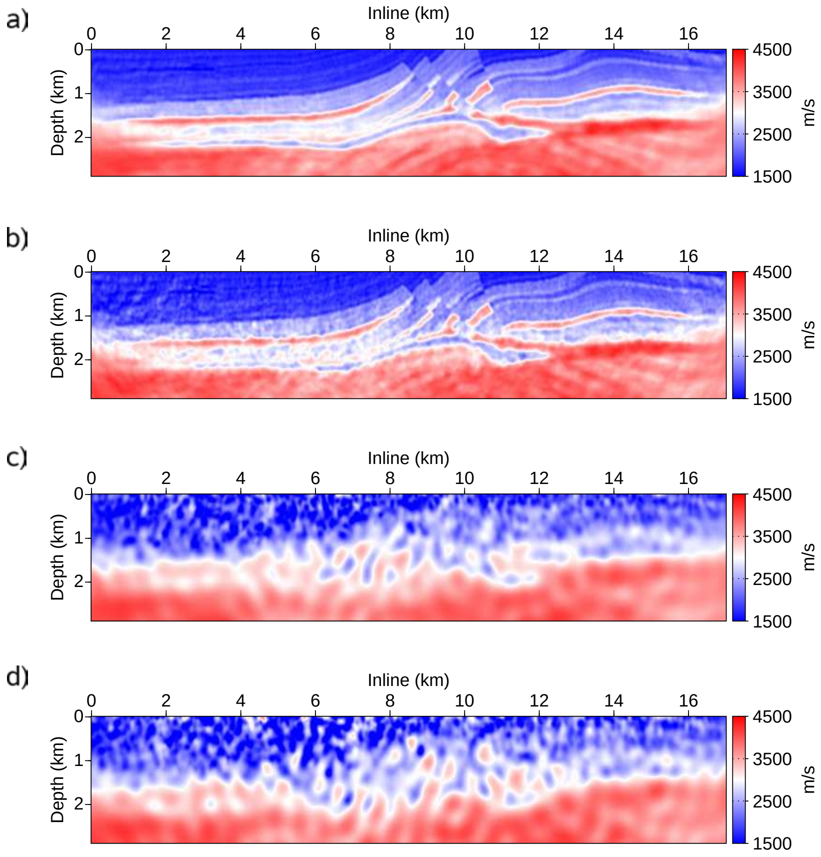

Figure 8.19: 2-D Marmousi model application. FWI velocity model with full source assembling and *RPE* for the last frequency inversion. Ten frequencies were inverted (100 iterations per frequency). Smoothing is applied to the gradient. a) No noise. b) S/N ratio=10. c) S/N ratio=5. d) S/N ratio=3.33.

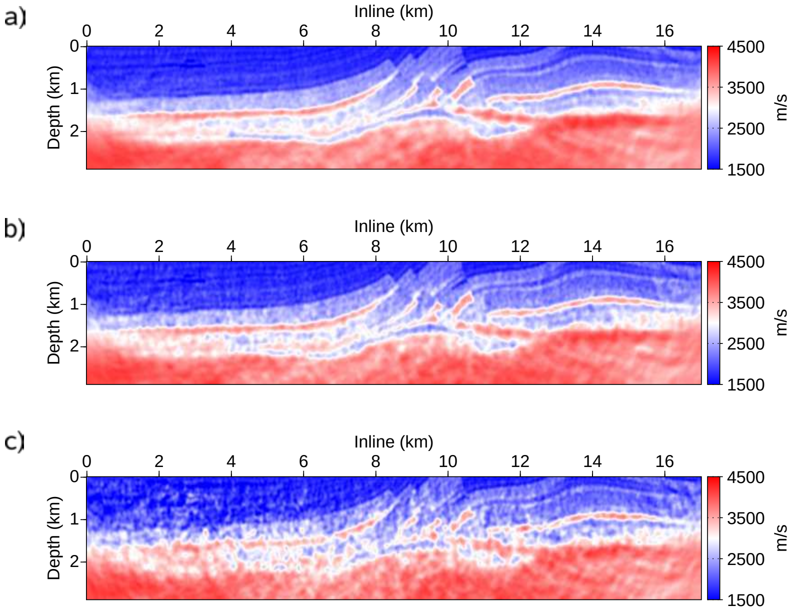

Figure 8.20: 2-D Marmousi model application. FWI velocity model with full source assembling and *RPE* for the last frequency group inversion. Five frequency groups were inverted (100 iterations per frequency). Smoothing is applied to the gradient. a) S/N ratio=10. b) S/N ratio=5. c) S/N ratio=3.33.

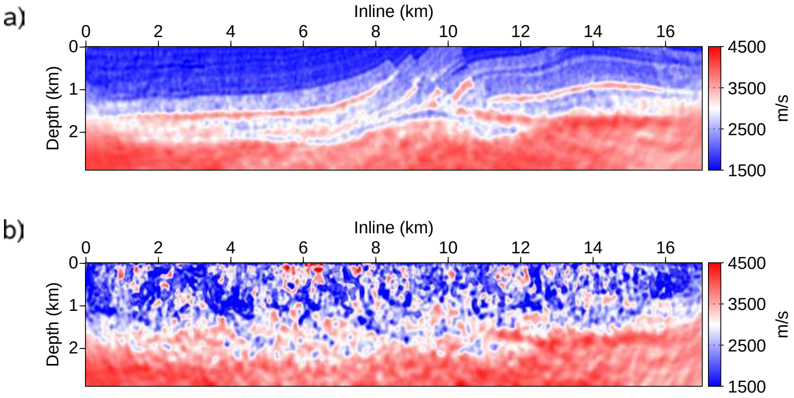

Figure 8.21: 2-D Marmousi model application. FWI velocity model with full source assembling and *RPE* for the last frequency group inversion. Four frequency groups were inverted (100 iterations per frequency). S/N ratio=5. Smoothing is applied to the gradient. a) 169 sources. b) 337 sources.

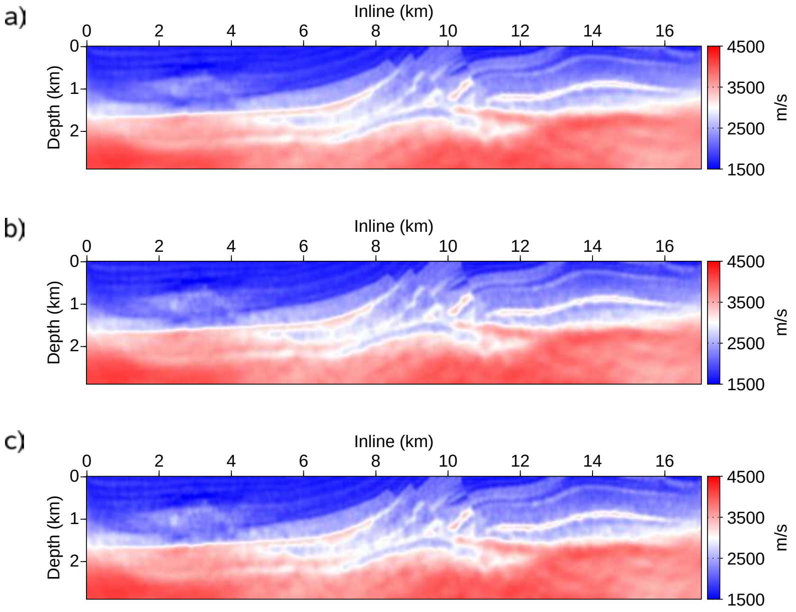

Figure 8.22: 2-D Marmousi model application. FWI velocity model with partial coarse source assembling and *RPE* for the last frequency group inversion. Five frequency groups were inverted (15 iterations per frequency). a) S/N ratio=10. b) S/N ratio=5. c) S/N ratio=3.33.

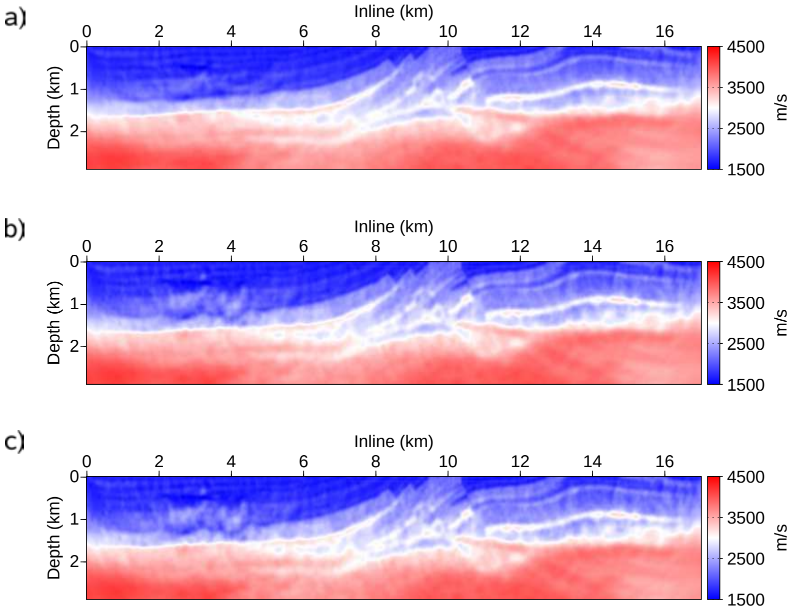

Figure 8.23: 2-D Marmousi model application. FWI velocity model with partial cluster source assembling and *RPE* for the last frequency group inversion. Five frequency groups were inverted (15 iterations per frequency). a) S/N ratio=10. b) S/N ratio=5. c) S/N ratio=3.33.

8.2.4 3-D overthrust model study case

We now consider 3-D applications on the SEG/EAGE overthrust velocity model. The FD grid is $61 \times 265 \times 265$ with grid spacing 75 m and 6 grid nodes PML layers all around. The domain was divided into $3 \times 11 \times 11$ subdomains. The acquisition is composed of 65×65 (4225) sources and 260×260 (67600) receivers. Full source assembling with RPE is used and the 4225 sources form one super-shot. True and initial velocity models are shown in Figures 8.24 and 8.25 respectively. Three single frequencies are inverted: 3.5, 5 and 7 Hz.

In the first application, data are noise free. At each frequency, at least 150 iterations (250 iterations for the first frequency) were necessary since the chosen simultaneous-shot strategy was the full source assembling. Results at 3.5, 5 and 7 Hz are shown respectively in Figure 8.26, 8.27 and 8.28. The three figures show the cross-talks footprint on the obtained models. Nevertheless, these cross-talks attenuate with increasing frequencies. The final result is in agreement with the true model and proves that the full source assembling joint to RPE can lead to acceptable inversion results.

In the second application, we added random noise. S/N ratio is equal to 10. At each frequency, 150 iterations were inverted. Results are in Figure 8.29, 8.30 and 8.31. Figures show coupled effects of noise and cross-talks. Nevertheless, the final model shows most of the geological macro-structures of the true model, especially in the shallow part. Profiles in Figure 8.32 show a comparison between initial, true and final inversion results with and without noise. Comparison shows an acceptable agreement between inversion results and true model, with a better fit for the result without noise.

Even if results are globally acceptable, we note that some strong artifacts (high velocities) appear, particularly at the subsurface (Inline ≈ 10 km). Regularization can be the solution to this type of artifacts.

The objective function history for the two applications are plotted in Figure 8.33 and 8.34. The curve has a common shape that exhibits the decrease of the objective function. Nevertheless, if we zoom on, we see that the objective function varies locally and do not decrease monotonically. This is due to the use of PE that changes the objective function expression at each iteration. This also explains the big jump in the 7-Hz curve of the application without noise (Figure 8.33).

The two 3-D applications have been simulated on the Blue Gene/P IDRIS machine BABEL. The simulation needed 363 cores with 1 *Gbytes* of memory each. The average time for one inversion iteration is about 1500 s (nearly 4 days per frequency).

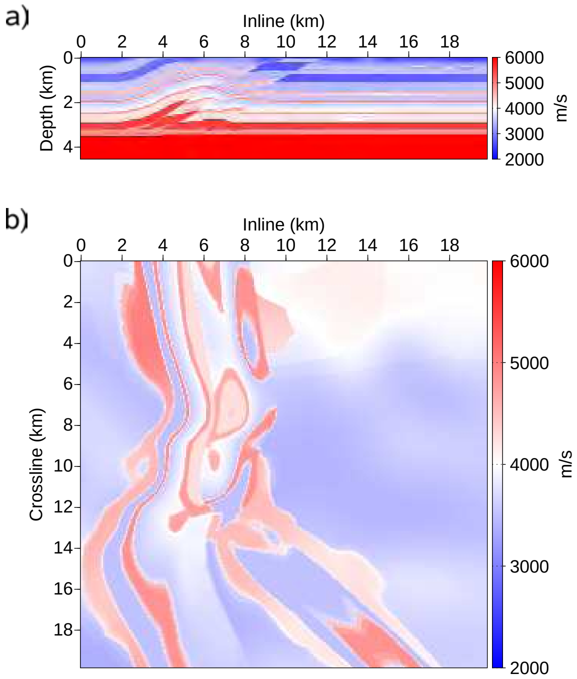

Figure 8.24: 3-D overthrust model application. True velocity model. a) Cross-section at Y=11.4 *km*. b) Horizontal slice at Z=1.5 *km*.

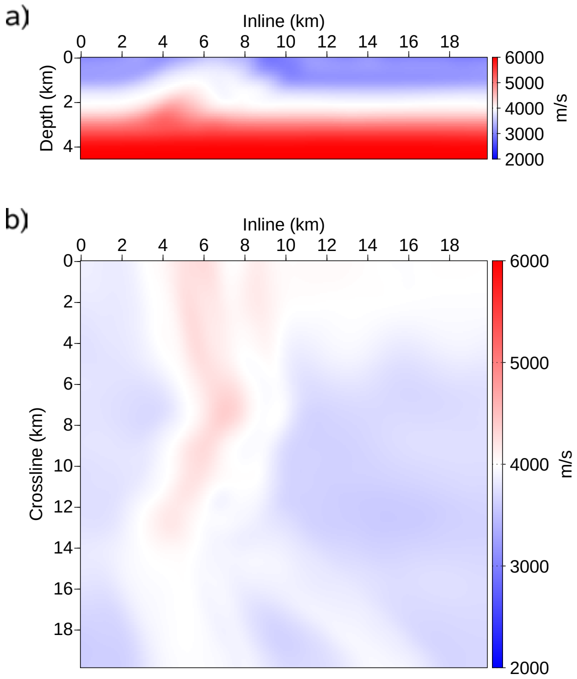

Figure 8.25: 3-D overthrust model application. Initial velocity model. a) Cross-section at Y=11.4 *km*. b) Horizontal slice at Z=1.5 *km*.

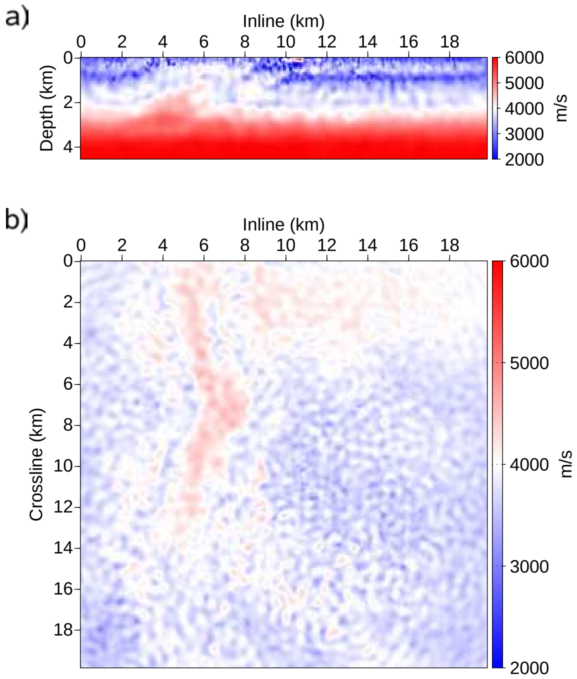

Figure 8.26: 3-D overthrust model application without noise. FWI velocity model at f=3.5 *Hz*. a) Cross-section at Y=11.4 *km*. b) Horizontal slice at Z=1.5 *km*.

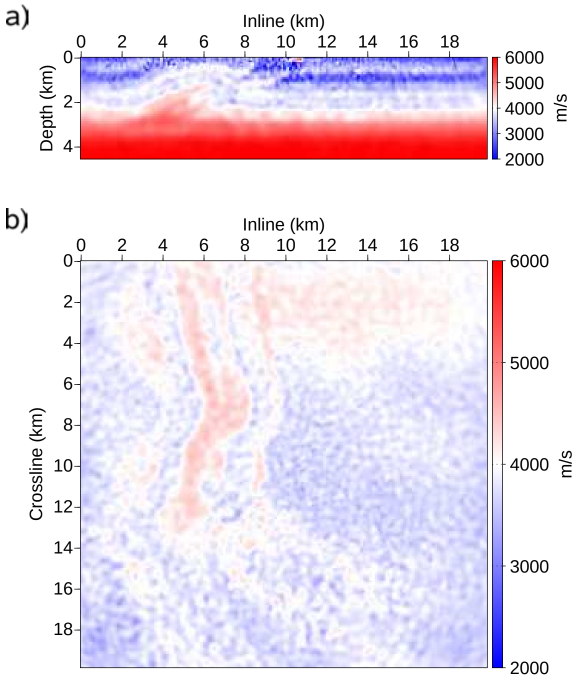

Figure 8.27: 3-D overthrust model application without noise. FWI velocity model at f=5 *Hz*. a) Cross-section at Y=11.4 *km*. b) Horizontal slice at Z=1.5 *km*.

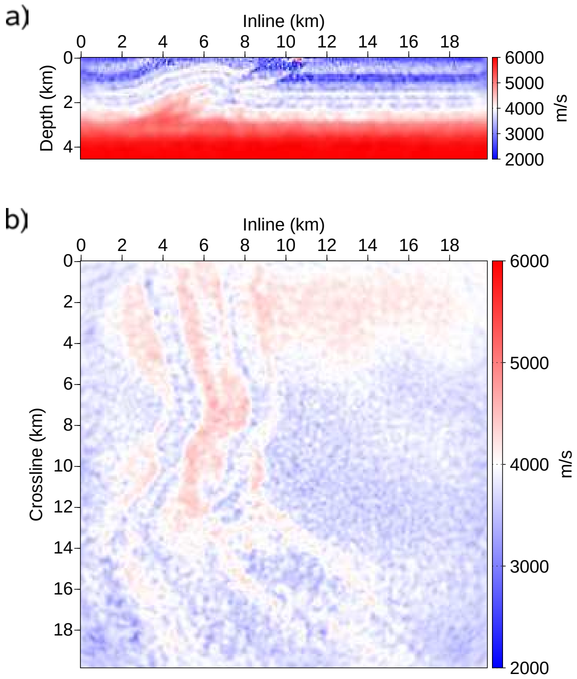

Figure 8.28: 3-D overthrust model application without noise. FWI velocity model at f=7 Hz. a) Cross-section at Y=11.4 km. b) Horizontal slice at Z=1.5 km.

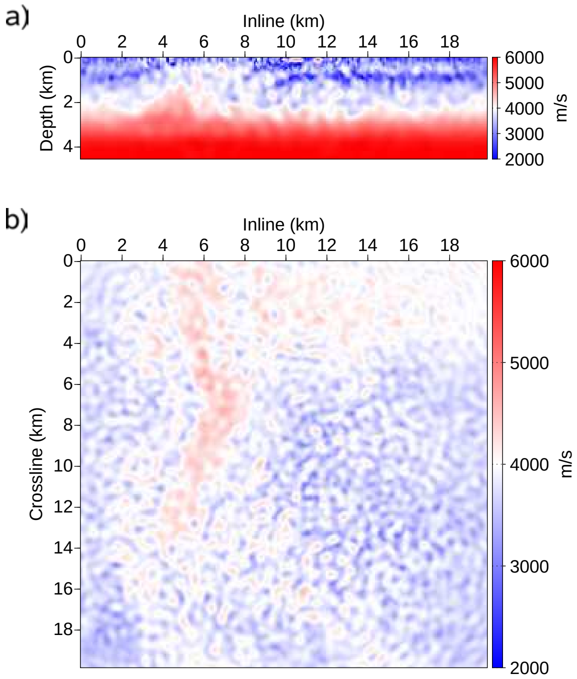

Figure 8.29: 3-D overthrust model application with noise. FWI velocity model at f=3.5 Hz. a) Cross-section at Y=11.4 km. b) Horizontal slice at Z=1.5 km.

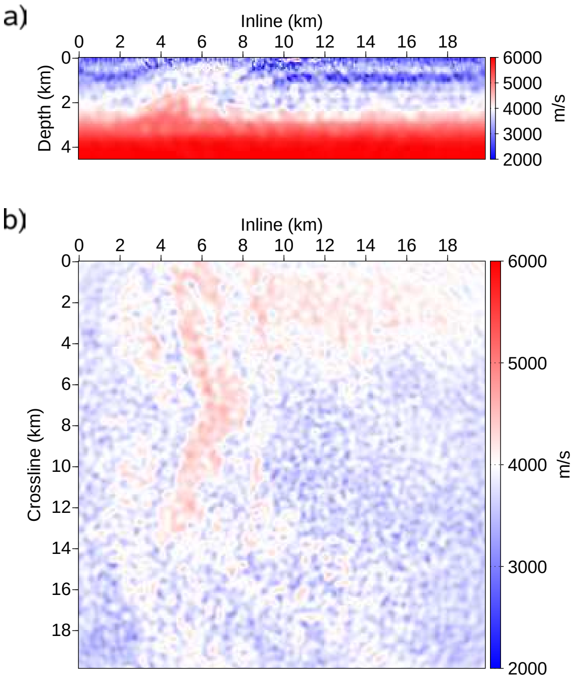

Figure 8.30: 3-D overthrust model application with noise. FWI velocity model at f=5 Hz. a) Cross-section at Y=11.4 km. b) Horizontal slice at Z=1.5 km.

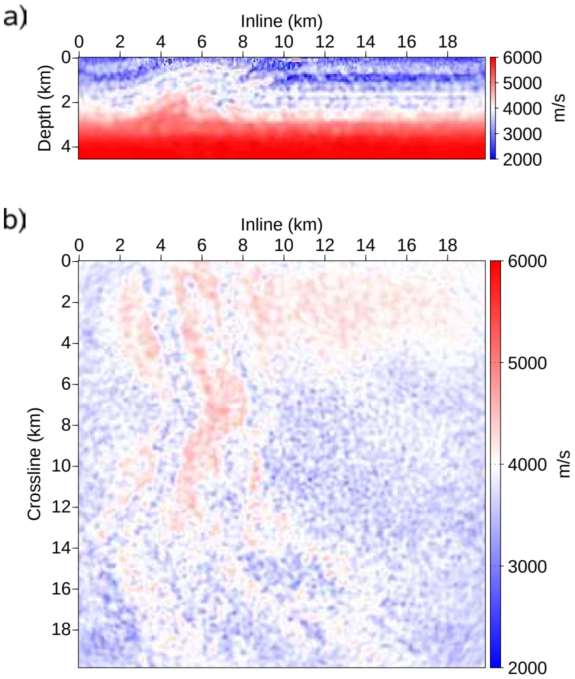

Figure 8.31: 3-D overthrust model application with noise. FWI velocity model at f=7 Hz. a) Cross-section at Y=11.4 km. b) Horizontal slice at Z=1.5 km.

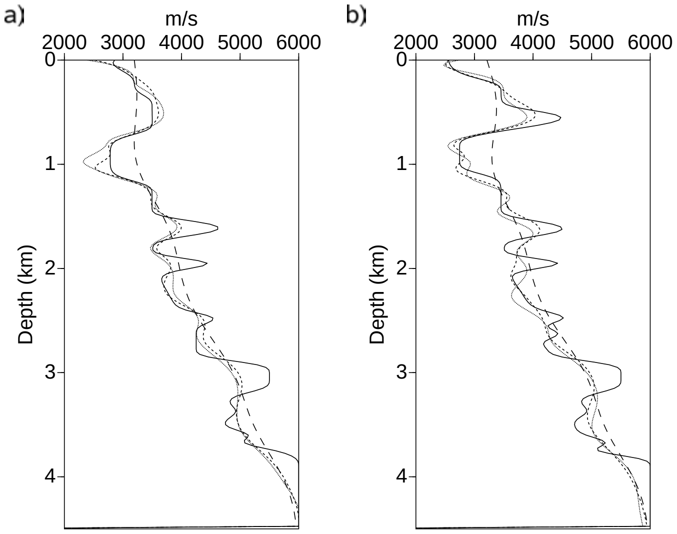

Figure 8.32: 3-D overthrust model application. Horizontal profiles extracted at: a) X=11.25 *km*, Y=7.35 *km*. b) X=7.5 *km*, Y=7.35 *km*. True model is in solid line, starting model in coarse dashed line, free-noise data model in fine dashed line and noisy data model in dotted line.

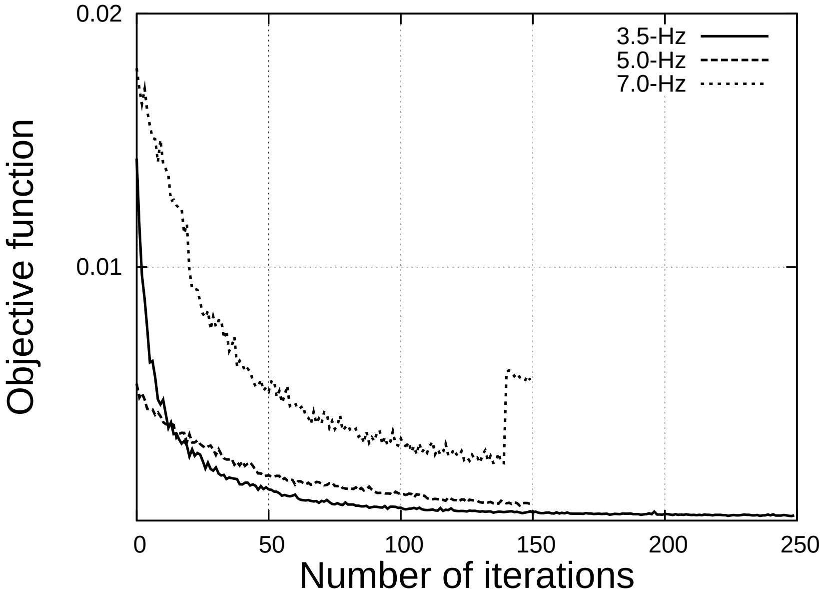

Figure 8.33: 3-D overthrust model application without noise. Objective function history.

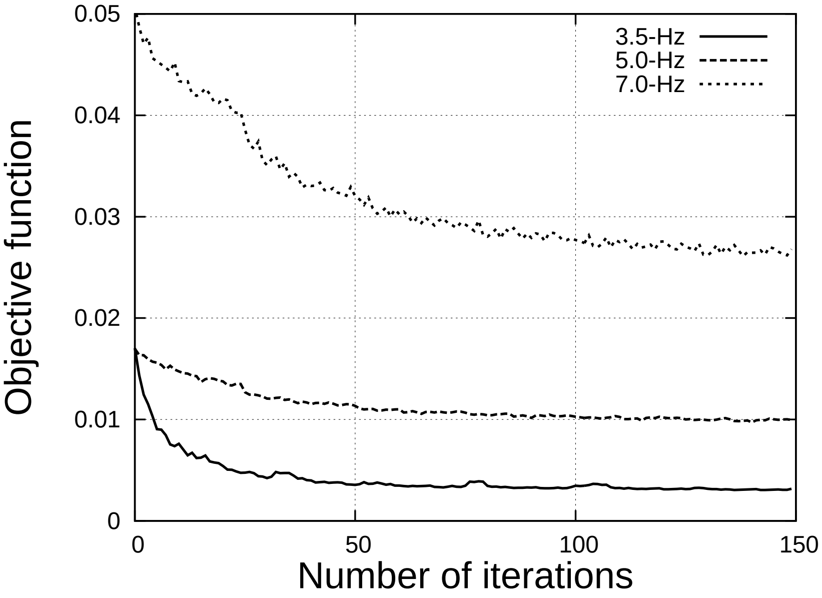

Figure 8.34: 3-D overthrust model application with noise. Objective function history.

8.3 Discussion

We have shown through the 2-D overthrust model application that the joint simultaneous-shot and phase encoding strategy applied to frequency domain FWI is able to produce good results comparing to those obtained with the single source approach and to notably decrease the computational cost of the simulation. We have reached a speed up equal to ten, which means that the simultaneous-shot simulation have lasted ten times less than the reference simulation.

An analysis to compare different phase encodings has been performed and have shown that the random phase encoding lead to the best results (Figure 8.5, 8.6 and 8.9). The analysis have also shown that it is necessary to regenerate the phase codes at each inversion iteration in order to appropriately mitigate the cross-talks introduced by simultaneous-shot approach (Figure 8.4).

Following the previously discussed choice and regeneration of encodings, both the partial and the full source assembling strategy have been efficient. Nevertheless, since all the sources are assembled in one super-shot if the full source assembling strategy is considered, the inversion requires much more iterations in order to mitigate the interference cross-talks. In addition, this strategy requires to use frequency-grouping approach which is more robust than the sequential frequency approach (Figure 8.8 and 8.9).

The noisy data applications suggest that the partial source assembling is robust with regards to noise. The comparison between the obtained results and the reference results obtained with a single source strategy depicts similarity and proves the insensitivity of partial source assembling strategy to noise (Figure 8.11 and 8.16). Nevertheless, some regularization has been necessary to get acceptable results. A Gaussian smoothing was applied to the gradient in order to filter the high frequency model perturbation content (Figure 8.12 and 8.13).

On the other hand, the full source assembling lead to relatively poor results (Figure 8.13). However, results may be improved through densifying the inverted frequencies (Figure 8.15). The speed up decreases to about 2.5 instead of 10 in the case where 29 frequencies instead of 7 are inverted. Therefore, this may favor a time domain waveform inversion if the number of frequencies become too big. In fact, the efficiency of frequency domain waveform inversion is closely related to the inversion of only a small group of discrete frequencies (Sirgue and Pratt, 2004).

We have analyzed the impact of increasing the number of the sources in the super-shot and concluded that this may alter the results (Figure 8.15). This can be explained by the increase of cross-talks generated by simultaneous-shot strategy.

First 3-D results obtained with noise-free data and full source assembling strategy are encouraging. The iteration computational cost is dramatically decreased and lasts few minutes instead of few hours. The reduced iteration computational cost obtained thanks to the full source assembling confers interactivity to FWI since the user can supervise the iterative process. When noise is involved, the results become relatively poor and noisy, as for the 2-D overthrust application.

8.4 Conclusion

Data reduction using super-shots or shot assemblages is an efficient way for a significant reduction of computational cost of 3-D full waveform inversion. Cross-talk noise associated with interference components could be attenuated using phase encoding techniques. Many encodings are relevant and lead to good results. We found that a random phase encoding is the most appropriate one either for the cluster, coarse or full super-shot techniques. Partial source assembling is robust with regards to noise and lead to similar models obtained with reference strategy using single sources. However, full source assembling is very sensitive to noise. It is necessary to apply some regularization and densify the inverted frequencies in order to improve results. Increasing the number of sources in the unique super-shot can alter the noisy data inversion result as shown in the 2-D overthrust study case.

Chapter 9

Conclusions & Perspectives

The objective of the thesis has been to investigate the feasibility of 3-D visco-acoustic frequency domain full waveform inversion and to analyze the expected imaging results that FWI would provide. A 3-D massively parallel FWI software has been developed in the frequency domain. FWI is formulated as a linearized least squares optimization problem which aims to minimize the differences between recorded and predicted data. The full seismic wave propagation modeling is also performed in the frequency domain and relies either on a direct or hybrid direct/iterative solver. The FWI software is written in Fortran90 and the parallel implementation is based on the *Message Passing Interface* (MPI) communication standard.

9.1 Seismic wave propagation modeling

The first part was devoted to the 3-D visco-acoustic wave propagation modeling. First, I was interested in the finite differences discretization of the Helmholtz wave equation. The discretization of the differential operators is performed with second-order-accurate stencils in order to minimize the numerical bandwidth of the sparse impedance matrix. This feature is very important in the prospect of using the direct solver, based on LU decomposition, to solve the discrete time-harmonic wave equation. Nevertheless, second-order finite differences stencils are poorly accurate. To overcome this limitation, a linear combination of different $\mathcal{O}(\Delta x2)$ stencils discretized on several coordinate systems (the so-called mixed-grid strategy) and a mass-term distribution allow to design both compact and accurate stencils. The $\mathcal{O}(\Delta x2)$ stencils are designed on staggered grids for the first-order velocity-stress system, and a parsimonious approach, the aim of which is to eliminate the auxillary velocity wavefields, is finally used to derive the discretized second-order wave equation in pressure (Operto et al., 2007). The differentiation procedure leads to a 27-node stencil which requires four grid nodes per wavelength to reach a good numerical accuracy.

The resulting linear system is solved with either a direct or a hybrid direct/iterative solver, and both solvers are implemented in the FWI software. The direct solver is very efficient for small-scale problems (involving less than 10 millions of unknowns) and for multi-RHS resolution problems such as in FWI where thousands of seismic sources are involved in 3-D. However, the LU decomposition time and memory complexities prevent the use of direct solver for large problems and limit its applicability to small problems of about ten million unknowns. In order to overcome the direct solver limitations, a hybrid (direct-iterative) solver based on the domain decomposition method and the Schur complement has been developed. The hybrid solver allows to reduce the memory requirements of the simulation. Nevertheless, this approach is less efficient with regards to multi-RHS resolution because the time complexity of the iterative component of the hybrid solver linearly increases with the number of RHSs.

The direct and the hybrid solvers have been implemented consistently in the FWI software since both rely on a domain decomposition performed by MUMPS for the direct solver approach and predefined by the user for hybrid solver approach. A similar domain decomposition strategy was implemented in an explicit time marching solver, that will be also implemented in the FWI software as a third modeling engine. The required frequency components can be efficiently retrieved by a discrete Fourier transform (DFT) (Sirgue et al., 2007b). Note, however, that the low memory requirements of time-marching algorithms allows for a coarse-grain parallelism over RHS (i.e., one processor can be assigned to one RHS, and sequential modeling is performed on each processor to perform multi-RHS modeling in parallel). However, if the number of processors significantly exceeds the number of RHSs, then, a second level of parallelism based on domain decomposition can be combined with that over RHSs.

Only $1 - level$ parallelism has been implemented in the hybrid approach as modeling engine in the FWI software. This classical parallel implementation of domain decomposition assigns one subdomain per MPI process. Haidar (2008, chapter 4) has developed an alternative $2 - level$ parallelism, where one subdomain is assigned to a group of MPI processes and each local problem in a subdomain, such as LU factorization, is computed in parallel on the group of MPI processes. The $2 - level$ parallelism has shown to be more efficient than the classical $1 - level$ one, because less subdomains are used leading to more accurate preconditioner. Performances of the $2 - level$ parallelism strategy on the Helmholtz equation can be found in Haidar (2008, chapter 8). In the future, this $2 - level$ parallelism approach can be implemented in the FWI software.

9.2 Inverse problem

In the second part of the manuscript, I was interested in the settlement of the fundamentals of an optimization problem and the presentation of the line search methods. I have introduced the different line search algorithms: Newton, quasi-Newton, steepest descent and conjugate gradient. In instance, FWI is formulated as a least squares minimization problem which tries to minimize the misfit between the recorded and predicted seismic data. In order to solve this inverse problem, the problem is linearized (quadratic approximation of the misfit function) and only small model perturbations handle (Born approximation). The model is updated in an iterative process.

The resolution of the linearized inverse problem is based on a preconditioned gradient method. The gradient is computed through the adjoint-state method and preconditioned by the inverse of the diagonal of either the approximate or pseudo Hessian. The optimal step length is estimated through a parabola fitting.

The conjugate gradient method is also implemented, but was not intensively validated. Conjugate gradient should speed up the convergence. L-BFGS would be implemented in the near future. In this approach, the approximation of the Hessian is more accurate since not only diagonal terms are involved but also extra diagonal terms. The benefits of the

L-BFGS method have been highlighted in Brossier et al. (2009b).

Several synthetic FWI applications on realistic models (SEG/EAGE overthrust velocity model) have been performed. The applications have been restricted to low frequencies ($\approx 7\ Hz$) and the numerical problem size does not exceed six million unknowns. In fact, I have proven the direct solver efficiency (multi-RHS resolution) for such problem sizes. Nevertheless, the high computational cost of the full seismic wave modeling involved in 3-D FWI prevents to deal with large problem sizes.

During the last three years, several synthetic and real data set visco-acoustic frequency domain FWI applications have been presented (Warner et al., 2008; Sirgue et al., 2008, 2009; Plessix and Perkins, 2009). The respective softwares are based on the time domain and iterative solvers while my software is based on the direct and hybrid solvers. The different FWI experiences have highlighted the significant spatial resolution improvements that FWI would provide. However, due to the computational limitations previously mentioned, the applications were limited to low frequencies ($\approx 7\ Hz$). Therefore, these computational limitations question the position of FWI in the imaging procedure flowchart. FWI has been historically proposed to replace the two-step “velocity model building/migration” imaging procedure. In 3-D, FWI is still investigated as an alternative approach to improve the velocity macro-model which would be used for migration.

In the near future, a real OBC data set (Valhall (Kommedal et al., 2004)) will be processed by frequency-domain FWI using the software developed during my PhD. This work will be the topic of a thesis of the SEISCOPE consortium, starting end of 2009.

9.3 Joint simultaneous-shot and phase encoding

As mentioned in the first section, the hybrid solver allows to overcome memory limitations related to the use of the direct solver but is less efficient for the multi-RHS resolution phase. An alternative approach to bypass the problem consists in reducing the number of sources involved in imaging through a simultaneous-shot strategy. The technique consists in assembling several sources in one super-shot and applying the forward problem and imaging condition to super-shots rather than to the individual shots. The simultaneous-shot approach can be applied using two main strategies based on full or partial source assembling. In the former case, all the shots are assembled to form one super-shot, whereas in the latter case, only a limited number of shots are assembled. In partial source assembling, closely-spaced sources (the so-called cluster assembling) or distant sources (the so-called coarse assembling) can be viewed to build super-shots. The computation cost theoretically decreases proportionally to the ratio between the number of shots and super-shots. However, the application of simultaneous-shot technique introduces cross-talks, generated by the interferences between the different sources in each super-shot, which results in artifacts in the final image. Although these artifacts can be mitigated by phase encoding techniques, an increasing number of FWI iterations and an increasing number of inverted frequencies can be required when source assemblage is used to bring down the footprint of the cross-talk artifacts to an acceptable level.

The phase encoding approach has been originally developed for prestack depth migration to mitigate the cross-talk artifacts through encoding the different sources in the super-shot. The phase codes have to be appropriately defined. Among the different choices of encoding, the random phase encoding has shown to be the most efficient for frequency domain FWI.

However, codes should be regenerated at each iteration of the inversion process in order to efficiently mitigate the cross-talk artifacts.

The joint simultaneous-shot and phase encoding strategies have shown to be very efficient to mitigate the computation cost of FWI. This imaging strategy has been validated with 2-D applications involving the SEG/EAGE overthrust and the Marmousi velocity models and with a 3-D application with overthrust model. Both full and partial source assembling strategies have been considered during this validation. Nonetheless, if randomly distributed noise is introduced, the full source assembling have shown to be very sensitive with regards to noise, unless the inverted frequencies are densified, while the partial source assembling remains sufficiently robust.

The joint simultaneous-shot and phase encoding strategy has been investigated for the time domain FWI on 2-D synthetic data (Krebs et al., 2009), but still not on real data set applications. An application on the 3-D Valhall data set can be viewed and results can be compared with that presented in Sirgue et al. (2009).

9.4 Other perspectives

Abubakar et al. (2009) have proposed an alternative FWI approach, called contrast source inversion (CSI). The contrast-source inversion is computationnally efficient in the frame of frequency-domain FWI based on a direct solver, because the LU factorization of the impedance matrix is performed only at the first iteration of one frequency inversion. The LU factors are stored in the core memory and used at each non linear iteration of the inversion process. The governing idea is to define a background model and to compute the scattered wavefield through an iterative process from the contrast source, which depends on the model perturbations, and a forward problem operator that depends only on the background model (i.e., the LU decomposition of the impedance matrix assembled from the background model). Multi-RHS solutions can be therefore efficiently computed during the non linear FWI iterations by substitutions without recomputing the LU decomposition of the impedance matrix at each iteration. The contrast source inversion was validated on 2D synthetic examples by Abubakar et al. (2009). My 3D FWI code could be adapted to implement the contrast source inversion method and test the method for 3D configurations. A natural application of the contrast source inversion is time-lapse inversion where the baseline model can be used as background model.

The second point concerns the FWI least squares formulation. The $\mathcal{L}2$-norm criterion have shown some limitations, in instance with regards to noise. Other criteria, such as the $\mathcal{L}1$-norm, have been proposed in order to bypass some specific limitations (Crase et al., 1990; Shin et al., 2007).

Dealing with real data set applications have raised questions about the relevance of the recorded amplitudes. A phase-only inversion strategy has been promoted. The phase-only inversion can be easily implemented in the FWI software by using the Logarithmic criterion instead of the $\mathcal{L}2$-norm criterion and taking only the imaginary part (Shin and Min, 2006). Alternatively, reconstruction of density and attenuation in addition to that of P-wave velocity can be viewed to model more accurately seismic wave amplitudes. Both heterogeneous density and attenuation can be taken into account in the modeling engine I developed. The reliability of the reconstruction of these two parameters when wide-aperture and wide-azimuth geometries are available should be investigated in the future.

The phase-only inversion can also perform first arrival traveltime tomography (FATT), following the approach described in Shin et al. (2002).

The last point is related to the Laplace domain inversion (Shin and Cha, 2008). This approach has been promoted for initial velocity model building necessary for waveform inversion. The Laplace domain inversion would be easily implemented in the frequency domain software by considering pure imaginary frequencies.

Part IV

Appendices

Appendix A: Ben Hadj Ali et al. (2008)

GEOPHYSICS, VOL. 73, NO. 5 (SEPTEMBER-OCTOBER 2008); P. VE101–VE117, 24 FIGS., 3 TABLES.
10.1190/1.2957948

Velocity model-building by 3D frequency-domain, full-waveform inversion of wide-aperture seismic data

Hafedh Ben-Hadj-Ali[1], Stéphane Operto[1], and Jean Virieux[2]

ABSTRACT

We assessed 3D frequency-domain (FD) acoustic full-waveform inversion (FWI) data as a tool to develop high-resolution velocity models from low-frequency global-offset data. The inverse problem was posed as a classic least-squares optimization problem solved with a steepest-descent method. Inversion was applied to a few discrete frequencies, allowing management of a limited subset of the 3D data volume. The forward problem was solved with a finite-difference frequency-domain method based on a massively parallel direct solver, allowing efficient multiple-shot simulations. The inversion code was fully parallelized for distributed-memory platforms, taking advantage of a domain decomposition of the modeled wavefields performed by the direct solver. After validation on simple synthetic tests, FWI was applied to two targets (channel and thrust system) of the 3D SEG/EAGE overthrust model, corresponding to 3D domains of $7 \times 8.75 \times 2.25$ km and $13.5 \times 13.5 \times 4.65$ km, respectively. The maximum inverted frequencies are 15 and 7 Hz for the two applications. A maximum of 30 dual-core biprocessor nodes with 8 GB of shared memory per node were used for the second target. The main structures were imaged successfully at a resolution scale consistent with the inverted frequencies. Our study confirms the feasibility of 3D frequency-domain FWI of global-offset data on large distributed-memory platforms to develop high-resolution velocity models. These high-velocity models may provide accurate macromodels for wave-equation prestack depth migration.

INTRODUCTION

Three-dimensional quantitative seismic imaging in complex environments (e.g., deep water, thrust belts, subsalt and subbasalt structures) is a primary challenge of seismic exploration for hydrocarbon exploitation. In the depth domain, the imaging flowchart for multichannel seismic reflection data is subdivided into two main steps: velocity macromodel estimation and prestack depth migration (PSDM). These steps typically are performed iteratively until flattening of reflectors in common image gathers (CIGs) is optimized. The human interactions during several tasks related to velocity model-building, such as CIG flattening, layer interpretation, and quality control of picking, makes the PSDM workflow time consuming and potentially subjective. Therefore, any approach that helps to automate and optimize velocity model-building will speed up the output of the final PSDM image.

Estimating the velocity macromodel is critical because it has a strong impact on the accuracy of the migrated images in terms of focusing and positioning in depth of the reflectors. The criteria that the velocity macromodel must verify to provide accurate migrated images are still unclear (for illustrations of the sensitivity of 2D and 3D true-amplitude PSDM to the accuracy of the velocity macromodel, see Operto et al., 2000, 2003). Estimating a reliable velocity macromodel for PSDM from conventional multichannel seismic reflection data is a difficult task — one that becomes even more dramatic in complex environments because of the velocity-depth ambiguity at significant depths (Bickel, 1990; Pon and Lines, 2005).

The most common approaches for building a PSDM velocity model rely on reflection traveltime tomography (e.g., Bishop et al., 1985; Stork, 1992) or migration velocity analysis (e.g., Biondi and Symes, 2004; Sava and Biondi, 2005). Both approaches result in approximations for modeling wave propagation, such as the high-frequency approximation or the one-way approximation of the wave equation. This incomplete modeling of wave propagation, together with the limitations imposed by narrow-aperture acquisition geometries, can prevent imaging of steeply dipping reflectors. For example, Zhang et al. (2006) illustrate that turning waves and multireflected arrivals can improve images of the flank of salt bodies. As a result, research has been dedicated to extending the one-way propagator to exploit these arrivals in PSDM (e.g., Zhang et al., 2007).

In this paper, we investigate frequency-domain (FD) full-waveform inversion (FWI) of wide-aperture data as a tool to build 3D

Manuscript received by the Editor 6 December 2007; revised manuscript received 22 February 2008; published online 1 October 2008.
[1] Université de Nice Sophia-Antipolis, Géosciences Azur, CNRS, Valbonne, France. E-mail: benhadj@geoazur.unice.fr; operto@geoazur.obs-vlfr.fr.
[2] Université Joseph Fourier, Laboratoire de Géophysique Interne et Tectonophysique, CNRS, Grenoble, France. E-mail: jean.virieux@obs.ujf-grenoble.fr.

high-resolution velocity models in complex environments (Pratt, 2004). By a wide-aperture acquisition survey, referred to as *global-offset acquisition*, we mean any acquisition geometry with sufficiently long offset coverage to record diving waves whose refraction depths cover the zone of interest. Wide-aperture arrivals such as diving waves and supercritical reflections are sensitive to the large and intermediate wavelengths of a medium (Pratt and Worthington, 1990; Sirgue and Pratt, 2004), which are difficult to image from multichannel seismic reflection acquisition and limited-bandwidth sources. Moreover, multifold wide-aperture surveys lead to a redundant control of frequency and aperture angle on the wavenumber illumination in the model space. This redundancy can be decimated to design efficient numerical approaches for seismic imaging in the frequency domain (Pratt and Worthington, 1990; Pratt, 1999). Global-offset acquisition surveys can be conducted at sea or on land with a network of stations (see Clarke et al. [2007] for a recent 3D wide-azimuth node survey).

FWI refers to a quantitative imaging method based on a complete solution of the full (two-way) wave equation for the forward problem and on inverse problem theory for the imaging problem (Tarantola, 1987). An improved model is built by minimizing the misfit between the recorded data and the data computed in the model. FWI shares some similarities with generalized diffraction tomography (Wu and Töksoz, 1987; Pratt et al., 1998). The partial-derivative and misfit wavefields can be interpreted as the wavefields emitted by the shots and scattered by secondary sources (virtual sources in Pratt et al. [1998, their equation 16]), triggered at the position of the heterogeneities lacking in the starting model. Zero-lag correlation between the misfit and the partial-derivative wavefields at the receiver locations provides an unscaled image of the missing heterogeneities (the so-called perturbation model) in the opposite direction of the gradient of the least-squares objective function.

The heterogeneities can be represented by a series of closely spaced diffractors. By virtue of the Huygens' principle, an image of the perturbation model is built by summing the elementary images of each diffractor. The gradient of the objective function can be computed more efficiently by zero-lag convolution of the incident wavefields with the backpropagated residual wavefields, thanks to the spatial reciprocity of the Green's function. The zero-lag convolution between the incident wavefields and the backpropagated residuals is similar to the imaging principle of reverse time migration originally proposed by Claerbout (1971) and recast in the framework of inverse problem theory by Lailly (1984) and Tarantola (1984).

There are two main drawbacks of FWIs. First, they are very expensive computationally because of the complete resolution of the wave equation for a large number of sources. Second, they lack robustness as a result of the complexity of the global-offset wavefields and their sensitivity to noise and to the inaccuracies of the starting model. In the 2D case, the FD formulation of FWI applied to global-offset acquisition provides a promising approach to mitigate these difficulties (Pratt, 1999; Sirgue and Pratt, 2004; Brenders and Pratt, 2007a). The extension of this approach to three dimensions is investigated in this paper.

The FD formulation of FWI was developed originally for 2D crosshole acquisition surveys, which involve wide-aperture recording (Song et al., 1995; Pratt, 1999). Because of the wavenumber redundancy provided by multifold wide-aperture geometries, only a few discrete frequencies are required to develop a reliable image of the medium. Some guidelines to define the optimal frequency interval for FWI are given in Sirgue and Pratt (2004). This FD decimation leads to a very compact volume of data to be managed, which may be critical for 3D applications.

The presence of many local minima in the least-squares objective function can prevent convergence of FWI based on local optimization toward the global minimum of the objective function. This originally motivated development of multiscale strategies in the time domain through successive inversions of subdata sets of increasing frequency bandwidth (Bunks et al., 1995). The FD formulation of FWI provides a more natural and flexible framework with which to design a hierarchical multiresolution imaging strategy, helping to manage the inherent nonlinearity of the inverse problem.

The multiscale approach in the frequency domain is generally implemented by successive inversions of single frequencies of increasing value (Pratt and Worthington, 1990; Pratt, 1999). This strategy differs from that of Bunks et al. (1995), which does not take advantage of the redundant control of frequency and aperture on wavenumber coverage. Indeed, in the approach of Bunks et al. (1995), higher frequencies are injected at a given step of the multiscale approach while keeping the frequencies from the previous steps involved in the inversion. Although this approach may be more robust in the sense that redundant information is involved simultaneously in the inversion during the last stages of the multiscale approach, it may also be prohibitively expensive for 3D FWI. When global-offset recording is available, a starting model for FWI typically is built by first-arrival traveltime tomography (for recent applications at subsurface and crustal scales, see Ravaut et al., 2004; Operto et al., 2006; Brenders and Pratt, 2007a, 2007b), which provides large-scale velocity models whose resolution limit is on the order of the first Fresnel zone width (Williamson, 1991; Williamson and Worthington, 1993).

Application of FWI to 2D real data case studies has been limited to frequencies less than 20 Hz (Hicks and Pratt, 2001; Shipp and Singh, 2002; Ravaut et al., 2004; Operto et al., 2006). In three dimensions, the computational cost of the forward problem suggests that it is difficult to handle frequencies greater than 10 Hz for representative problems (Operto et al., 2007). At this scale, the resulting velocity models might be used as macromodels for PSDM. However, the relevance of FWI velocity models as reference models for PSDM requires further demonstration. An illustration with a real data case study is provided in Operto et al. (2004, 2005) in the frame of imaging a thrust belt in the southern Apennines by combined first-arrival traveltime tomography, FWI, and true-amplitude ray-based PSDM.

Full-wave-propagation modeling is a critical issue in FWI methods because it is the most computationally expensive task in the process. In the frequency domain, the forward problem reduces to resolving a large, sparse system of linear equations for each frequency considered. In two dimensions, the few frequencies involved in the inversion can be modeled efficiently for multiple shots using a direct solver (Marfurt, 1984). Since the original work of Pratt and Worthington (1990), optimal finite-difference stencils have been designed for the FD method based on direct solvers (Jo et al., 1996; Stekl and Pratt, 1998; Hustedt et al., 2004). The extension to three dimensions of this modeling approach is addressed in Operto et al. (2007), who show that problems of representative size (e.g., the 3D SEG/EAGE overthrust model) can be addressed at low frequencies ($<$10 Hz) on currently available distributed-memory platforms.

However, it remains unclear which approach (time versus FD based on direct or iterative solvers) is the most efficient for 3D FWI (Nihei and Li, 2007; Plessix, 2007). This may depend on several parameters related to the experimental setup, such as dimensions of the model, frequency bandwidth, number of traces in the acquisition, and acquisition geometry. Three-dimensional FD modeling methods based on iterative solvers are presented by Plessix (2007), Riyanti et al. (2007), and Warner et al. (2007). These approaches demand far less memory and have better scalability than approaches based on direct solvers, but their run time depends linearly on the number of sources, which may be a significant drawback with 3D surveys involving several thousands of shots or receivers. Alternatively, time-domain modeling methods were proposed to extract the frequency response of the wavefields by discrete Fourier transform (Sirgue et al., 2007b) or phase-sensitive detection (Nihei and Li, 2007) and subsequently invert in the frequency domain. The time-domain approach shares with the iterative FD approach similar advantages and drawbacks with respect to memory complexity, scalability, and computational burden for multishot simulations.

A few applications of 3D frequency-domain FWI to synthetic models are presented by Stekl et al. (2007) and Sirgue et al. (2007a). Sirgue et al. (2007a) apply frequency-domain FWI at low frequencies (3–5 Hz) to the 3D SEG/EAGE overthrust model. Their scheme is based on a finite-difference time-domain approach (Sirgue et al., 2007b). They illustrate the footprint of narrow- and wide-azimuth multichannel seismic-reflection acquisitions on the reconstructed velocity models and compare the convergence rate of 2D and 3D FWI. Stekl et al. (2007) apply 3D frequency-domain FWI to a channel model. Their scheme is based on an iterative solver (Warner et al., 2007). To overcome the computational burden of multishot simulations, several shots are triggered simultaneously, following an approach proposed by Capdeville (2005).

Our paper presents a massively parallel algorithm for a distributed-memory platform that performs 3D frequency-domain FWI using FD modeling based on a direct solver. We provide insight into the relevance of 3D frequency-domain FWI for building high-resolution velocity models of isotropic acoustic media and quantify the associated computational requirement thanks to realistic synthetic case studies. Numerical examples focus on surface wide-aperture/wide-azimuth surveys conducted with networks of sources and receivers on the surface. The most representative example, performed in a 12- × 12- × 4.5-km target of the overthrust model, uses a receiver spacing of 300 m, like the order used during a node survey in the Gulf of Mexico (426 m) (Clarke et al., 2007). The source-receiver patch was deployed above the 12- × 12-km area of the target. Therefore, the maximum far-inline and far-crossline offsets are 12 km for receivers located near the ends of the target area and 6 km for receivers located in the middle.

We do not address building a reliable starting model for FWI that is carried out conventionally by first-arrival traveltime tomography when global-offset acquisitions are considered. The maximum offset and the coarsest source and receiver spacings required to build a reliable large-scale model of the FWI target by first-arrival traveltime tomography must be clarified to assess the feasibility of this tomographic approach. We assume that a starting model describing the long wavelengths of the true medium is available. The second issue is to verify that the velocity models inferred from FWI can be used as a macromodel for 3D wave-equation PSDM.

First, we briefly review the theory of FD full-waveform modeling and inversion. Second, we discuss the parallel implementation of frequency-domain FWI for distributed-memory platforms. Third, we present several numerical examples of increasing complexity whose aim is to validate the algorithm, to illustrate the sensitivity of the imaging resolution to the acquisition geometry, and to provide some insight on the computational complexity of the approach for representative studies.

THEORY

Frequency-domain full-waveform modeling and inversion is a well-established method for imaging 2D media. Extension to the 3D case closely follows 2D strategies. Therefore, only a brief review of FD modeling and inversion is given here. The reader is referred to Operto et al. (2007) for the method used in this paper for FD wave-propagation modeling, to Pratt et al. (1998) for theoretical aspects of frequency-domain FWI, and to Pratt (1999) for practical aspects such as waveform-inversion data preprocessing and source estimation.

3D acoustic finite-difference FD modeling

The 3D viscoacoustic wave equation in the frequency domain is given by

$$\begin{aligned}\frac{\omega^2}{\kappa(x,y,z)}P(x,y,z,\omega) &+ \frac{\partial}{\partial x}\left(\frac{1}{\rho(x,y,z)}\frac{\partial P(x,y,z,\omega)}{\partial x}\right)\\ &+ \frac{\partial}{\partial y}\left(\frac{1}{\rho(x,y,z)}\frac{\partial P(x,y,z,\omega)}{\partial y}\right)\\ &+ \frac{\partial}{\partial z}\left(\frac{1}{\rho(x,y,z)}\frac{\partial P(x,y,z,\omega)}{\partial z}\right) = -S(x,y,z,\omega),\end{aligned} \tag{1}$$

where $\rho(x,y,z)$ is density, $\kappa(x,y,z)$ is the complex bulk modulus, ω is frequency, $P(x,y,z,\omega)$ is the pressure field, and $S(x,y,z,\omega)$ is the source. Various attenuation models can be implemented easily in equation 1 using complex velocities in the expression of the bulk modulus (Toksöz and Johnston, 1981). Sponge-like perfectly matched layer (PML) boundary conditions can be implemented easily in the frequency domain to absorb outgoing energy (Berenger, 1994; Operto et al., 2007).

The relationship between the pressure wavefield and the source is linear, so the discrete acoustic wave equation 1 can be recast in a matrix form as

$$\mathbf{Ap} = \mathbf{s}, \tag{2}$$

where the complex-valued impedance matrix $\mathbf{A}$ depends on the frequency and the medium's properties. System equation 1 can be discretized with the so-called parsimonious mixed-grid finite-difference method (Jo et al., 1996; Hustedt et al., 2004; Operto et al., 2007). The mixed-grid discretization, which uses multiple rotated coordinate systems, is complemented by a mass-term distribution (an antilumped mass) that significantly improves the accuracy of the stencil (Marfurt, 1984). The combined use of the mixed coordinate systems and mass distribution allows one to design both accurate and spatially compact stencils.

Dispersion analysis demonstrates that only four grid points per wavelength are needed to obtain accurate simulations in homogeneous media. This discretization rule is optimal for FWI, whose resolution limit is $\lambda/2$, where λ is the wavelength. A compact stencil is critical if a direct method is used to solve the system resulting from discretizing the Helmholtz equation because compact stencils limit the numerical bandwidth of the matrix and hence its fill-in during LU factorization. The use of a direct solver is interesting in the case of multishot simulations as required by tomographic applications because LU factorization is independent of the right-hand side terms in equation 2. To solve system 2, we use the MUMPS massively parallel direct solver, developed for distributed-memory platforms (Amestoy et al., 2007). A detailed complexity analysis of this approach is provided in Operto et al. (2007).

FD full-waveform inversion

The inverse problem is posed as a classic weighted least-squares optimization problem and is solved by the steepest-descent method (Tarantola, 1987). Newton and quasi-Newton (Gauss-Newton) methods were rejected because of the computer cost of calculating either the Hessian or approximate Hessian (Pratt et al., 1998).

The weighted least-squares objective function is given by

$$C(\mathbf{m}) = \Delta\mathbf{d}^{\dagger}\mathbf{W}_d\Delta\mathbf{d}, \qquad (3)$$

where $\Delta\mathbf{d}$ is the misfit (the difference between the observed data and the data computed with model $\mathbf{m}$) and the superscript † indicates the adjoint (transpose conjugate). The value $\mathbf{W}_d$ is a weighting operator applied to the data; it scales the relative contribution of each component of the vector $\Delta\mathbf{d}$ in the inversion. Minimizing the objective function leads to the following solution for the model perturbation $\Delta\mathbf{m}$ after scaling and smoothing the gradient (Pratt et al., 1998; Ravaut et al., 2004; Operto et al., 2006):

$$\Delta m_i = -\alpha(\mathrm{diag}\,\mathbf{H}_a + \varepsilon)^{-1} \times G_{\mathbf{m}}\,\mathrm{Re}\left\{\mathbf{p}^t\left[\frac{\partial\mathbf{A}^t}{\partial m_i}\right]\mathbf{A}^{-1}\mathbf{W}_d\Delta\mathbf{d}^*\right\}, \qquad (4)$$

where $\mathrm{diag}\,\mathbf{H}_a = \mathrm{diag}\,\mathrm{Re}\{\mathbf{J}^t\mathbf{W}_d\mathbf{J}^*\}$ denotes the diagonal elements of the weighted approximate Hessian $\mathbf{H}_a$, $\mathbf{J}$ denotes the sensitivity matrix, and $G_{\mathbf{m}}$ is a smoothing regularization operator.

One element of the sensitivity matrix is given by

$$\mathbf{J}_{k(m,n),i} = \mathbf{p}_m^t\left[\frac{\partial\mathbf{A}^t}{\partial m_i}\right]\mathbf{A}^{-1}\delta_n, \qquad (5)$$

where $k(m,n)$ denotes a source-receiver couple of the acquisition geometry; m and n denote a shot and a receiver position, respectively; and δ_n is an impulse source located at receiver position n.

The diagonal of the approximate Hessian provides a preconditioner of the gradient that properly scales the perturbation model (Shin et al., 2001). The damping parameter ε is used to avoid numerical instabilities (i.e., division by zero). The matrix $G_{\mathbf{m}}$ is implemented in the form of a 3D Gaussian spatial filter whose correlation lengths are adapted to the inverted frequency component (Ravaut et al., 2004). Amplitude gain with offset can be applied to each seismic trace within the operator $\mathbf{W}_d$:

$$w_d(o_{SR}) = |o_{SR}|^g, \qquad (6)$$

where the scalar g controls the amplitude of the gain with respect to the source-receiver offset o_{SR}. Originally, this operator was introduced to mitigate the contribution of the high-amplitude direct water wave when inverting long-offset ocean-bottom-seismic data (Operto et al., 2006). In our algorithm, the gradient scaling could be estimated once per frequency before the first iteration and kept constant over iterations or recomputed at each iteration. The term $(\partial\mathbf{A}/\partial m_i)$ is the radiation pattern of the diffraction by the model parameter m_i.

For P-wave velocity, the pattern is an explosion. In other words, the matrix whose number of rows corresponds to the number of diffractor points in the 3D finite-difference grid has only one nonzero element located on the diagonal of the ith row. This can be checked easily by noting that the P-wave velocity only appears in the coefficient $(\omega^2/\kappa(x,y,z))$ of acoustic wave equation 1. The finite-difference discretization without antilumped mass of this term leads to one nonzero coefficient per row on the diagonal of $\mathbf{A}$. Differentiating these diagonal coefficients with respect to the ith model parameter reduces to one nonzero coefficient on the ith row. The same conclusion would apply to the attenuation embedded in the expression of the complex velocity. On the other hand, a more complex radiation pattern would be observed for density whose expression appears in the stiffness matrix of the discrete wave equation (Forgues and Lambaré, 1997).

The source term in the FWI algorithm can be estimated by solving a linear inverse problem (Pratt, 1999). The inversion code can be applied to vertical geophone or hydrophone data generated by explosive sources. Indeed, vertical geophone data can be processed as pressure data thanks to the reciprocity principle (Operto et al., 2006). The inversion is applied in cascade to several groups of discrete frequencies. All frequencies of one group are inverted simultaneously. The final model obtained close to inversion of one group of frequencies is used as a starting model for the next group of frequencies. For each frequency group, several iterations can be computed.

PARALLEL NUMERICAL IMPLEMENTATION

We use the MUMPS massively parallel direct solver (Amestoy et al., 2006, 2007) based on a multifrontal method (Duff and Reid, 1983) to solve the forward problem (system 2). Before LU decomposition, the matrix coefficients are ordered to minimize dependencies in the graph of the matrix. Using nested dissection ordering, the theoretical memory complexity of the factorization for a 3D finite-difference problem is $O(n^4)$ and the number of floating-point operations is $O(n^6)$, where n is the number of grid points along one dimension of the 3D square finite-difference grid (Ashcraft and Liu, 1998). The source vectors for the resolution phase are provided in sparse format on the host processor. After resolution, the multiple solutions are distributed over processors following a domain decomposition driven by the distribution of the LU factors. This means that each processor stores a spatial subdomain of all the solutions. We take advantage of this distributed in-core storage of the forward-problem solutions (FPS) to solve the inverse problem in parallel.

The central component of the FWI algorithm is computing the gradient of the objective function. This operator is computed basically by a weighted summation of the FPS, namely, the incident and the backpropagated residual wavefields computed in the starting model (equation 4). The weights in the summation account for the radiation pattern of the diffraction tomography reconstruction (the operator $\partial\mathbf{A}/\partial m_i$ in equation 5) and for the data residuals. This weighted summation is computed in parallel straightforwardly by taking advantage of the distribution of the FPS: each processor computes the subdomain of the gradient corresponding to the subdomain of the FPS stored on this processor. At the end of the summation, the distributed gradient is gathered on the master processor with a collective communication.

When only the P-wave velocity parameter is involved in the inversion, the matrix $\partial\mathbf{A}/\partial m_i$ reduces to a scalar located on the ith diagonal. This implies that the gradient at position of m_i depends only on the values of the FPS at this same position. In that case, the parallel computation of the gradient does not require any point-to-point communication, leading to a parallelism efficiency close to one for this task. Also, all FPS remain in core in the algorithm without disk swapping. If not enough memory is available to store in core all FPS values in addition to the LU factors, the FPS, the gradient, and the diagonal Hessian are computed in a sequential loop over partitions of the right-hand-side terms. Each partition loads in core the maximum number of solutions fitting the available memory. The efficiency of the parallel inversion algorithm is controlled mainly by that of the LU factorization. We obtain a maximum speed-up of 13 with MUMPS on our applications (Operto et al., 2007).

The parallel FWI algorithm is summarized in Figure 1. More details on the 2D version of the parallel FWI algorithm can be found in Sourbier et al. (2007).

SYNTHETIC EXAMPLES

In this section, we present several numerical examples of 3D FWI of increasing complexity to validate the algorithm and to give some insight on the computing cost of the approach on realistic cases. All examples were computed on an HP DL 145G2 Beowulf cluster at the SIGAMM computer center, located in the Observatoire de la Côte d'Azur (France). This parallel-distributed computer is a 48-node cluster, each node comprising two dual-core 2.4-GHz processors, giving 19.2 Gflops peak performance per node. The computer has a distributed-memory architecture, wherein each node has 8 GB of RAM. The interconnection network between processors is Infiniband 4X. Data are shared among processors using the MPIHP message-passing library. For the examples presented here, the PML layers spread along five grid points on each side and each direction. Therefore, no free-surface multiples are considered in the examples. These PML grid points are not taken into account in the description of the finite-difference grids.

```
Loop over groups of frequencies [ifreqgroup/Nfreqgroup]

    Loop over iterations [it/nitermax]
    Initialization of gradient, diagonal Hessian and cost function
    Read starting model

        Loop over frequencies in one group [ifreq/nfreq]
        Build impedance matrix
        Parallel factorization with MUMPS

        Diagonal Hessian computation (if it = 1)
            Build Nshot + Nreceiver RHSs on P0
            Parallel multi-RHS resolution with MUMPS
            Compute subdomains of diagonal Hessian on $P_{[i]}, i = 1,Nproc$

        Gradient computation
            Build Nshot RHS for shot positions on P0
            Parallel multi-RHS resolution with MUMPS
            Extract wavefields at receiver positions on $P_{i}, i = 1,Nproc$
            Compute data residuals and partial RMS on $P_{i}, i = 1,Nproc$
            Estimate source
            Build Nshot RHS for residual positions on P0
            Parallel multi-RHS resolution with MUMPS
            Compute subdomains of gradient on $P_{[i]}, i = 1,Nproc$
        End of loop over frequencies

        Reduce objective function on P0
        Centralize the gradient and the diagonal Hessian on P0
        Scale the gradient by the diagonal Hessian on P0

        Compute step length $\alpha$
        Updated the velocity model on P0

    End of loop over iterations

End of loop over groups of frequencies
```

Figure 1. Outline of the FWI algorithm. Parallel tasks are written in gray. RHS — right-hand-side terms (i.e., sources in wave modeling); Nproc — number of MPI processes in the parallel run; nitermax — maximum number of iterations of one frequency-group inversion; P_i — processor i, where P0 is the master processor. An arbitrary number of frequencies N can be inverted simultaneously (set Nfreqgroup $= 1$ and nfreq $= N$) or successively (set Nfreqgroup $= N$ and nfreq $= 1$).

In the following examples, we use the inverse crime, which consists of computing the data with the modeling tool implemented in the FWI code. The whole wavefield, including refractions, turning waves, and reflections, is involved simultaneously in the inversion. All of the inversions were performed with unweighted data, i.e., using $g = 0$ in equation 6.

3D FWI in 2D configuration

In the first step, we validate the 3D FWI algorithm by comparing the results obtained using a 2D FWI code and the 3D code applied in a 2D configuration. Two-dimensional experiments can be designed considering 2.5D velocity models (laterally invariant in the y-direction) and an infinite line source in the y-direction. The infinite line source in the y-direction was implemented on a limited computational domain in the y-direction using periodic boundary conditions on the two faces of the model corresponding to $y = 0$ and $y = y_{\max}$. The periodic boundary conditions are

$$\left[\frac{\partial P}{\partial y}\right]_{y=-h/2,\,y_{\max}+h/2} = 0. \tag{7}$$

They are applied on two virtual ghost faces located outside the computational domain at positions $y = -h/2$ and $y = y_{\max} + (h/2)$, where h stands for the grid interval.

We applied 3D and 2D FWI to a dip section of the overthrust model (Aminzadeh et al., 1997) (Figure 2), discretized on an 801×187 grid with a grid spacing $h = 25$ m. For the 3D application, the dip section of the overthrust model was duplicated three times in the y-direction, leading to a 3D $801 \times 3 \times 187$ finite-difference grid. A 2D wavefield computed in this 2.5D model with the above-men-

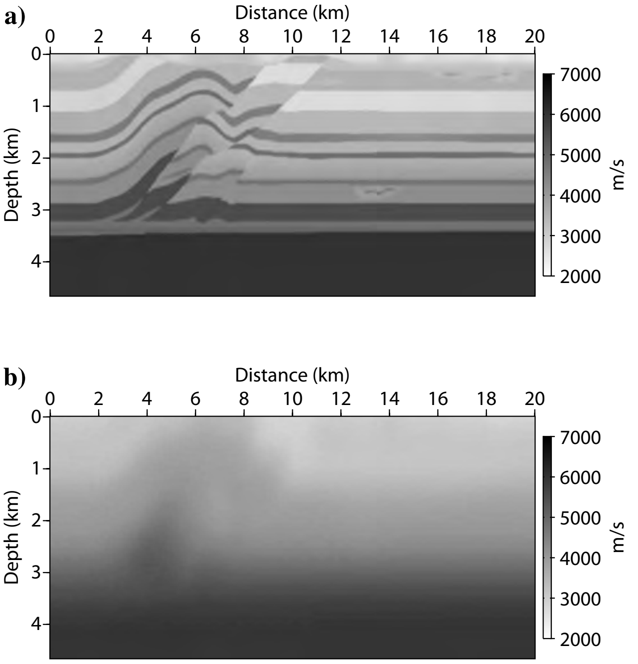

Figure 2. Imaging a dip section of the overthrust model. (a) True velocity model. (b) Starting velocity model.

tioned boundary conditions is shown in Figure 3. PML absorbing boundary conditions are set on the four edges of the 2D model.

The 2D acquisition geometry consists of a line of 200 sources and receivers, equally spaced on the surface. The corresponding 2.5D acquisition geometry consists of duplicating three times the source and receiver lines in the *y*-direction. The true model was augmented with a 250-m-thick layer on top of it; the sources and receivers were deepened accordingly to avoid having sources and receivers just below the PML-model interface. Velocities in this layer are vertically homogeneous and are equal to that on the surface of the original model. We observed strong instabilities during FWI in the near-surface velocities when this layer was not added to the model (Figure 1c in Ravaut et al., 2004). These instabilities can be removed by setting the true velocities in the first 100 m of the starting model without augmenting the model with an artificial layer (Operto et al., 2008).

The starting model for inversion is obtained by smoothing the true velocity model with a Gaussian function of horizontal and vertical correlation lengths of 500 m (Figure 2).

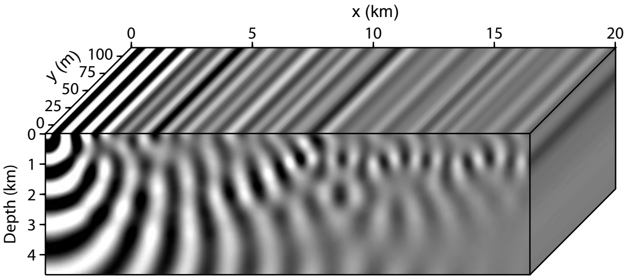

Figure 3. Example of a 2D wavefield computed in a 3D FD grid. Note the limited dimension of the grid in the *y*-dimension. Five grid points are used in the *y*-direction.

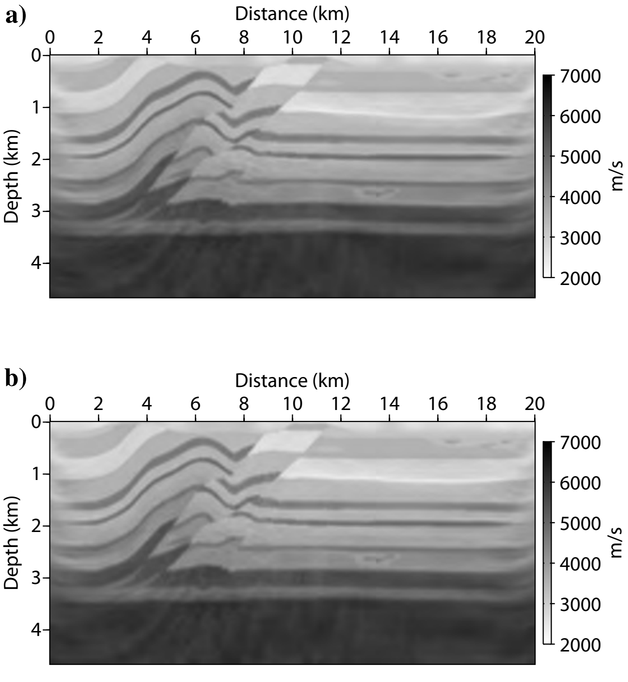

Figure 4. Imaging a dip section of the overthrust model where $f = 20.63$ Hz. (a) Final velocity model from 2D FWI. (b) Final velocity model from 3D FWI.

We inverted sequentially seven frequencies ranging from 5 to 20 Hz. For each frequency, we computed 40 iterations. The final velocity models inferred from 2D and 3D FWI are shown in Figure 4. Some vertical profiles extracted from these models are compared in Figure 5. They are very similar, providing a first validation of the 3D FWI algorithm. The agreement between the final FWI models and the true model is also quite good, although some discrepancies exist between the true and reconstructed velocities around a low-velocity layer located from 0.7 to 1 km deep (Figure 5b). This discrepancy is not observed when the first 100 m of the true model are set in the starting model (Operto et al., 2008). Some high-amplitude perturbations are still slightly underestimated, mainly because of an insufficient number of iterations. (The profiles in Figure 5 can be compared with those obtained from 2D FWI using 80 iterations per frequency in Operto et al. [2008].)

These results also give some insight on the high spatial resolution that can be achieved in the velocity models at relatively low frequencies (i.e., <15 Hz) by FWI of global-offset data thanks to continuously sampling the wavenumber spectrum up to a maximum wavenumber of $2/\lambda_{15\ \mathrm{Hz}}\ \mathrm{m}^{-1}$. (We define the wavenumber as the inverse of the wavelength λ. The value $\lambda_{15\ \mathrm{Hz}}$ is the wavelength corresponding to a frequency of 15 Hz.)

Inclusion models

In this section, we apply 3D FWI for simple velocity models composed of a homogeneous background with one and two inclusions. The models are discretized on a small $31 \times 31 \times 31$ grid with 50-m cubic cells. The velocity in the background medium is 4000 m/s. The inverted frequencies are 3.72, 6.07, 10.00, and 16.27 Hz for the one- and two-inclusion models. To select the inverted frequencies, we chose to remove the wavenumber redundancy in the model space (Sirgue and Pratt, 2004). The frequencies were inferred from the relationship between wavenumber, frequency, and aperture angle provided by the theory of diffraction tomography (Wu and Töksoz, 1987). The linear relation between wavenumber and frequency allows us to increase the frequency interval when the frequency increases (Sirgue and Pratt, 2004). One hundred sources (10×10) and

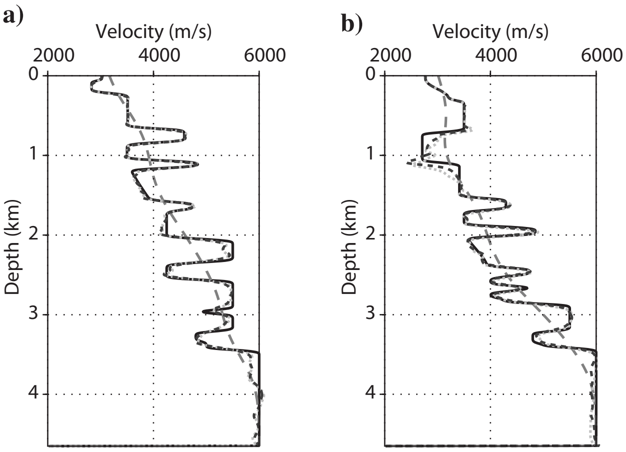

Figure 5. Imaging a dip section of the overthrust model: Comparison between vertical profiles extracted from the true (black solid line), the starting (gray dashed line), and the 2D and 3D FWI models (light and dark gray dotted lines, respectively). The two profile series are located at (a) 4.5 and (b) 13.5 km distance. The 2D and 3D FWI profiles are almost identical.

36 receivers (6 × 6) were distributed uniformly on the top and bottom of the 3D model, respectively. Source and receiver spacings were 150 and 250 m, respectively.

We first consider a velocity model with one inclusion in the homogeneous background. The velocity in the inclusion is 3500 m/s (Figure 6). The inclusion is centered on the 3D grid. The four frequencies were inverted successively. Some horizontal and vertical sections of the inclusion are shown in Figure 6. Note the vertically elongated shape of the inclusion in the vertical cross section and the symmetric shape of the inclusion in the horizontal slice. The vertical elongation results because the top and bottom of the inclusion are sampled mainly by downgoing transmitted wavepaths (i.e., forward-scattered wavepaths) that have limited resolution power, but the shape of the inclusion in a horizontal plane is mainly controlled by reflections (i.e., backward-scattered wavepaths) associated with shots and receivers located near the same face of the 3D model.

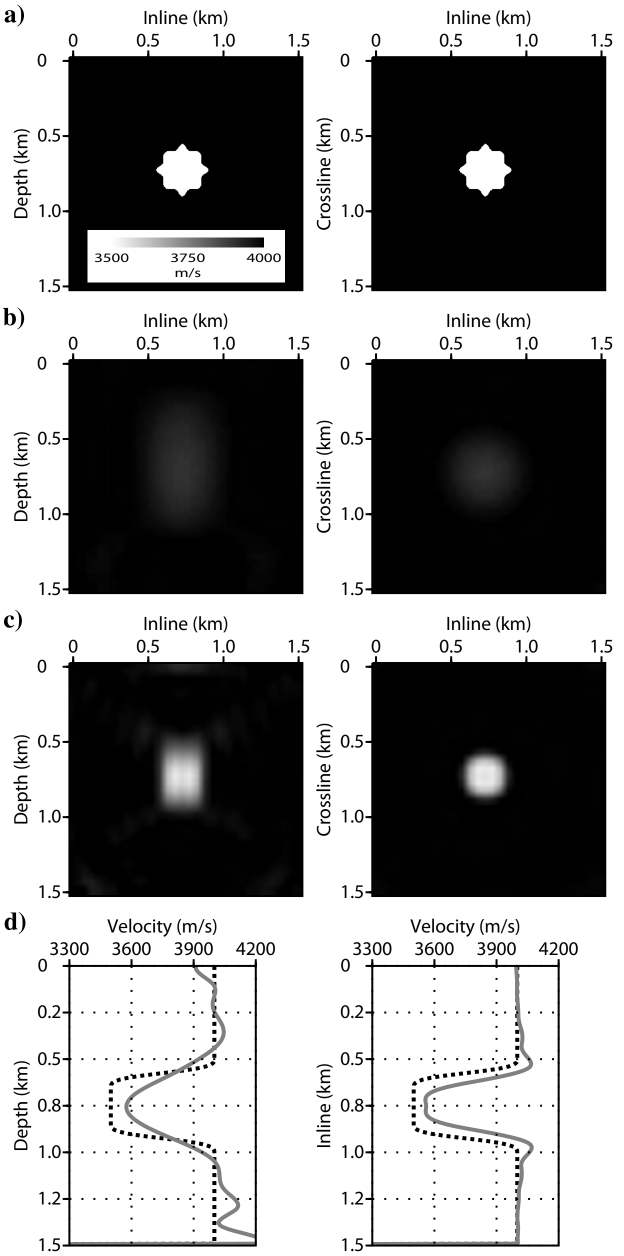

Figure 6. Imaging one inclusion by 3D FWI. Vertical (left) and horizontal (right) sections of (a) the true inclusion, (b) the inclusion after inverting the 3.72-Hz frequency, and (c) the inclusion after inverting the four frequencies successively. (d) Vertical (left) and horizontal (right) profiles extracted from models shown in (a) (dotted lines) and (c) (solid lines).

This relationship between aperture illumination and image resolution is also illustrated on the two profiles extracted from a vertical and horizontal section running through the inclusion. The vertical profile exhibits a clear deficit of high wavenumbers as a result of transmission-like reconstruction, but the horizontal profile exhibits only a slight deficit of small wavenumbers as a result of reflection-like reconstruction. The symmetry of the image of the inclusion in the horizontal plane, which results from the symmetry of the inclusion with respect to the acquisition geometry, is additional validation of the 3D FWI algorithm.

The second example contains two spherical inclusions (3500 and 4500 m/s) corresponding to positive and negative perturbations in the homogeneous background (Figure 7a). The center of the inclusions lies on the same vertical plane in the middle of the grid. The

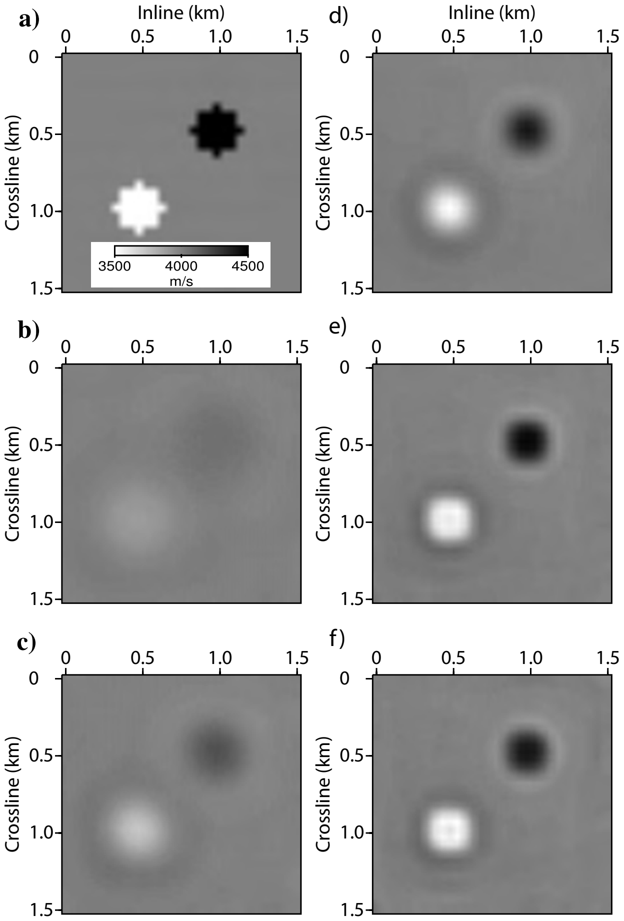

Figure 7. Imaging two inclusions by 3D FWI. (a) True model. Also included are vertical cross sections of the FWI velocity models after successive inversion of frequencies of (b) 3.72, (c) 6.07, (d) 10.00, and (e) 16.27 Hz. (f) Vertical cross section of the 3D FWI velocity model after simultaneous inversion of the four frequencies: 3.72, 6.07, 10.00, and 16.27 Hz.

goal of this test is to verify that the 3D inversion properly handles multiple scattering occurring between the two inclusions. For this case study, the four frequencies were inverted successively and simultaneously (Figure 7b-f). In both cases, the inversion successfully imaged the two inclusions.

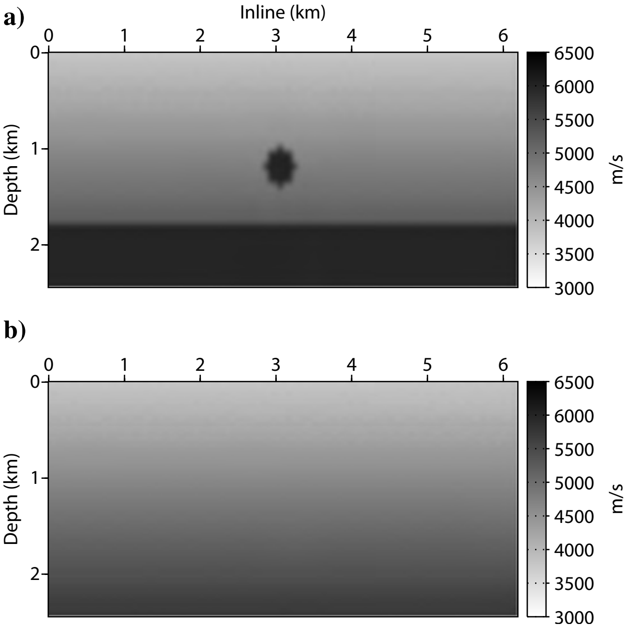

Figure 8. Imaging the inclusion/interface model. (a) True velocity model. (b) Starting velocity model for FWI.

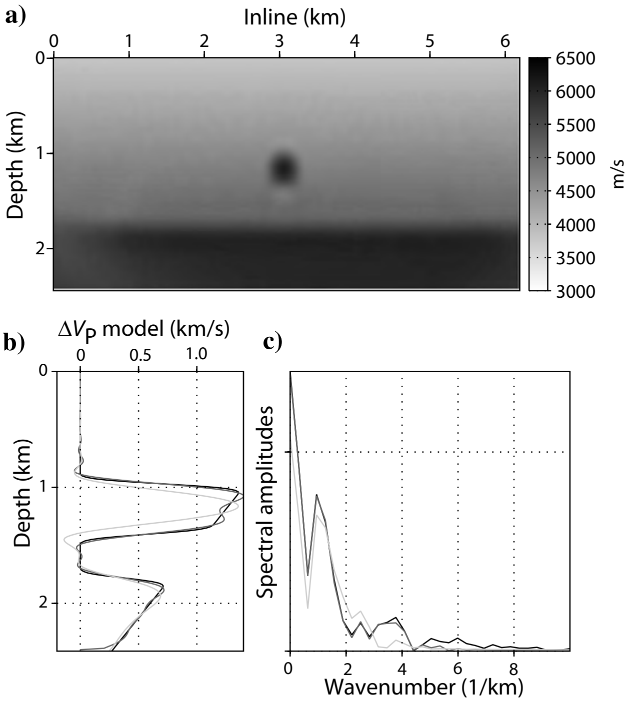

Figure 9. Imaging the inclusion/interface model. (a) Final FWI velocity model. (b) Vertical profile across the inclusion extracted from the true perturbation velocity model (black), from the true perturbation velocity model after low-pass filtering at the theoretical resolution of 12.15-Hz FWI (gray), and from the final FWI perturbation velocity model. (c) Spectral amplitudes as a function of wavenumber of the profiles shown in (b). These spectra suggest that wavenumbers between 0.003 and 0.004 m^{-1} corresponding to wavelengths of 333 and 250 m, respectively, were unsuccessfully imaged.

Inclusion + interface velocity model

A more realistic example consists of a velocity-gradient layer above a homogeneous layer. A high-velocity inclusion corresponding to a velocity perturbation of >1 km/s was incorporated into the velocity-gradient layer (Figure 8). The minimum and maximum velocities were 3.8 and 6.0 km/s, respectively. It was discretized on a $100 \times 100 \times 40$ grid with a grid spacing of $h = 62.5$ m, which corresponds to a physical domain of $6.25 \times 6.25 \times 2.5$ km. The grid spacing h was kept constant over the successive monofrequency inversions and was set according to the maximum inverted frequency.

The starting model for inversion was the velocity-gradient layer extended down to the bottom of the model (Figure 8). The acquisition geometry consisted of two regular grids of $17 \times 17 = 289$ sources and receivers deployed on the surface. The receiver grid was shifted with respect to the shot grid, such that each receiver was midway between four adjacent shots. This source-receiver configuration was chosen to avoid recording high-amplitude zero-offset data, which degrade the inversion conditioning. The distance between either two sources or receivers was 312.5 m. We sequentially inverted five frequencies, ranging from 1.76 to 12.15 Hz, and computed 20 iterations per frequency.

The final FWI velocity model is shown in Figure 9a. A vertical profile across the inclusion extracted from the final FWI perturbation model (i.e., the difference between the final FWI model and the starting model) is shown in Figure 9b. It is compared with that extracted from the true perturbation model after low-pass filtering at the theoretical resolution of FWI at 12 Hz. The bottom layer is well recovered, thanks to the large offset coverage, allowing it to image a broad range of the layer wavelengths quantitatively. The shape of the inclusion is recovered incompletely with respect to the expected resolution of the imaging at 12 Hz, although the velocity amplitude in the inclusion is recovered fully (Figure 9b and c).

The spectra of the two profiles reveal that the amplitudes of the low wavenumbers were recovered incompletely. This is probably from an insufficient number of iterations, as suggested by the plot of the objective function as a function of iteration number (Figure 10). Moreover, some discrepancies in the shape of the two spectra of Figure 9c for wavenumbers greater than 0.0017 m^{-1} suggest that the inversion may have converged toward a local minimum. This may have been caused by the high amplitudes of the model perturbations and the related complex interactions between waves multiscattered between the bottom of the inclusion and the top of the layer, which make the inverse problem more nonlinear. We speculate that simultaneous inversion of multiple frequencies following the multiscale approach of Bunks et al. (1995) may help to manage this nonlinearity.

Qualitative inspection of the vertical profiles also reveals a slight deficit of small (vertical) wavenumbers in the image of the inclusion and of the bottom layer (suggested by the negative velocity perturbations with respect to the true model). This deficit is again explained by the surface acquisition geometry, which illuminates the vertical components of the wavenumber vector with reflections only.

SEG/EAGE overthrust model

The 3D SEG/EAGE overthrust model is a constant-density acoustic model dimensions are $20 \times 20 \times 4.65$ km (Aminzadeh et al., 1997). It is discretized with 25-m^3 cells, representing a uniform mesh of $801 \times 801 \times 187$ nodes. The minimum and maximum velocities in the overthrust model are 2.2 and 6.0 km/s, respectively (Figure 11).

Overthrust model: Channel target

Because of limited computer resources, our first application was restricted to a small section of the overthrust model centered on a channel. The maximum frequency involved in the inversion was 15 Hz. A horizontal slice and a vertical section of the model are shown in Figure 12. The model dimensions are $7 \times 8.75 \times 2.25$ km and is discretized with a grid spacing $h = 50$ m, leading to a $141 \times 176 \times 46$ grid. The minimum and maximum velocities are 3.3 and 6.0 km/s, respectively. The acquisition geometry consists of two regular grids of 44×33 sources and receivers on the surface corresponding to 1452 sources and receivers. The distance between two sources or two receivers is 200 m. The receiver array is shifted according to the source array, following the geometry in the previous example.

We sequentially inverted five frequencies ranging from 5 to 15 Hz. For each frequency, we computed seven iterations. The starting velocity model was obtained by smoothing the true model with a wavenumber filter having a cut-off wavenumber of $1/500\ m^{-1}$ (Figure 13). The final FWI model provides a low-pass version of the true model (Figure 14). To assess the accuracy of the FWI, we low-pass-filtered the true model in the time domain with a cut-off frequency of 15 Hz to mimic the exact velocity model that would have been inferred by FWI (Figure 15). Qualitative comparison between the final FWI velocity model and the low-pass-filtered true model shows good agreement between the two models.

Figure 16 compares a vertical profile extracted from the starting model, the low-pass true velocity model, and the final FWI model. The agreement is reasonably good with, again, a slight deficit of

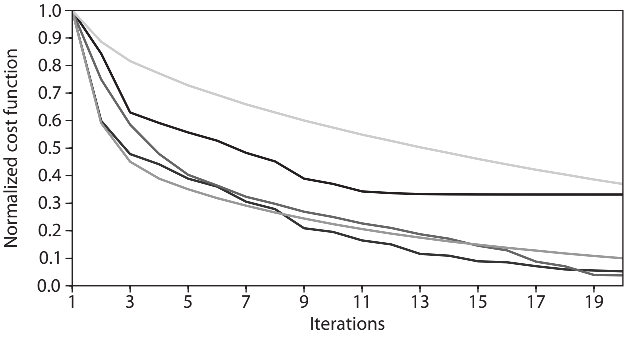

Figure 10. Imaging the inclusion/interface model: cost function versus iteration number for each inverted frequency (dark to light gray for increasing frequency). Convergence was not achieved, which partially explains underestimated velocities.

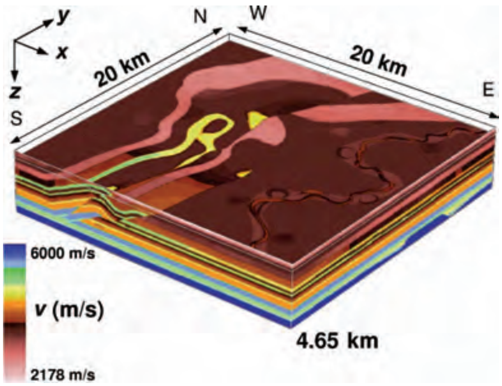

Figure 11. The 3D SEG/EAGE overthrust model (Aminzadeh et al., 1997).

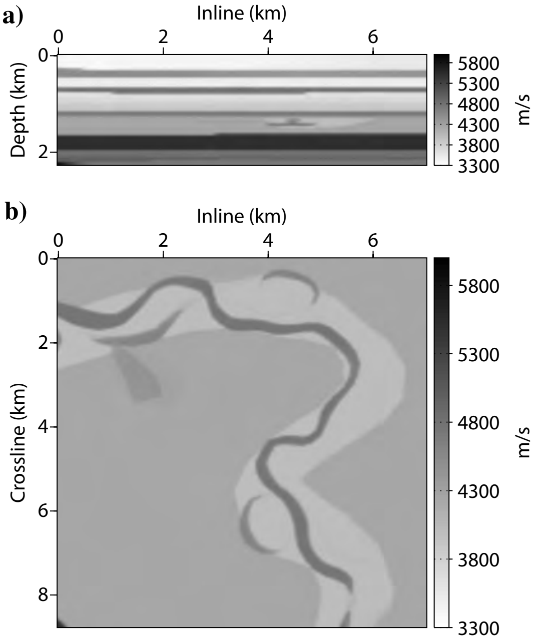

Figure 12. Imaging a channel in the overthrust model, true velocity model. (a) Cross-section at $x = 4$ km. (b) Horizontal slice at $z = 1.5$ km.

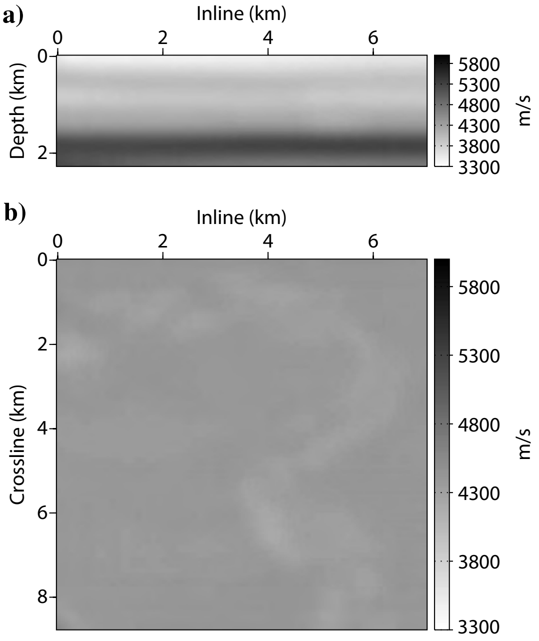

Figure 13. Imaging a channel in the overthrust model, starting velocity model. (a) Cross section at $x = 4$ km. (b) Horizontal slice at $z = 1.5$ km.

small wavenumbers in the FWI profile because of the surface-to-surface illumination. We also note an underestimation of velocities in the deep part of the model (see the high-velocity layer above 2 km in depth). This may result from an insufficient number of iterations. Indeed, the deep structures are mainly constrained by later-arriving reflections of smaller amplitude recorded at larger offsets. Misfit reduction may be slower for these arrivals because the value of the objective function is dominated by the residuals of the high-amplitude shallow arrivals during the first iterations.

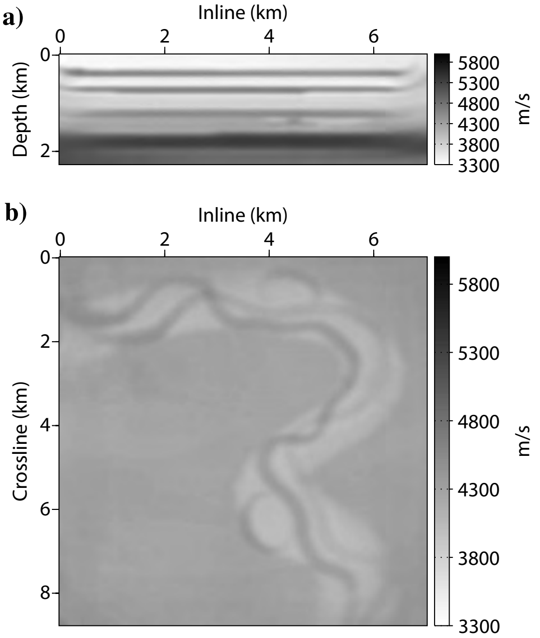

Figure 14. Imaging a channel in the overthrust model, FWI velocity model after successive inversion of the five frequencies, $f = 14.9$ Hz. (a) Cross section at $x = 4$ km. (b) Horizontal slice at $z = 1.5$ km.

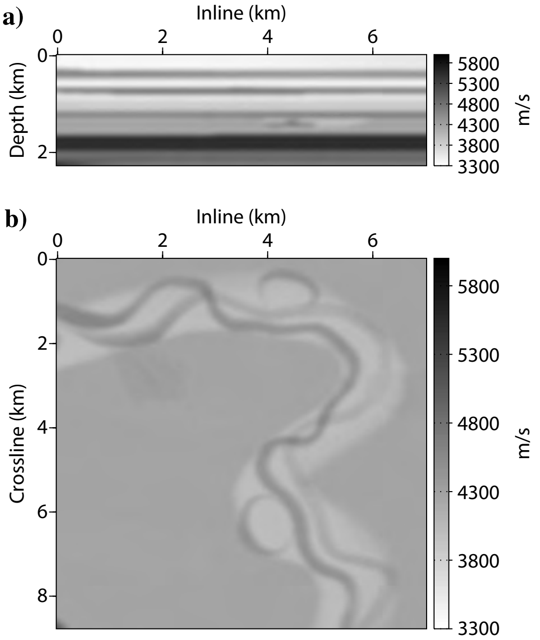

Figure 15. Imaging a channel in the overthrust model, low-pass filtered true velocity model (~15 Hz). (a) Cross section at $x = 4$ km. (b) Horizontal slice at $z = 1.5$ km.

The weighting operator in the data space corresponding to an amplitude gain with offset (equation 6) may accelerate the reduction of the long-offset residuals at the partial expense of the short-offset ones during late iterations. However, this strategy requires that we ensure short-offset residuals are reduced enough to avoid propagating errors associated with inaccurate shallow structures deeper in the model. This detailed tuning of 3D FWI requires further investigation.

To perform this application, we used 60 MPI processes distributed over 15 dual-core biprocessor nodes. Each MPI process used 1.5 GB of RAM (Table 1). Seven iterations of the inversion of one frequency

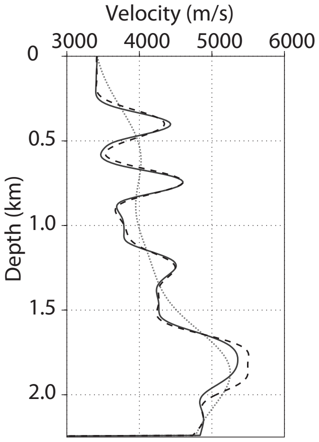

Figure 16. Imaging a channel in the overthrust model — comparison between vertical profiles extracted from the starting model (dotted line), the low-pass filtered true model (dashed line), and the final FWI velocity model (solid line). The profile is located at ($x = 3.5$ km, $y = 4.4$ km).

Table 1. Computational cost of imaging the overthrust model (channel system).

Requirement term	Value
$\mathrm{MEM}_{\mathrm{FACTO\ ALL}}$ (GB)	67.0
$\mathrm{MEM}_{\mathrm{FACTO\ PROC}}$ (GB)	1.5
$\mathrm{TIME}_{\mathrm{FACTO}}$ (s)	510.0
$\mathrm{TIME}_{\mathrm{SOLVE\ ALL}}$ (s)	1270.0
$\mathrm{TIME}_{\mathrm{SOLVE\ SOURCE}}$ (s)	0.9
$\mathrm{TIME}_{\mathrm{GRADIENT}}$ (s)	4.0
$\mathrm{TIME}_{\mathrm{diag\ HESSIAN\ a}}$ (s)	3093.0
$\mathrm{TIME}_{\mathrm{ITERATION}}$ (s)	18,865.0

$\mathrm{MEM}_{\mathrm{FACTO\ ALL}}$ — Total memory allocated during factorization

$\mathrm{MEM}_{\mathrm{FACTO\ PROC}}$ — Average allocated memory per working processor during factorization

$\mathrm{TIME}_{\mathrm{FACTO}}$ — Elapsed time for factorization

$\mathrm{TIME}_{\mathrm{SOLVE\ ALL}}$ — Total elapsed time for multishot resolution

$\mathrm{TIME}_{\mathrm{SOLVE\ SOURCE}}$ — Elapsed time for resolution for one source

$\mathrm{TIME}_{\mathrm{GRADIENT}}$ — Elapsed time to compute gradient

$\mathrm{TIME}_{\mathrm{diag\ HESSIAN\ a}}$ — Elapsed time to compute diagonal Hessian

$\mathrm{TIME}_{\mathrm{ITERATION}}$ — Average elapsed time to process one iteration took approximately 45 hours. Table 1 lists information related to run time and memory requirement for LU factorization, multishot resolutions (both tasks being devoted to the forward problem), gradient, and diagonal Hessian computation. Run time for the solution phase is very small (0.9 s per source) and illustrates the main advantage of FD modeling methods based on direct solvers for tomographic applications involving a few thousand sources. Computation of the gradient is also negligible in the frequency domain (4 s) as a result of summation without disk swapping over a very compact volume of data limited to few frequency components. Increasing the number of cores in the inversion would have reduced computational time significantly but would have increased the memory requirement because of overheads during parallel LU factorization (Operto et al., 2007).

Overthrust model: Thrust target

We now consider imaging a significant target of the overthrust model that incorporates the main thrusts of the model (Figure 17). The minimum and maximum velocities are 2.2 and 6.0 km/s, respectively. The model dimensions are 13.425 × 13.425 × 4.65 km.

The acquisition geometry consists of two coincident 43 × 43 = 1849 grids of sources and receivers deployed on the surface. The distance between two sources or two receivers is 300 m. A receiver spacing of 300 m is representative of the spacing between two adjacent nodes in a dense 3D wide-azimuth node survey (Clarke et al., 2007). We used the same spacing between two adjacent shots and receivers, although a more representative shot survey could have been designed by using smaller shot and line intervals in the dip-line and crossline directions, respectively. The increased number of shots should not dramatically increase the computational time needed to compute the gradient because the residuals recorded at the shot positions (by virtue of shot-receiver reciprocity) can be propagated at one time for each receiver.

The extra computational cost caused by a denser shot survey would have resulted from building and storing denser residual sources ($\Delta\mathbf{d}^*$ in equation 4) and from the more expensive backward/forward substitutions solving $\mathbf{A}^{-1}\mathbf{W}_d\Delta\mathbf{d}^*$. On the contrary, the CPU time required to compute the diagonal Hessian would increase dramatically because it requires a forward simulation for nonredundant shot and receiver positions. However, a good approximation of the diagonal Hessian can be computed on a coarser shot grid with a shot interval of the same order as the receiver's (Operto et al., 2006).

We sequentially inverted three frequencies — 3.5, 5, and 7 Hz — and computed 10 iterations per frequency. For this application, we adapted the grid interval to the inverted frequency. Grid intervals were $h = 150$, 100, and 75 m for frequencies of 3.5, 5, and 7 Hz, respectively. The source and receiver positions were chosen to coincide with the position of the nodes of the FD grids associated with the three inverted frequencies. This allowed us to bypass the problem of accurate implementation of point sources in a coarse FD grid, a critical issue in 3D frequency-domain FWI when the grid interval is adapted to the frequency (Hicks, 2002). These discretizations led to grids measuring 90 × 90 × 32, 135 × 135 × 47, and 180 × 180 × 63. The starting velocity model was obtained by smoothing the true model with a 3D Gaussian function having a correlation length of 1000 m in the three directions (Figure 18).

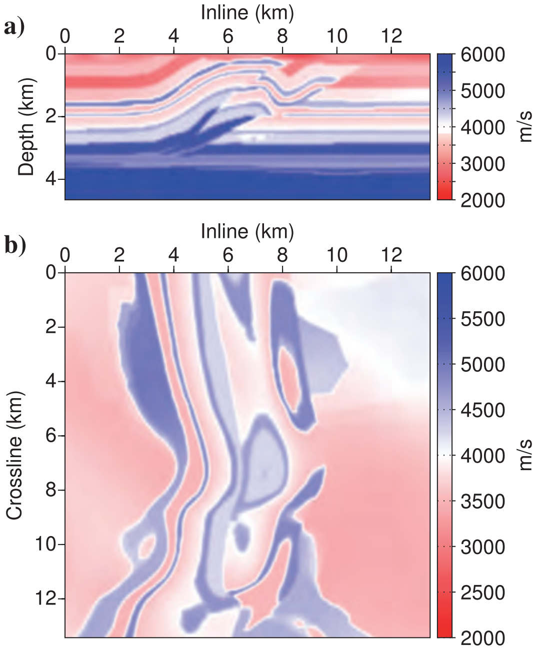

Figure 17. Imaging the thrust system in the overthrust model, true velocity model. (a) Cross section at $x = 3.3$ km. (b) Horizontal slice at $z = 1.5$ km.

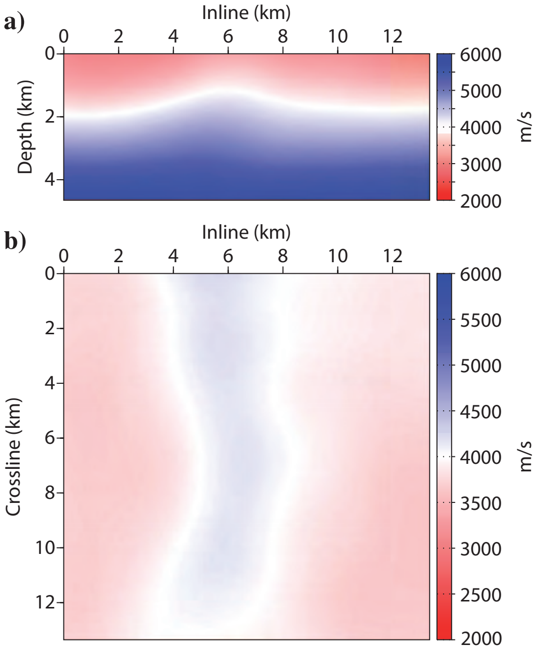

Figure 18. Imaging the thrust system in the overthrust model, starting velocity model. (a) Cross section at $x = 3.3$ km. (b) Horizontal slice at $z = 1.5$ km.

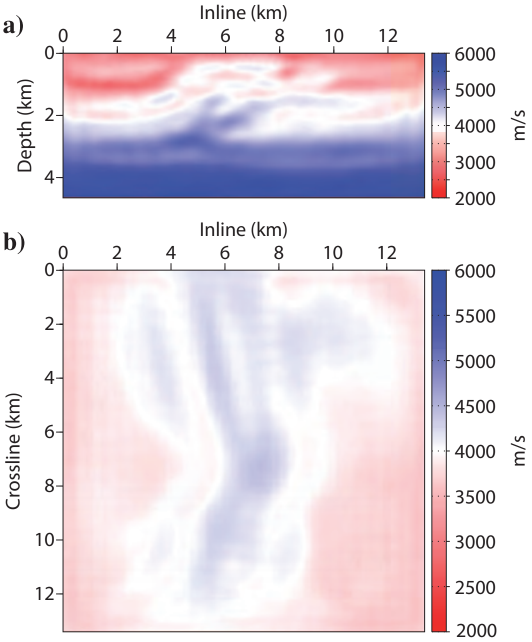

Figure 19. Imaging the thrust system in the overthrust model, 3.5-Hz FWI velocity model. (a) Cross section at $x = 3.3$ km. (b) Horizontal slice at $z = 1.5$ km.

The FWI velocity models after inversion of the 3.5-, 5-, and 7-Hz frequencies are shown in Figures 19–21. One can note a square pattern superimposed on the horizontal and vertical slices of the FWI velocity models obtained after inverting frequencies of 5 and 7 Hz (Figures 20 and 21). The size of the square matches the shot and receiver spacing, suggesting that it corresponds to the footprint of the coarse acquisition geometry. This footprint increases from 5 to 7 Hz as model resolution increases. The acquisition footprint has no preferential orientation because both shots and receivers are deployed uniformly all over the surface, with a constant spacing in the dip and cross directions.

Another illustration of the footprint of acquisition coarseness on 3D frequency-domain FWI is illustrated by Sirgue et al. (2007a). For this application, we ran 40 MPI processes distributed over 10 dual-core biprocessor nodes (four MPI processes/node) for the 150-m grid (frequency = 3.5 Hz), 60 processes distributed over 20 dual-core biprocessor nodes (three MPI processes/node) for the 100-m grid (frequency = 5 Hz), and 90 processes distributed over 30 dual-core biprocessor nodes (three MPI processes/node) for the 75-m grid (frequency = 7 Hz). The number of processes per dual-core biprocessor node decreased from four to three as the size of the problem increased in order to increase the amount of shared memory assigned to each processor for large problems. This process optimized memory use at a partial expense of run time because memory overhead decreased with number of processes. The 10 iterations took about 24, 72, and 120 hours for the 3.5-, 5-, and 7-Hz frequencies, respectively. More detailed information is shown in Table 2.

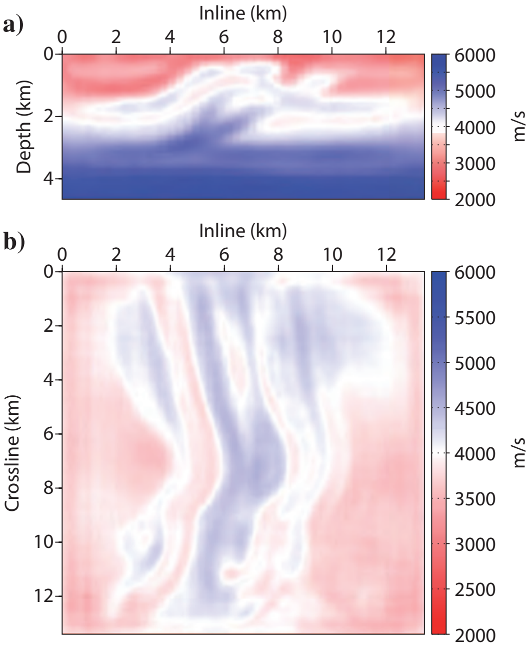

Figure 20. Imaging the thrust system in the overthrust model, 5-Hz FWI velocity model. (a) Cross section at $x = 3.3$ km. (b) Horizontal slice at $z = 1.5$ km.

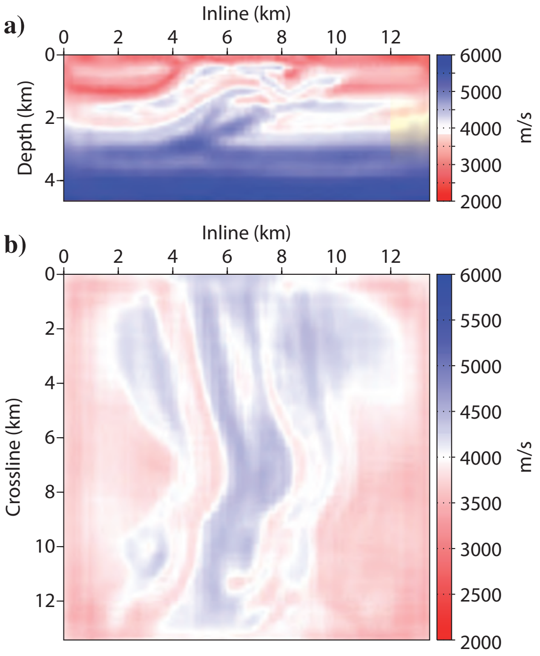

Figure 21. Imaging the thrust system in the overthrust model, 7-Hz FWI velocity model. (a) Cross section at $x = 3.3$ km. (b) Horizontal slice at $z = 1.5$ km.

We also performed the simulations using multithreading. We used 20, 40, and 60 processes distributed over 10, 20, and 30 dual-core biprocessor nodes (i.e., one process per processor) for the 3.5-, 5-, and 7-Hz inversions, respectively, and two threads per processor in each case (i.e., one thread per core). Fewer processes were used in the multithread configuration, so less distributed memory was allocated during LU factorization as a result of reduced memory overhead (Table 3). Moreover, CPU time was reduced significantly for the LU-factorization and multiple-shot solution phases (Table 3). These results suggest that nodes with a large amount of shared memory and multiple cores provide the optimal architecture to perform FD wave modeling based on a direct solver to take advantage of multithreading while mitigating the memory requirement.

Figure 22 shows the convergence rate for each processed frequency. This convergence rate can be compared with that shown in Figure 7a of Sirgue et al. (2007a) for 3.5-Hz frequency, keeping in mind that the whole overthrust model is imaged in Sirgue et al. (2007a).

In Figures 23 and 24, the data fit is illustrated in the frequency domain for the 3.5-, 5-, and 7-Hz frequencies. We compare the spectral amplitude and phase of the monochromatic wavefields at the receiver positions computed in the true velocity model and in the FWI models at the first and last iterations of the three monofrequency inversions. One shot is located at the upper-left corner of the receiver plane (Figure 23); the second shot is in the middle of the receiver array (Figure 24). The misfit reduction between the first and last iterations is more obvious at 3.5 Hz, illustrating slower convergence as frequency increases (Figure 22). We also note this misfit reduction is more effective for the shot located in the middle of the receiver array, again illustrating the difficulty of matching the lower-amplitude arrivals recorded at larger offsets. As for the channel case study, the objective function is less sensitive to the residuals of these low-amplitude arrivals; more iterations would have been required to cancel them.

Table 2. Computational cost of imaging the overthrust model (thrust system). The memory available per MPI process is 2, 2.7, and 2.7 GB for frequencies of 3.5, 5, and 7 Hz, respectively.

Requirement term	At 3.5 Hz frequency	At 5.0 Hz frequency	At 7.0 Hz frequency
N_P	40	60	90
GRID	$90\times90\times32$	$135\times135\times47$	$180\times180\times63$
$\mathrm{MEM}_{\mathrm{FACTO\ ALL}}$ (GB)	11.5	45.0	124.0
$\mathrm{MEM}_{\mathrm{FACTO\ PROC}}$ (GB)	0.3	0.8	1.8
$\mathrm{TIME}_{\mathrm{FACTO}}$ (s)	72.0	340.0	1850.0
$\mathrm{TIME}_{\mathrm{SOLVE\ ALL}}$ (s)	310.0	990.0	3450.0
$\mathrm{TIME}_{\mathrm{SOLVE\ SOURCE}}$ (s)	0.2	0.6	1.8
$\mathrm{TIME}_{\mathrm{GRADIENT}}$ (s)	0.7	1.5	35.0
$\mathrm{TIME}_{\mathrm{diag\ HESSIAN\ a}}$ (s)	1999.0	3432.0	4000.0
$\mathrm{TIME}_{\mathrm{ITERATION}}$ (s)	2940.0	13,650.0	44,870.0

N_P — Number of MPI processes
GRID — Dimension of the 3D FD grid
$\mathrm{MEM}_{\mathrm{FACTO\ ALL}}$ — Total memory allocated during factorization
$\mathrm{MEM}_{\mathrm{FACTO\ PROC}}$ — Average allocated memory per working processor during factorization
$\mathrm{TIME}_{\mathrm{FACTO}}$ — Elapsed time for factorization
$\mathrm{TIME}_{\mathrm{SOLVE\ ALL}}$ — Total elapsed time for multi-shot resolution
$\mathrm{TIME}_{\mathrm{SOLVE\ SOURCE}}$ — Elapsed time for resolution of one source
$\mathrm{TIME}_{\mathrm{GRADIENT}}$ — Elapsed time to compute gradient
$\mathrm{TIME}_{\mathrm{diag\ HESSIAN\ a}}$ — Elapsed time to compute diagonal Hessian
$\mathrm{TIME}_{\mathrm{ITERATION}}$ — Average elapsed time to MPI process one iteration

Table 3. Computational cost of imaging the overthrust model (thrust system) using a multithread configuration (two threads per MPI process). The memory available per MPI process is 4 GB for the three frequencies. Compare these values with those of Table 2.

Requirement term	At 3.5 Hz frequency	At 5.0 Hz frequency	At 7.0 Hz frequency
N_P	20	40	60
$\mathrm{MEM}_{\mathrm{FACTO\ ALL}}$ (GB)	9.9	40.0	108.0
$\mathrm{MEM}_{\mathrm{FACTO\ PROC}}$ (GB)	0.5	1.0	1.4
$\mathrm{TIME}_{\mathrm{FACTO}}$ (s)	56.0	222.0	650.0
$\mathrm{TIME}_{\mathrm{SOLVE\ ALL}}$ (s)	290.0	650.0	1375.0

N_P — Number of MPI processes
$\mathrm{MEM}_{\mathrm{FACTO\ ALL}}$ — Total memory allocated during factorization
$\mathrm{MEM}_{\mathrm{FACTO\ PROC}}$ — Average allocated memory per working processor during factorization
$\mathrm{TIME}_{\mathrm{FACTO}}$ — Elapsed time for factorization
$\mathrm{TIME}_{\mathrm{SOLVE\ ALL}}$ — Total elapsed time for multishot resolution

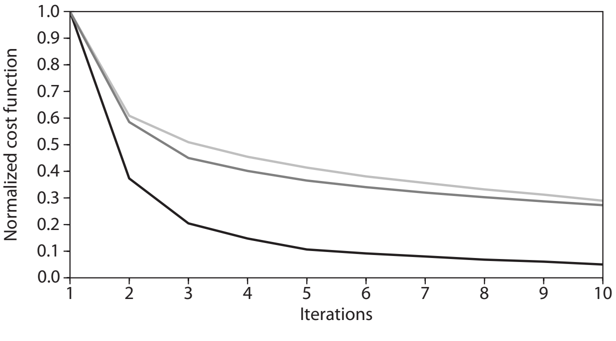

Figure 22. Imaging the thrust system in the overthrust model: objective function versus iteration number for the 3.5-, 5-, and 7-Hz frequencies (dark to light gray for increasing frequency).

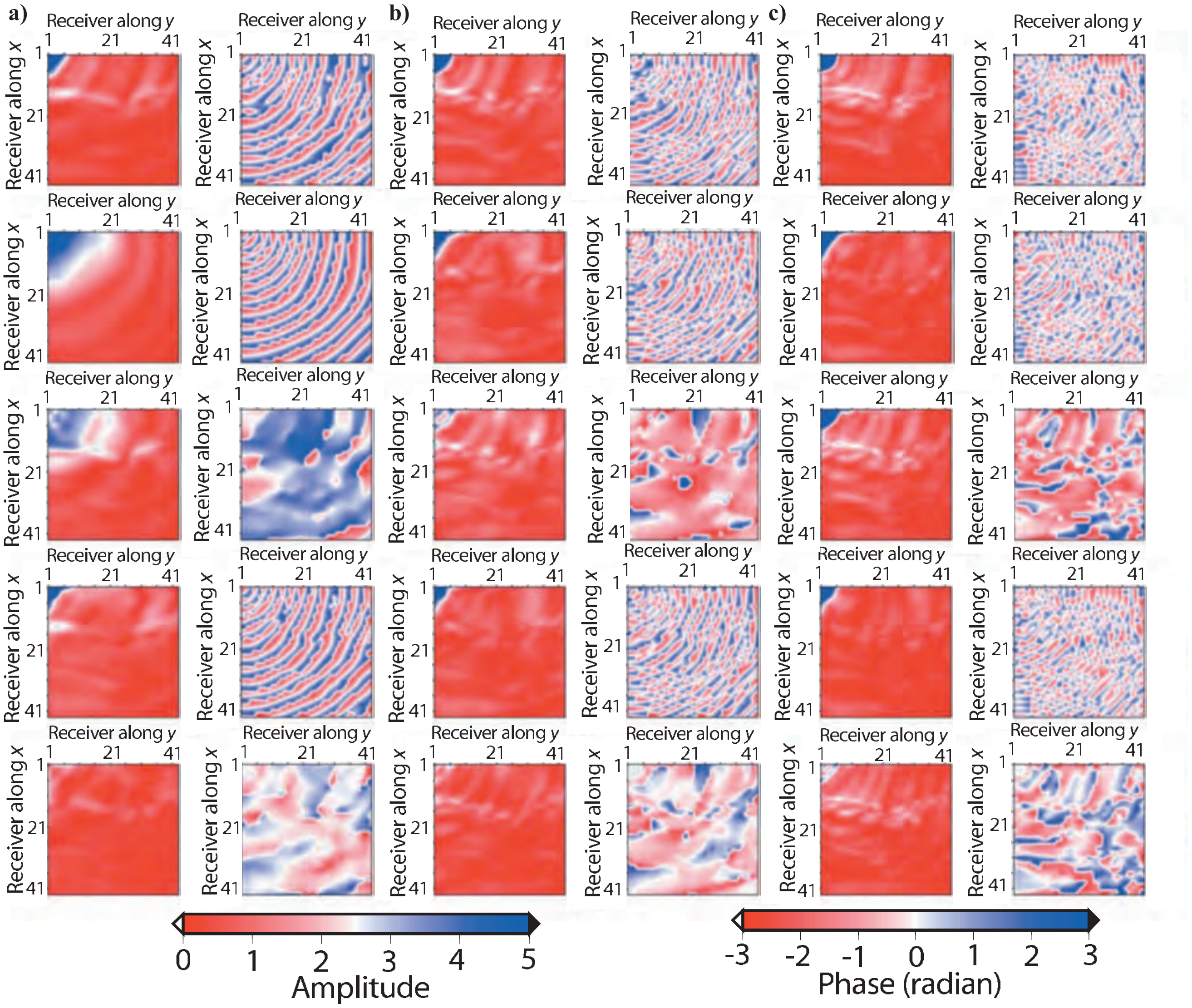

Figure 23. Imaging the thrust system in the overthrust model. Amplitude (left) and phase (right) of the (a) 3.5-Hz, (b) 5-Hz, and (c) 7-Hz wavefields at the receiver positions. The horizontal and vertical axes label the receiver number in the dip-line and crossline directions, respectively. The source is located in the upper-left corner. From top to bottom, the first and second panels show the wavefield computed in the true and the starting models, respectively; the third is the difference between the two wavefields. The fourth panel shows the wavefield computed in the final model of the frequency inversion, and the fifth is the residual between the wavefields computed in the true model and in the final model of the frequency inversion.

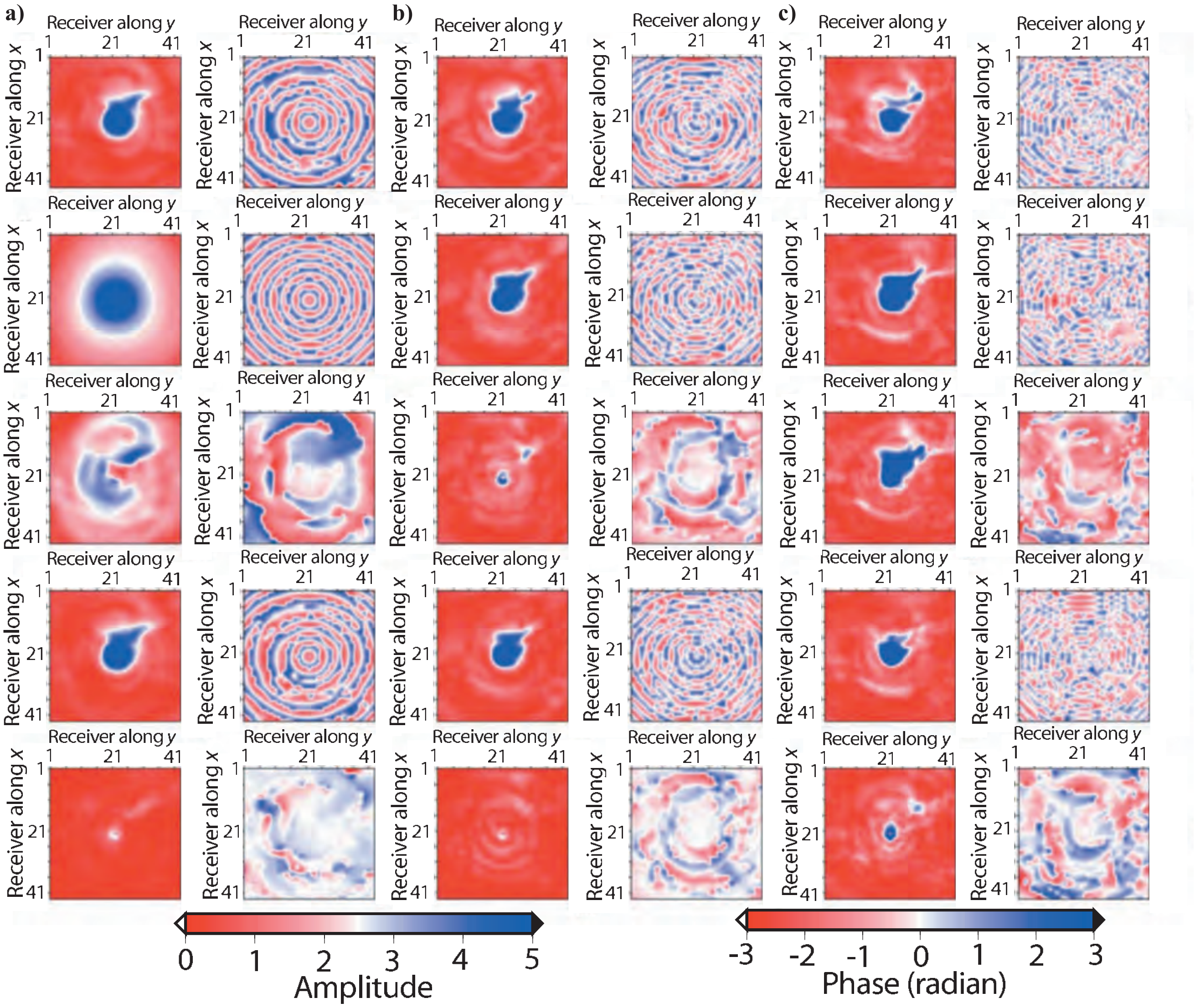

Figure 24. Imaging the thrust system in the overthrust model. The same as for Figure 23, but for a source located in the middle of the receiver array.

CONCLUSION

We have presented a 3D massively parallel frequency-domain FWI algorithm based on a direct solver. Advantages of our approach include the robustness of the forward problem provided by a high-performance direct solver, its efficiency to perform multishot simulations in relatively small 3D finite-difference grids, and a straightforward parallelization of the inverse problem resulting from a domain decomposition of the monochromatic wavefields performed by the direct solver. By robustness, we mean that the time required to perform a simulation with a direct solver depends less on the complexity of the velocity model than the time required to perform this simulation with an iterative solver. Its main drawbacks are the memory and CPU time complexity of the LU factorization phase and its limited scalability, which restricts the size of the models and the frequency bandwidth that can be addressed on realistic distributed-memory platforms.

We have presented several applications on synthetic examples of increasing complexity to validate the algorithm and to give insight into the feasibility of our approach. Some preliminary applications to the overthrust model suggest that frequency-domain FWI can be applied successfully at low frequencies (<7 Hz) on limited-size PC clusters to develop 3D velocity models with a maximum resolution on the order of half the wavelength (that is, 285 m for a velocity of 4000 m/s). This resolution scale should be compared with that provided by reflection tomography and migration-based velocity analysis to assess whether velocity models developed by 3D FWI can provide accurate background models for wave-equation PSDM.

Assessment of velocity models developed by 3D FWI as macromodels for PSDM will be one aim of future work. A second aim will be to assess whether accurate starting models for 3D FWI can be developed by first-arrival traveltime tomography, provided that global-offset recording is available. A third objective will be to mitigate the memory limitations imposed by using a direct solver only.

In addition to time-domain and iterative approaches, a possible direction is to evolve toward hybrid direct-iterative solvers implemented in a domain-decomposition method based on the Schur complement approach, for which the direct solver is applied to subdomains of limited dimension while the iterative solver is applied to the grid points located at the boundaries between the subdomains. Compared to a purely iterative approach, the expected benefit is that the iterative solver in the hybrid approach is applied to a system of smaller dimension (the Schur complement system) that is better preconditioned theoretically. The drawback is that the hybrid approach remains more memory demanding because of the direct solver. The relevance of this approach will have to be demonstrated when many sources are considered.

ACKNOWLEDGMENTS

We thank associate editor Tamas Nemeth, Bee Bednar, and three anonymous reviewers for their comments, which improved the manuscript. Access to the high-performance equipment at MESOCENTRE SIGAMM computer center provided the required computer resources; we gratefully acknowledge both this facility and the support of the staff. We are particularly grateful to A. Miniussi for his help during the installation of the software on the cluster. We thank Florent Sourbier (Geosciences Azur — CNRS) for his help during the development and installation of the FWT3D software on the MESOCENTRE SIGAMM. Finally, this work was conducted within the frame of the SEISCOPE consortium sponsored by BP, CGG-Veritas, EXXON-Mobil, Shell, and Total. This work was partly funded by ANR project ANR-05-NT05-2-42427.

REFERENCES

Amestoy, P. R., I. S. Duff, J. Y. L'Excellent, and J. Koster, 2007, Multifrontal massively parallel solver (MUMPS version 4.7.3) users'guide, http://enseeiht.fr/apo/MUMPS/, accessed June 2008.

Amestoy, P. R., A. Guermouche, J. Y. L'Excellent, and S. Pralet, 2006, Hybrid scheduling for the parallel solution of linear systems: Parallel Computing, **32**, 136–156.

Aminzadeh, F., J. Brac, and T. Kunz, 1997, 3-D salt and overthrust models: SEG/EAGE 3-D modeling series 1, SEG.

Ashcraft, C., and J. W. H. Liu, 1998, Robust ordering of sparse matrices using multisection: SIAM Journal on Matrix Analysis and Applications, **19**, 816–832.

Berenger, J.-P., 1994, A perfectly matched layer for absorption of electromagnetic waves: Journal of Computational Physics, **114**, 185–200.

Bickel, S. H., 1990, Velocity-depth ambiguity of reflection traveltimes: Geophysics, **55**, 266–276.

Biondi, B., and W. Symes, 2004, Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging: Geophysics, **69**, 1283–1298.

Bishop, T. N., K. P. Bube, R. T. Cutler, R. T. Langan, P. L. Love, J. R. Resnick, R. T. Shuey, and D. A. Spinder, 1985, Tomographic determination of velocity and depth in laterally varying media: Geophysics, **50**, 903–923.

Brenders, A. J., and R. G. Pratt, 2007a, Efficient waveform tomography for lithospheric imaging: Implications for realistic 2D acquisition geometries and low frequency data: Geophysical Journal International, **168**, 152–170.

———, 2007b, Full waveform tomography for lithospheric imaging: Results from a blind test in a realistic crustal model: Geophysical Journal International, **168**, 133–151.

Bunks, C., F. M. Saleck, S. Zaleski, and G. Chavent, 1995, Multiscale seismic waveform inversion: Geophysics, **60**, 1457–1473.

Capdeville, Y., 2005, Towards global earth tomography using the spectral element method: A technique based on source stacking: Geophysical Journal International, **162**, 541–554.

Claerbout, J. F., 1971, Toward a unified theory of reflector mapping: Geophysics, **36**, 467–481.

Clarke, R., G. Xia, N. Kabir, L. Sirgue, and S. Michell, 2007, Processing of a novel deepwater, wide-azimuth node seismic survey: The Leading Edge, **26**, 504–509.

Duff, I. S., and J. K. Reid, 1983, The multifrontal solution of indefinite sparse symmetric linear systems: ACM Transactions on Mathematical Software: ACM Transactions on Mathematical Software, **9**, 302–325.

Forgues, E., and G. Lambaré, 1997, Parameterization study for acoustic and elastic ray + Born inversion: Journal of Seismic Exploration, **6**, 253–278.

Hicks, G., 2002, Arbitrary source and receiver positions in finite-difference schemes using Kaiser windowed sinc functions: Geophysics, **67**, 156–166.

Hicks, G. J., and R. G. Pratt, 2001, Reflection waveform inversion using local descent methods: Estimating attenuation and velocity over a gas-sand deposit: Geophysics, **66**, 598–612.

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

Jo, C. H., C. Shin, and J. H. Suh, 1996, An optimal 9-point, finite-difference, frequency-space 2D scalar extrapolator: Geophysics, **61**, 529–537.

Lailly, P., 1984, The seismic inverse problem as a sequence of before stack migrations: SIAM Conference on Inverse Scattering, Theory and Application, 206–220.

Marfurt, K., 1984, Accuracy of finite-difference and finite-elements modeling of the scalar and elastic wave equation: Geophysics, **49**, 533–549.

Nihei, K. T., and X. Li, 2007, Frequency response modelling of seismic waves using finite difference time domain with phase sensitive detection (TD-PSD): Geophysical Journal International, **169**, 1069–1078.

Operto, S., G. Lambaré, P. Podvin, and P. Thierry, 2003, 3D ray-Born migration/inversion, part 2: Application to the SEG/EAGE overthrust experiment: Geophysics, **68**, 1357–1370.

Operto, S., C. Ravaut, L. Improta, J. Virieux, A. Herrero, and P. Dell' Aversana, 2004, Quantitative imaging of complex structures from multi-fold wide aperture seismic data: Geophysical Prospecting, **52**, 625–651.

Operto, S., J. Virieux, P. Amestoy, J.-Y. L'Éxcellent, L. Giraud, and H. Ben-Hadj-Ali, 2007, 3D finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver, A feasibility study: Geophysics, **72**, no. 5, SM195-SM211.

Operto, S., J. Virieux, J. X. Dessa, and G. Pascal, 2006, Crustal imaging from multifold ocean bottom seismometers data by frequency-domain full-waveform tomography: Application to the eastern Nankai Trough: Journal of Geophysical Research, **111**, B09306 doi: , doi: 10.1029/2005JB003835.

Operto, S., J. Virieux, and A. Ribodetti, 2005, Multiple arrival migration/inversion of real global aperture seismic data: 67th Annual Conference and Exhibition, EAGE, Extended Abstracts, P272.

Operto, S., J. Virieux, and F. Sourbier, 2008, Documentation for FWM2D/FWT2D programs: 2D acoustic frequency-domain waveform modelling/tomography, http://seiscope.unice.fr/FWT/doc.html, accessed December 2007.

Operto, S., S. Xu, and G. Lambaré, 2000, Can we image quantitatively complex models with rays?: Geophysics, **65**, 1223–1238.

Plessix, R. E., 2007, A Helmholtz iterative solver for 3D seismic-imaging problems: Geophysics, **72**, no. 5, SM185-SM194.

Pon, S., and L. Lines, 2005, Sensitivity analysis of seismic depth migrations: Geophysics, **70**, no. 2, S39-S42.

Pratt, R. G., 1999, Seismic waveform inversion in the frequency domain, part I: Theory and verification in a physic scale model: Geophysics, **64**, 888–901.

———, 2004, Velocity models from frequency-domain waveform tomography: Past, present and future, Presented at the 66th Annual Conference and Exhibition, EAGE, workshop W8.

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

Pratt, R. G., and M. H. Worthington, 1990, Inverse theory applied to multi-source cross-hole tomography. Part 1: Acoustic wave-equation method: Geophysical Prospecting, **38**, 287–310.

Ravaut, C., S. Operto, L. Improta, J. Virieux, A. Herrero, and P. dell' Aversana, 2004, Multi-scale imaging of complex structures from multifold wide-aperture seismic data by frequency-domain full-wavefield inversions: Application to a thrust belt: Geophysical Journal International, **159**, 1032–1056.

Riyanti, C. D., A. Kononov, Y. A. Erlangga, C. Vuik, C. Oosterlee, R. E. Plessix, and W. A. Mulder, 2007, A parallel multigrid-based preconditioner for the 3D heterogeneous high-frequency Helmholtz equation: Journal of Computational Physics, **224**, 431–448.

Sava, P. C., and B. Biondi, 2005, Wave-equation migration velocity analysis — I, Theory: Geophysical Prospecting, **52**, 593–606.

Shin, C., K. Yoon, K. J. Marfurt, K. Park, D. Yang, H. Y. Lim, S. Chung, and S. Shin, 2001, Efficient calculation of a partial derivative wavefield using reciprocity for seismic imaging and inversion: Geophysics, **66**, 1856–1863.

Shipp, R. M., and S. C. Singh, 2002, Two-dimensional full wavefield inversion of wide-aperture marine seismic streamer data: Geophysical Journal International, **151**, 325–344.

Sirgue, L., J. Etgen, and U. Albertin, 2007a, 3D full waveform inversion: Wide versus narrow azimuth acquisitions: 77th Annual International Meeting, SEG, Expanded Abstracts, 1760–1764.

Sirgue, L., T. J. Etgen, U. Albertin, and S. Brandsberg-Dahl, 2007b, System and method for 3D frequency-domain waveform inversion based on 3D time-domain forward modeling: U.S. Patent US2007/0282535 A1.

Sirgue, L., and R. G. Pratt, 2004, Efficient waveform inversion and imaging: A strategy for selecting temporal frequencies: Geophysics, **69**, 231–248.

Song, Z., P. Williamson, and R. Pratt, 1995, Frequency-domain acoustic-wave modeling and inversion of crosshole data, Part 2: Inversion method, synthetic experiments and real-data results: Geophysics, **60**, 784–795.

Sourbier, F., S. Operto, J. Virieux, P. Amestoy, and J. Y. L'Excellent, 2007, FWT2D: A massively parallel frequency-domain full-waveform algorithm for imaging acoustic media: 77th Annual International Meeting, SEG, Expanded Abstracts, 1893–1897.

Stekl, I., and R. G. Pratt, 1998, Accurate viscoelastic modeling by frequency-domain finite difference using rotated operators: Geophysics, **63**, 1779–1794.

Stekl, I., M. R. Warner, and A. P. Umpleby, 2007, 3D frequency domain waveform inversion — Synthetic shallow channel example: 69th Annual Conference and Exhibition, EAGE, Extended Abstracts, C026.

Stork, C., 1992, Reflection tomography in the post migrated domain: Geophysics, **57**, 680–692.

Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, **49**, 1259–1266.

———, 1987, Inverse problem theory: Methods for data fitting and model parameter estimation: Elsevier Scientific Publishing Co., Inc.

Toksöz, M. N., and D. H. Johnston, 1981, Seismic wave attenuation: SEG

Warner, M., I. Stekl, and A. Umpleby, 2007, Full wavefield seismic tomography — Iterative forward modelling in 3D: 69th Annual Conference and Exhibition, EAGE, Extended Abstracts, C025.

Williamson, P., 1991, A guide to the limits of resolution imposed by scattering in ray tomography: Geophysics, **56**, 202–207.

Williamson, P. R., and M. H. Worthington, 1993, Resolution limits in ray tomography due to wave behavior, Numerical experiments: Geophysics, **58**, 727–735.

Wu, R.-S., and M. N. Töksoz, 1987, Diffraction tomography and multisource holography applied to seismic imaging: Geophysics, **52**, 11–25.

Zhang, Y., S. Xu, and G. Zhang, 2006, Imaging complex salt bodies with turning-wave one-way wave equation: 76th Annual International Meeting, SEG, Expanded Abstracts, 2323–2327.

Zhang, Y., G. Zhang, D. Yingst, and J. Sun, 2007, Explicit marching method for reverse-time migration: 77th annual meeting, SEG, Expanded Abstracts, 2300–2304.

Bibliography

Abubakar, A., Hu, W., Habashy, T., and van den Berg, P. (2009). A finite-difference contrast source inversion algorithm for full-waveform seismic applications. *Geophysics*, This issue.

Akcelik, V. (2002). *Multiscale Newton-Krylov methods for inverse acoustic wave propagation.* PhD thesis, Carnegie Mellon University, Pittsburgh, Pennsylvania.

Aki, K. and Richards, P. (1980). *Quantitative Seismology: Theory and Methods.* W. H. Freeman & Co, San Francisco.

Amestoy, P., Davis, T. A., and Duff, I. S. (1996). An approximate minimum degree ordering algorithm. *SIAM J. Matrix. Anal. and Applics.*, 17:886–905.

Amestoy, P. R., Guermouche, A., L'Excellent, J. Y., and Pralet, S. (2006). Hybrid scheduling for the parallel solution of linear systems. *Parallel Computing*, 32:136–156.

Aminzadeh, F., Brac, J., and Kunz, T. (1997). *3-D Salt and Overthrust models.* Society of Exploration Geophysicists.

Ashcraft, C. and Liu, J. W. H. (1998). Robust ordering of sparse matrices using multisection. *SIAM Journal on Matrix Analysis and Applications*, 19(3):816–832.

Audebert, F., Nichols, D. E., Rekdal, T., Biondi, B., Lumley, D. E., and Urdaneta, H. (1997). Imaging complex geologic structure with single-arrival Kirchhoff prestack depth migration. *Geophysics*, 62(5):1533–1543.

Ben-Hadj-Ali, H., Operto, S., and Virieux, J. (2008). Velocity model building by 3D frequency-domain, full-waveform inversion of wide-aperture seismic data. *Geophysics*, 73(5):VE101–VE117.

Ben-Hadj-Ali, H., Operto, S., and Virieux, J. (2009a). Efficient 3D frequency-domain full waveform inversion (FWI) with phase encoding. In *Expanded Abstracts*, page 5812. European Association of Geoscientists & Engineers.

Ben-Hadj-Ali, H., Operto, S., and Virieux, J. (2009b). Three-dimensional frequency-domain full waveform inversion with phase encoding. In *79th Annual SEG Conference & Exhibition, Houston.* Society of Exploration Geophysicists.

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

Berkhout, A. J. and Wapenaar, C. P. A. (1989). One-way versions of the kirchhoff integral. *Geophysics*, 54:460–467.

Beydoun, W. B. and Tarantola, A. (1988). First Born and Rytov approximation: Modeling and inversion conditions in a canonical example. *J. acoust. Soc. Am.*, 83:1045–1055.

Billette, F. (1998). *Estimation de macro-modèles de vitesse en sismique réflexion par stéréotomographie.* PhD thesis, Université Paris VII.

Billette, F., Le Bégat, S., Podvin, P., and Lambaré, G. (2003). Practical aspects and applications of 2D stereotomography. *Geophysics*, 68:1008–1021.

Biondi, B. (2001). Kirchhoff imaging beyond aliasing. *Geophysics*, 66:654–666.

Biondi, B. L. (2006). *3D seismic imaging.* SEG.

Bishop, T. N., Bube, K. P., Cutler, R. T., Langan, R. T., Love, P. L., Resnick, J. R., Shuey, R. T., and Spinder, D. A. (1985). Tomographic determination of velocity and depth in laterally varying media. *Geophysics*, 50:903–923.

Born, M. and Wolf, E. (1993). *Principles of optics : electromagnetic theory of propagation, interference and diffraction of light; sixth edition with corrections.* Pergamon Press.

Brenders, A. J. and Pratt, R. G. (2006). Full waveform tomography for lithospheric imaging: results from a blind test in a realistic crustal model. *Geophysical Journal International.*

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

Brossier, R., Operto, S., and Virieux, J. (2009a). Seismic imaging of complex onshore structures by two-dimensional elastic frequency-domain full-waveform inversion. *Geophysics*, in press.

Brossier, R., Operto, S., and Virieux, J. (2009b). Two-dimensional seismic imaging of the valhall model from synthetic OBC data by frequency-domain elastic full-waveform inversion. *SEG Technical Program Expanded Abstracts*, in-press.

Brossier, R., Virieux, J., and Operto, S. (2008). 2D frequency-domain elastic full-waveform inversion using a $p0$ finite volume forward problem. In *Expanded Abstracts, 78th Annual SEG Conference & Exhibition, Las Vegas.* Society of Exploration Geophysics.

Bunks, C., abd S. Zaleski, F. M. S., and Chavent, G. (1995). Multiscale seismic waveform inversion. *Geophysics*, 60(5):1457–1473.

Capdeville, Y., Gung, Y., and Romanowicz, B. (2005). Towards global earth tomography using the spectral element method: a technique based on source stacking. *Geophysical Journal International*, 162:541–554.

Cerjan, C., Kosloff, D., Kosloff, R., and Reshef, M. (1985). A nonreflecting boundary condition for discrete acoustic and elastic wave equations. *Geophysics*, 50(4):2117–2131.

Červený, V., Molotkov, I. A., and Pšenčik, I. (1977). *Ray theory in Seismology.* Charles University Press, Praha.

Chen, H.-W. and McMechan, G. A. (1992). 3-D pre-stack depth migration for salt and subsalt structures using reverse-VSP data. *Journal of Seismic Exploration*, 1:281–291.

Choi, Y. and Shin, C. (2008). Frequency-Domain Elastic Full Waveform Inversion Using the New Pseudo-Hessian Matrix: Experience Of Elastic Marmousi 2 Synthetic Data. *Bulletin of the Seismological Society of America*, 98(5):2402–2415.

Claerbout, J. F. (1976). *Fundamentals of Geophysical Data Processing.* McGraw-Hill Book Co.

Claerbout, J. F. (1985). *Imaging the Earth's interior.* Blackwell Scientific Publications.

Clarke, R., Xia, G., Kabir, N., Sirgue, L., and Michell, S. (2007). Processing of a novel deepwater, wide-azimuth node seismic survey. *The leading edge*, April:504–509.

Clayton, R. and Engquist, B. (1977). Absorbing boundary conditions for acoustic and elastic wave equation. *Bull. Seismol. Soc. Am.*, 67:1529–1540.

Collino, F. and Tsogka, C. (2001). Application of the perfectly matched absorbing layer model to the linear elastodynamic problem in anisotropic heterogeneous media. *Geophysics*, 66:294–307.

Crase, E., Pica, A., Noble, M., McDonald, J., and Tarantola, A. (1990). Robust elastic non-linear waveform inversion: application to real data. *Geophysics*, 55:527–538.

Dessa, J. X., Operto, S., Kodaira, S., Nakanishi, A., Pascal, G., Virieux, J., and Kaneda, Y. (2004). Multiscale seismic imaging of the eastern nankai trough by full waveform inversion. *Geophysical Research Letters*, 31(L18606):doi:10.1029/2004GL020453.

Docherty, D. (1991). A brief comparison of some Kirchhoff integral formulas for migration. *Geophysics*, 56:1164–1169.

Docherty, P., Silva, R., Singh, S., Song, Z., and Wood, M. (1997). Migration velocity analysis using a genetic algorithm. *Geophysical Prospecting*, 45:865–878.

Duff, I. S. and Reid, J. K. (1983). The multifrontal solution of indefinite sparse symmetric linear systems. *ACM Transactions on Mathematical Software*, 9:302–325.

Erlangga, Y. A. (2005). *A robust and efficient iterative method for the numerical solution of the Helmholtz equation.* PhD thesis, Delft University of Technology.

Erlangga, Y. A. and Herrmann, F. J. (2008). An iterative multilevel method for computing wavefields in frequency-domain seismic inversion. *SEG Technical Program Expanded Abstracts*, 27(1):1956–1960.

Erlangga, Y. A., Vuik, C., and Osterlee, C. W. (2003). On a class of preconditioners for solving the helmholz equation. Reports of the Departement of Applied Mathematical Analysis report 03-01, Delft University of Technology.

Erlangga, Y. A., Vuik, C., and Osterlee, C. W. (2004). On a class of preconditioners for solving the helmholtz equation. *Apllied Numerical Mathematics*, 50:409–425.

Fletcher, R. and Reeves, C. M. (1964). Function minimization by conjugate gradient. *Computer Journal*, 7:149–154.

Frayssé, V., Giraud, L., and Gratton, S. (1997). A set of 'gmres' routines for real and complex arithmetics. Technical Report 49, CERFACS. Technical Report TR/PA/97/49.

Futterman, W. (1962). Dispersive body waves. *Journal of Geophysics Research*, 67:5279–5291.

Gao, F., Levander, A. R., Pratt, R. G., Zelt, C. A., and Fradelizio, G. L. (2006). Waveform tomography at a groundwater contamination site: Vsp-surface data set. *Geophysics*, 71(1):H1–H11.

Gauthier, O., Virieux, J., and Tarantola, A. (1986). Two-dimensional nonlinear inversion of seismic waveforms: numerical results. *Geophysics*, 51(7):1387–1403.

Gazdag, J. (1978). Wave equation migration with the phase-shift method. *Geophysics*, 43:1342–1351.

George, A. and Liu, J. W. (1981). *Computer solution of large sparse positive definite systems*. Prentice-Hall, Inc.

Gholami, A. and Siahkoohi, H. R. (2009). Simultaneous constraining of model and data smoothness for regularization of geophysical inverse problems. *Geophysical Journal International*, 176:151–163.

Haidar, A. (2008). *On the parallel scalability of hybrid linear solvers for large 3D problems*. PhD thesis, Institut National Polytechnique de Toulouse - CERFACS.

Herrmann, F. J., Erlangga, Y. A., and Lin, T. T. Y. (2009). Compressive simultaneous full-waveform simulation. *Geophysics*, 74(4):A35–A40.

Hicks, G. J. (2002). Arbitrary source and receiver positioning in finite-difference schemes using kaiser windowed sinc functions. *Geophysics*, 67:156–166.

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

Jaiswal, P., Zelt, C., Bally, A. W., and Dasgupta, R. (2008). 2-D traveltime and waveform inversion for improved seismic imaging: Naga Thrust and Fold Belt, India. *Geophysical Journal International*, doi:10.1111/j.1365-246X.2007.03691.x.

Jannane, M., Beydoun, W., Crase, E., Cao, D., Koren, Z., Landa, E., Mendes, M., Pica, A., Noble, M., Roeth, G., Singh, S., Snieder, R., Tarantola, A., and Trezeguet, D. (1989). Wavelengths of Earth structures that can be resolved from seismic reflection data. *Geophysics*, 54(7):906–910.

Jing, X., Finn, C. J., Dickens, T. A., and Willen, D. E. (2000). Encoding multiple shot gathers in prestack migration. In *70th Annual SEG Conference & Exhibition, Calgary*. Society of Exploration Geophysicists.

Jo, C. H., Shin, C., and Suh, J. H. (1996). An optimal 9-point, finite-difference, frequency-space 2D scalar extrapolator. *Geophysics*, 61:529–537.

Judson, D. R., Lin, J., Schultz, P. S., and Sherwood, J. W. C. (1980). Depth migration after stack. *Geophysics*, 45:361–375.

Karypis, G. and Kumar, V. (1998). *METIS - A software package for partitioning unstructured graphs, partitioning meshes and computing fill-reducing orderings of sparse matrices - Version 4.0*. University of Minnesota.

Kolsky, H. (1956). The propagation of stress pulses in viscoelastic solids. *Philosophical Magazine*, 1:693–710.

Komatitsch, D. and Martin, R. (2007). An unsplit convolutional perfectly matched layer improved at grazing incidence for the seismic wave equation. *Geophysics*, 72(5):SM155–SM167.

Kommedal, J. H., Barkved, O. I., and Howe, D. J. (2004). Initial experience operating a permanent 4C seabed array for reservoir monitoring at Valhall. *SEG Technical Program Expanded Abstracts*, 23(1):2239–2242.

Krebs, J., Anderson, J., Hinkley, D., Neelamani, R., Lee, S., Baumstein, A., and Lacasse, M. D. (2009). Fast full-wavefield seismic inversion using encoded sources. *Geophysics*.

Lailly, P. (1984). The seismic inverse problem as a sequence of before stack migrations. In Bednar, R. and Weglein, editors, *Conference on Inverse Scattering, SIAM, Philadelphia*, pages 206–220. Soc. Ind. appl. Math.

Lambaré, G. (2008). Stereotomography. *Geophysics*, 73(5):VE25–VE34.

Lambaré, G., Duquet, B., and Xu, S. (2002). 3D multi-arrival Kirchhoff versus wave equation migration : application to the SEG/EAGE Salt model. In *DIG annual report*, volume 2, pages 1–18. Depth Imaging Group.

Levander, A. R. (1988). Fourth-order finite-difference P-SV seismograms. *Geophysics*, 53(11):1425–1436.

Liu, F., Stolt, R. H., Hanson, D. W., and Day, R. S. (2002). Plane wave source composition: An accurate phase encoding scheme for prestack migration. In *72nd Annual SEG Conference & Exhibition, Salt Lake City*. Society of Exploration Geophysicists.

Liu, J. W. H. (1992). The multifrontal method for sparse matrix solution: theory and practice. *SIAM review*, 34(1):82–109.

Liu, Z. and Bleistein, N. (1995). Migration velocity analysis: Theory and an iterative algorithm. *Geophysics*, 60:142–153.

Luo, Y. and Schuster, G. T. (1990). Parsimonious staggered grid finite-differencing of the wave equation. *Geophysical Research Letters*, 17(2):155–158.

Marfurt, K. (1984). Accuracy of finite-difference and finite-elements modeling of the scalar and elastic wave equation. *Geophysics*, 49:533–549.

Miller, D., Oristaglio, M., and Beylkin, G. (1987). A new slant on seismic imaging: Migration and integral geometry. *Geophysics*, 52(7):943–964.

Min, D. J. and Shin, C. (2006). Refraction tomography using a waveform-inversion back-propagation technique. *Geophysics*, 71(3):R21–R30.

Mora, P. (1988). Elastic wavefield inversion of reflection and transmission data. *Geophysics*, 53:750–759.

Mora, P. (1989). Inversion = migration + tomography. *Geophysics*, 54(12):1575–1586.

Mora, P. R. (1987). Nonlinear two-dimensional elastic inversion of multi-offset seismic data. *Geophysics*, 52:1211–1228.

Morton, S. A. and Ober, C. C. (1998). Fastershot-record depth migration using phase encoding. In *68th Annual SEG Conference & Exhibition, New Orleans*. Society of Exploration Geophysicists.

MUMPS-team (2007). *MUMPS - MUltifrontal Massively Parallel Solver users' guide - version 4.7 (April 2007).* ENSEEIHT-ENS Lyon, http://www.enseeiht.fr/apo/MUMPS/ or http://graal.ens-lyon.fr/MUMPS.

Munns, J. W. (1985). The Valhall field: a geological overview. *Marine and Petroleum Geology*, 2:23–43,.

Nocedal, J. (1980). Updating Quasi-Newton Matrices With Limited Storage. *Mathematics of Computation*, 35(151):773–782.

Nocedal, J. and Wright, S. J. (1999). *Numerical Optimization.* New York, US : Springer.

Operto, S., Virieux, J., Amestoy, P., L'Éxcellent, J.-Y., Giraud, L., and Ben-Hadj-Ali, H. (2007). 3D finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver: A feasibility study. *Geophysics*, 72(5):SM195–SM211.

Operto, S., Virieux, J., and Dessa, J. X. (2006a). Frequency-domain full-waveform inversion of OBS wide-angle seismic data. In *Expanded Abstracts*. EAGE.

Operto, S., Virieux, J., Dessa, J. X., and Pascal, G. (2006b). Crustal imaging from multifold ocean bottom seismometers data by frequency-domain full-waveform tomography: application to the eastern Nankai trough. *Journal of Geophysical Research*, 111(B09306):doi:10.1029/2005JB003835.

Operto, S., Virieux, J., and Sourbier, F. (2008). Documentation for FWM2D/FWT2D programs: 2D acoustic frequency-domain waveform modelling/tomography. Technical report, http://seiscope.unice.fr/FWT/doc.html, accessed december 2007.

Phillips, W. S. and Fehler, M. C. (1991). Traveltime tomography: A comparison of popular methods. *Geophysics*, 56(10):1639–1649.

Plessix, R.-E. (2006). A review of the adjoint-state method for computing the gradient of a functional with geophysical applications. *Geophysical Journal International*, 167(2):495–503.

Plessix, R. E. (2007). A helmholtz iterative solver for 3D seismic-imaging problems. *Geophysics*, 72(5):SM185–SM194.

Plessix, R. E. (2009). 3-D frequency-domain full-waveform inversion with an iterative solver. *Geophysics*, in-press.

Plessix, R. E. and Perkins, C. (2009). 3D full-waveform inversion with a frequency-domain iterative solver. In *Expanded Abstracts*. European Association of Geoscientists & Engineers.

Podvin, P. and Lecomte, I. (1991). Finite difference computation of traveltimes in very contrasted velocity model : a massively parallel approach and its associated tools. *Geophys. J. Int.*, 105:271–284.

Polak, E. and Ribière, G. (1969). Note sur la convergence de méthodes de directions conjuguées. *Revue Française d'Informatique et de Recherche Opérationnelle*, 16:35–43.

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

Pratt, R. G. (1999). Seismic waveform inversion in the frequency domain, part I : theory and verification in a physic scale model. *Geophysical Journal International*, 64:888–901.

Pratt, R. G. and Goulty, N. R. (1991). Combining wave-equation imaging with traveltime tomography to form high-resolution images from crosshole data. *Geophysics*, 56(2):204–224.

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

Pratt, R. G., Song, Z. M., and Warner, M. (1996a). Two-dimensional velocity models from wide-angle seismic data by wavefield inversion. *Geophysical Journal International*, 124:323–340.

Pratt, R. G., Song, Z. M., Williamson, P. R., and Warner, M. (1996b). Two-dimensional velocity model from wide-angle seismic data by wavefield inversion. *Geophysical Journal International*, 124:323–340.

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

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1992). *Numerical recipes in FORTRAN : the art of scientific computing; second edition.* Cambridge university press.

Prieux, V., Operto, S., Brossier, R., and Virieux, J. (2009). Application of acoustic full waveform inversion to the synthetic Valhall model. *SEG Technical Program Expanded Abstracts*, in-press.

Ravaut, C., Operto, S., Improta, L., Virieux, J., Herrero, A., and dell'Aversana, P. (2004). Multi-scale imaging of complex structures from multi-fold wide-aperture seismic data by frequency-domain full-wavefield inversions: application to a thrust belt. *Geophysical Journal International*, 159:1032–1056.

Reshef, M. and Kosloff, D. (1986). Migration of common-shot gathers. *Geophysics*, 51:324–331.

Ribodetti, A., Valero, H. P., Operto, S., Virieux, J., and Gibert, D. (1998). Viscoacoustic asymptotic waveform inversion of ultrasonic laboratory data. In *Extended Abstracts*, pages 5–01. Eur. Ass. Expl.Geophys.

Riyanti, C. D., Erlangga, Y. A., Plessix, R. E., Mulder, W. A., Vuik, C., and Oosterlee, C. (2006). A new iterative solver for the time-harmonic wave equation. *Geophysics*, 71(E):57–63.

Riyanti, C. D., Kononov, A., Erlangga, Y. A., Vuik, C., Oosterlee, C., Plessix, R. E., and Mulder, W. A. (2007). A parallel multigrid-based preconditioner for the 3D heterogeneous high-frequency helmholtz equation. *Journal of Computational physics*, 224:431–448.

Romero, L. A., Ghiglia, D. C., Ober, C. C., and Morton, S. A. (2000). Phase encoding of shot records in prestack migration. *Geophysics*, 65, (2):426–436.

Saad, Y. (2003). *Iterative methods for sparse linear systems*, volume second edition. SIAM, Philadelphia.

Saenger, E. (2000). *Wave propagation in fractured media: theory and applications of the rotated staggered finite-difference grid.* Phd thesis, Karlsruhe University, Karlsruhe.

Scales, J. A. (1995). *Theory of seismic imaging.* New York, Springer-Verlag.

Scales, J. A. and Smith, M. L. (1994). *Introductory geophysical inverse theory.* Samizdat press.

Schultz, P. S. and Sherwood, J. W. C. (1980). Depth migration before stack. *Geophysics*, 45:376–393.

Sears, T., Singh, S., and Barton, P. (2008). Elastic full waveform inversion of multi-component OBC seismic data. *Geophysical Prospecting*, 56(6):843–862.

Sen, M. K. and Stoffa, P. L. (1995). *Global Optimization Methods in Geophysical Inversion.* Elsevier Science Publishing Co.

Shin, C. and Cha, Y. H. (2008). Waveform inversion in the Laplace domain. *Geophysical Journal International*, 173(3):922–931.

Shin, C. and Ha, W. (2008). A comparison between the behavior of objective functions for waveform inversion in the frequency and laplace domains. *Geophysics*, 73(5):VE119–VE133.

Shin, C., Jang, S., and Min, D. J. (2001). Improved amplitude preservation for prestack depth migration by inverse scattering theory. *Geophysical Prospecting*, 49:592–606.

Shin, C. and Min, D.-J. (2006). Waveform inversion using a logarithmic wavefield. *Geophysics*, 71(3):R31–R42.

Shin, C., Min, D.-J., Marfurt, K. J., Lim, H. Y., Yang, D., Cha, Y., Ko, S., Yoon, K., Ha, T., and Hong, S. (2002). Traveltime and amplitude calculations using the damped wave solution. *Geophysics*, 67:1637–1647.

Shin, C., Pyun, S., and Bednar, J. B. (2007). Comparison of waveform inversion, part 1: conventional wavefield vs logarithmic wavefield. *Geophysical Prospecting*, 55(4):449–464.

Sirgue, L., Barkved, O. I., Gestel, J. P. V., Askim, O. J., and Kommedal, J. H. . (2009). 3D waveform inversion on Valhall wide-azimuth OBC. In *Expanded Abstracts*, page U038.

Sirgue, L., Etgen, J., and Albertin, U. (2007a). 3D full-waveform inversion: Wide- versus narrow-azimuth acquisitions. *SEG Technical Program Expanded Abstracts*, 26(1):1760–1764.

Sirgue, L., Etgen, J. T., and Albertin, U. (2008). 3D Frequency Domain Waveform Inversion using Time Domain Finite Difference Methods. In *Proceedings 70th European Association of Geoscientists and Engineers, Conference and Exhibition, Roma, Italy*, page F022.

Sirgue, L., Etgen, T. J., Albertin, U., and Brandsberg-Dahl, S. (2007b). System and method for 3D frequency-domain waveform inversion based on 3D time-domain forward modeling. *US Patent Application Publication*, US2007/0282535 A1.

Sirgue, L. and Pratt, R. G. (2004). Efficient waveform inversion and imaging : a strategy for selecting temporal frequencies. *Geophysics*, 69(1):231–248.

Sourbier, F., Haidar, A., Giraud, L., Operto, S., and Virieux, J. (2008a). Frequency-domain full-waveform modeling using a hybrid direct-iterative solver based on a parallel domain decomposition method: A tool for 3D full-waveform inversion? *SEG Technical Program Expanded Abstracts*, 27(1):2147–2151.

Sourbier, F., Haiddar, A., Giraud, L., Brossier, R., Operto, S., and Virieux, J. (2008b). Frequency-domain full-waveform modeling using a hybrid direct-iterative solver based on a parallel domain decomposition method. In *70th Annual EAGE Conference & Exhibition, Roma*. Eur. Ass. Expl.Geophys.

Sourbier, F., Operto, S., Virieux, J., Amestoy, P., and L'Excellent, J.-Y. (2009). Fwt2d: A massively parallel program for frequency-domain full-waveform tomography of wide-aperture seismic data–part 1: Algorithm. *Computers & Geosciences*, 35(3):487 – 495.

Stekl, I. and Pratt, R. G. (1998). Accurate viscoelastic modeling by frequency-domain finite difference using rotated operators. *Geophysics*, 63:1779–1794.

Stolt, R. H. (1978). Migration by fourier transform. *Geophysics*, 43:23–48.

Symes, W. W. and Carazzone, J. J. (1991). Velocity inversion by differential semblance optimization. *Geophysics*, 56:654–663.

Tarantola, A. (1984a). Inversion of seismic reflection data in the acoustic approximation. *Geophysics*, 49(8):1259–1266.

Tarantola, A. (1984b). Linearized inversion of seismic reflection data. *Geophysical Prospecting*, 32:998–1015.

Tarantola, A. (1987). *Inverse problem theory: methods for data fitting and model parameter estimation.* Elsevier, New York.

Tikhonov, A. and Arsenin, V. (1977). *Solution of ill-posed problems.* Winston, Washington, DC.

Toksöz, M. N. and Johnston, D. H. (1981). *Geophysics reprint series, No. 2: Seismic wave attenuation.* Society of exploration geophysicists, Tulsa, OK.

Tromp, J., Tape, C., and Liu, Q. (2005). Seismic tomography, adjoint methods, time reversal and banana-doughnut kernels. *Geophysical Journal International*, 160:195–216.

Vidale, D. (1988). Finite-difference calculation of travel time. *Bull. seism. Soc .Am.*, 78:2062–2076.

Vidale, J. (1990). Finite-difference calculation of travel times in three dimensions. *Geophysics*, 55(5):521–526.

Vigh, D. and Starr, E. W. (2008). 3D prestack plane-wave, full-waveform inversion. *Geophysics*, 73(5):VE135–VE144.

Virieux, J. (1984). SH wave propagation in heterogeneous media, velocity-stress finite difference method. *Geophysics*, 49:1259–1266.

Virieux, J. (1986). P-SV wave propagation in heterogeneous media, velocity-stress finite difference method. *Geophysics*, 51:889–901.

Virieux, J. and Madariaga, R. (1982). Dynamic faulting studied by a finite difference method. *Bull. Seismol. Soc. Am.*, 72:345–369.

Virieux, J., Operto, S., Ben-Hadj-Ali, H., Brossier, R., Etienne, V., Sourbier, F., Giraud, L., and Haidar, A. (2009). Seismic wave modeling for seismic imaging. *The Leading Edge*, 28(5)(5):538–544.

Vuik, C., Erlangga, Y. A., and Osterlee, C. W. (2003). Shifted laplace preconditioners for the helmholz equations. Reports of the Departement of Applied Mathematical Analysis report 03-18, Delft University of Technology.

Warner, M., Stekl, I., and Umpleby, A. (2007). Full wavefield seismoc tomography - iterative forward modelling in 3D. In *Expanded Abstracts*, page C025. EAGE.

Warner, M., Stekl, I., and Umpleby, A. (2008). 3D wavefield tomography: synthetic and field data examples. *SEG Technical Program Expanded Abstracts*, 27(1):3330–3334.

Yee, K. S. (1966). Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media. *IEEE Trans. Antennas and Propagation*, 14:302–307.

Yilmaz, O. (2001). *Seismic data analysis*, volume 1. Society of Exploration Geophyscists: processing, inversion and interpretation of seismic data.

Yilmaz, Ö. and Chambers, R. (1984). Migration velocity analysis by wavefield extrapolation. *Geophysics*, 49:1664–1674.

Zelt, C. and Smith, R. B. (1992). Seismic traveltime inversion for 2-D crustal velocity structure. *Geophys. J. Int.*, 108:16–34.

Zhu, J. and Lines, L. R. (1996). Comparison of Kirchhoff and reverse-time migration methods with applications to prestack depth imaging of complex structures. In *Expanded Abstracts*, pages 539–542. Soc. Expl. Geophys.