Waveform-based velocity estimation from reflection seismic data

Yong Ma

- Doctoral Thesis -
Geophysics

Defended on October 26, 2012

Advisor: Prof. Dave Hale
Committee Members: Prof. Yaoguo Li
Prof. Mark Lusk
Prof. Roel Snieder
Prof. Andrzej Szymczak

Center for Wave Phenomena
Colorado School of Mines
Golden, Colorado 80401
303.384.2178
http://cwp.mines.edu
WAVEFORM-BASED VELOCITY ESTIMATION
FROM REFLECTION SEISMIC DATA

by
Yong Ma
ABSTRACT

The objective of this thesis is to investigate new methods in reflection seismology for estimating subsurface velocity models accurately and efficiently.

Full waveform inversion (FWI) is a state-of-the-art approach that can yield high-resolution models. However, multiple problems, including high computational cost, spurious local minima and possible geologically-implausible solutions, have prevented widespread applications of FWI for reflection data. These problems correspond to the fact that we in practice must estimate numerous model parameters with inadequate data (e.g., limited aperture, insufficient low frequencies).

To tackle the problems and achieve the goal of this thesis, I develop image-guided sparse-model FWI (IGFWI) and wave-equation reflection traveltime inversion (WERTI).

IGFWI uses subsurface structures to constrain the inversion of a sparse model. I extract the subsurface structures from migrated seismic images. With the structures, I construct a sparse model, which can, with substantially fewer parameters, explain the subsurface in a geologically plausible way. The IGFWI optimization problem is then formulated with respect to the sparse model that is constrained by structures. With fewer parameters, IGFWI converges faster in fewer iterations than does conventional FWI; with structural constraints, estimated models are geologically plausible; with a sparse representation of the model, blocky updates mitigate the lack of low frequencies. I solve IGFWI using a image-guided conjugate-gradient method.

With respect to the sparse model, I explicitly compute a projected Hessian matrix and its inverse using a projected BFGS (P-BFGS) method. Compared to the classic limited-memory BFGS (L-BFGS) method, P-BFGS substantially saves both computational time and memory, due to fewer model parameters of the sparse model. With
the P-BFGS method, I propose an efficient quasi-Newton method to solve large-scale inversion problems, such as FWI.

I also introduce WERTI to recover the low-wavenumber velocity background, and thereby mitigate local minima in reflection FWI with insufficient low frequencies. By alternating between WERTI and FWI, we can overcome the velocity-depth ambiguity in reflection traveltime inversion and estimate both the low- and high-wavenumber components of velocity models using only high-frequency reflection data.
# TABLE OF CONTENTS

ABSTRACT ......................................................... i
LIST OF FIGURES ................................................ ix
LIST OF TABLES .................................................. xv
ACKNOWLEDGMENTS ............................................. xvii

CHAPTER 1 INTRODUCTION ................................. 1
  1.1 Methods for Velocity Estimation ....................... 1
  1.2 Traveltime Inversion and Challenges .................... 3
    1.2.1 Traveltime Shift Estimation ....................... 3
    1.2.2 Velocity-Depth Ambiguity ......................... 4
  1.3 Full Waveform Inversion and Challenges ................. 4
    1.3.1 High Computational Cost ......................... 5
    1.3.2 Non-unique Solutions ............................ 5
  1.4 Thesis Overview ......................................... 7

CHAPTER 2 IMAGE-GUIDED SPARSE-MODEL FULL WAVEFORM
  INVERSION .............................................. 13
  2.1 Summary ................................................. 13
  2.2 Introduction ............................................ 13
  2.3 Sparse-Model Full Waveform Inversion ................. 16
    2.3.1 Inverse Problem in Sparse Model Space .......... 17
    2.3.2 Solution in Sparse Model ......................... 18
4.6 Conclusions ........................................... 105
4.7 Acknowledgments ................................. 108

CHAPTER 5 APPLICATION OF IMAGE-GUIDED FULL WAVEFORM INVERSION TO A 2D OCEAN-BOTTOM CABLE DATA SET ........................................... 109

5.1 Summary ........................................... 109
5.2 Introduction ...................................... 109
5.3 Field Data and Source Wavelet ................. 113
   5.3.1 Field Data .................................. 115
   5.3.2 Source Wavelet ............................. 115
5.4 Methodology ...................................... 115
   5.4.1 Synthetic Data .............................. 119
   5.4.2 Refraction FWI + Reflection FWI ............ 121
5.5 Inversion with Refraction Data .................. 121
5.6 Inversion with Reflection Data ................... 124
   5.6.1 Conventional FWI ........................... 124
   5.6.2 Image-Guided FWI ......................... 124
   5.6.3 Discussions ................................ 128
5.7 Conclusions ..................................... 132
5.8 Acknowledgments ............................... 135

CHAPTER 6 CONCLUSIONS AND SUGGESTIONS ................. 137

6.1 Main Results ..................................... 137
   6.1.1 Structure-Constrained Inversion ............... 137
LIST OF FIGURES

Figure 1.1 Schematic of the relationship between a model \( m \) and data \( d \). Simulating data in a model with a forward operator \( F \) illustrates a forward problem \( d = F(m) \). Retrieving model parameters or properties from recorded data with an inverse operator \( F^{-1} \) is a corresponding inverse problem, \( m = F^{-1}(d) \). In general, the operator \( F \) is nonlinear, so that the data \( d \) depend nonlinearly on the model \( m \).

Figure 1.2 Schematic of the local-minima problem in FWI. The data misfit has spurious local minima because of the nonlinearity in forward modeling \( F(m) = d \). If FWI starts from an initial model (starting model 1) that is too far way from the true model, FWI may converge to a local minimum. If FWI starts from an initial model (starting model 2) that is close to the true model, FWI may converge to the global minimum.

Figure 1.3 Two synthetic monochromatic seismograms with a period \( T \) in (a) and (c) are compared with the recorded seismogram in (b) to illustrate the cycle-skipping problem in FWI. The synthetic monochromatic seismogram in (a) has a time delay larger than half a period, and in this case FWI will update the model such that the \((n+1)\)th cycle in (a) matches the \(n\)th cycle in (b). Therefore, FWI produces an inaccurate model. The seismogram in (c) has a time delay smaller than half a period, and in this case FWI will update the model to make the \(n\)th cycle in (c) match the \(n\)th cycle in (b), leading to a correct model. Image courtesy of.

Figure 2.1 A uniform sample selection (a) and a pseudo-random sample selection (b). A total of 165 samples (white dots) are chosen from the densely sampled model space.

Figure 2.2 A migrated image \( I(x) \) (a) and a distance map \( d(x) \) of the migrated image (b). Zero distance indicates reflectors, and nonzero values indicate areas between reflectors.
Figure 2.3 A chosen sample (the white dot) in geological layers and three different exclusion regions: a rectangular, a circle, and a structurally constrained region (black). An exclusion region is where no a second sample appears. 27

Figure 2.4 A metric tensor field $\mathbf{D}(\mathbf{x})$ illustrated by ellipses (a) and structurally constrained sample selection (b). A total of 165 samples (white dots) are chosen for sparse-model FWI. 29

Figure 2.5 The Marmousi II velocity model (a) and the initial velocity (b) for FWI, which is a highly smoothed version of the true velocity. 31

Figure 2.6 A shot gather from the recorded data $\mathbf{d}$ (a), a shot gather from the synthetic data $\mathbf{F}(\mathbf{m}_0)$ (b), and the initial data residual $\mathbf{d} - \mathbf{F}(\mathbf{m}_0)$ (c). 33

Figure 2.7 The gradient $\mathbf{g}_0$ (a) and the image-guided gradient $\mathbf{R}\mathbf{R}^T\mathbf{g}_0$ (b). 34

Figure 2.8 Estimated velocity models with the conventional FWI (a) and the image-guided FWI (b), both in the 10th iteration. 35

Figure 2.9 The data residual for one shot after 10 iterations of conventional FWI (a) and image-guided FWI (b). 37

Figure 2.10 Convergence of between image-guided FWI and conventional FWI: the data misfit function (a) and the model misfit function (b). 38

Figure 2.11 Comparison of vertical velocity profiles at distance = 2.5 km (a) and the corresponding wavenumber-domain representation (b). 39

Figure 2.12 Using conventional FWI to enhance sharp boundaries such as fault planes. The IGFWI velocity in Figure 2.8b is used as an initial model in conventional FWI to produce this result. 40

Figure 3.1 Schematic for computing a projected Hessian $\tilde{\mathbf{H}}_i$. The projected Hessian reduces computation time and memory consumption. 52

Figure 3.2 The Marmousi II velocity model (a) and an initial velocity model (b). 58

Figure 3.3 Tensors that indicate structural characteristics of the model (a) and 165 sample locations of a sparse model space (b). 60
Figure 3.4  The initial Hessian $\tilde{H}_0$ (a), the Hessian matrix $\tilde{H}_1$ after the 1st iteration (b), the Hessian update $\tilde{H}_1 - \tilde{H}_0$ in the 1st iteration (c), and the inverse Hessian $\tilde{H}_1^{-1}$ (d). ......................... 62

Figure 3.5  Eigenvalues of projected Hessian in 1st (a), 4th (b), 7th (c), and 10th (d) iterations. ......................... 63

Figure 3.6  Eigenvectors of $\tilde{H}_1$ corresponding to the largest (a) and the smallest (b) eigenvalues. Geometrically, an eigenvector is a direction on the multidimensional surface of an objective function $E(m)$, and the associated eigenvalue determines the curvature on the surface. If the model is updated in the eigenvector direction in (a), the rate of change of $E(m)$ is largest. If the model is updated in the eigenvector direction in (b), the rate of change of $E(m)$ is smallest. ......................... 65

Figure 3.7  Update directions $p_1$ in FWI (a) and image-guided FWI (b). ....... 67

Figure 3.8  Velocity models estimated with FWI (a) and image-guided FWI (b). Both in the 10th iteration. ......................... 68

Figure 3.9  Update directions $p_1$ in QNFWI with L-BFGS (a) and QNFWI with P-BFGS (b). ......................... 70

Figure 3.10  Velocity models estimated with QNFWI with L-BFGS (a) and QNFWI with P-BFGS (b). Both in the 10th iteration. ......................... 71

Figure 3.11  Convergence of FWI, image-guided FWI, QNFWI with L-BFGS, and QNFWI with P-BFGS: the data misfit function (a) and the model misfit function (b). ......................... 73

Figure 4.1  True velocity model (a) that contains a high-velocity Gaussian anomaly and an initial model (b) for inversion. ......................... 90

Figure 4.2  Gradient $g_F(m^r)$ (a) and updated velocity (b) in the 1st iteration of conventional FWI, which fails to estimated the anomaly. ....... 91

Figure 4.3  Common shots gathers simulated (a) in the true model (Figure 4.1a) and (b) in the FWI model (Figure 4.2b). The data difference between (a) and (b) is displayed in (c), where cycle skipping appears in distance greater than 2 km. ......................... 93
Figure 4.4  (a) Normalized travelt ime misfit estimated with dynamic warping between the recorded data (Figure 4.3a) and the synthetic data (Figure 4.3b). The maximum shift is greater than 0.5, indicating an existence of cycle skipping.  (b) One example of the adjoint source $\mathcal{A}(x_r, t; x_s)$ in WERTI.  

Figure 4.5  Gradient $g_T(m^*)$ (a) in the first iteration and updated velocity (b) in the 10th iteration of WERTI.  

Figure 4.6  Gradient $g_F(m^*)$ (a) in the first iteration and updated velocity (b) in the 10th iteration of ’s modified FWI. Due to strong side lobes in the gradient, this method converges very slowly.  

Figure 4.7  True velocity model (a) that is part of the Sigsbee2A model and the initial model (b) for inversion.  

Figure 4.8  Two common shot gathers simulated in the true model (Figure 4.7a). The maximum offset is about 3 km.  

Figure 4.9  RTM images produced with the true model (a) and the initial model (b).  

Figure 4.10  Estimated velocity model (a) with 10 iterations of conventional FWI and the corresponding RTM image (b).  

Figure 4.11  (a) - (b): Two common shot gathers simulated in the FWI model (Figure 4.10a).  (c) - (d): Corresponding travelt ime shifts to the recorded data in Figure 4.8 estimated by dynamic warping. Large time shift indicates the failure of conventional FWI in recovering the background velocity.  

Figure 4.12  Estimating velocity model (a) with 10 iterations of WERTI and the corresponding RTM image (b).  

Figure 4.13  Convergence of WERTI: traveltime misfit function (a) and model misfit function (b).  

Figure 4.14  Using the velocity model in Figure 4.12a as the new initial model, conventional FWI refines the velocity model with high-wavenumber details.
Figure 4.15 (a) - (b): Two common shot gathers simulated in the final model (Figure 4.14). (c) - (d): Corresponding traveltime shifts to the recorded data in Figure 4.8 estimated by dynamic warping.

Figure 4.16 Data misfit (a) and model misfit (b) in conventional FWI and FWI after WERTI.

Figure 5.1 One 2D survey line extracted from a 3D OBC survey. The red dot indicates the location of a shot; white dots are receivers spanning about 12 km; green dots are common-depth point locations.

Figure 5.2 One typical shot gather for the 2D OBC data set. This shot gather shows an offset of about 12 km. Because of this large offset, significant refraction energy appears at far offset. The red line indicates the traveltime of the direct arrival and energy that arrives earlier than this line is refraction.

Figure 5.3 Source signature (a) that is measured in laboratory and its spectrum (b).

Figure 5.4 Source wavelet (a) that is used for inversion and its spectrum (b). This source wavelet is obtained by applying a bandpass filter to the source signature in (a).

Figure 5.5 Initial P-wave velocity model.

Figure 5.6 1D profiles of anisotropy parameters: $\epsilon$ (a) and $\delta$ (b).

Figure 5.7 Synthetic data simulated with the initial velocity model and anisotropic parameters.

Figure 5.8 Estimated velocity (a) after 5 iterations of refraction FWI and the difference (b) between this estimation and the initial velocity.

Figure 5.9 Gradient $g$ of the objective function in reflection FWI. The velocity for computing this gradient is the refraction-FWI updated model in (a).

Figure 5.10 Estimated velocity (a) after 5 iterations of reflection FWI and the difference (b) between this estimation and the initial velocity.

Figure 5.11 Structure features (a) indicated by tensor fields (ellipses) and a sparse model represented by dots (b).
Figure 5.12 Image-guided gradient $\mathbf{R} \mathbf{R}^T \mathbf{g}$ of the objective function in reflection FWI. The velocity for computing this gradient is the refraction-FWI updated model in (a). ........................................... 128

Figure 5.13 Estimated velocity (a) after 5 iterations of image-guided reflection FWI and the difference (b) between this estimation and the initial velocity. ................................................................. 129

Figure 5.14 Zoom-in views of the initial model (a) and the refraction updated model (b). ................................................................. 130

Figure 5.15 Zoom-in views of the reflection updated model (a) and the image-guided reflection updated model (b). ........................................... 131

Figure 5.16 RTM with the initial model (a) and the refraction-FWI model (b). 133

Figure 5.17 RTM with the reflection-FWI model (a) and the image-guided reflection-FWI model (d). ................................................................. 134

Figure B.1 A binary image (a) and its distance transform (b). ..................... 152
LIST OF TABLES

Table 3.1  Update directions $\mathbf{p}_i$ for the conjugate gradient (CG) method, the image-guided CG method, the quasi-Newton method with L-BFGS, and our quasi-Newton method with P-BFGS. .......................... 66

Table 4.1  Comparison between WERTI and FWI. ................................. 86
ACKNOWLEDGMENTS

When I stepped into the CWP office in August 2007 with literally no geophysical background, I was not quite sure whether it was a right idea or not. And now, as my Ph.D. study is approaching a conclusion, I must say that the past several years have taught me a lot, making an irreplaceable period in my life with both cheers and frustrations.

I first want to express my gratitude to my thesis advisor, Dr. Dave Hale, who not only has given me adequate freedom to work on my Ph.D. thesis but also has offered me tremendous help when I encountered difficulties along the way. In particular, Dave’s sharp scientific insights for multidisciplinary geophysical research and his ability to clearly and elegantly explain complex ideas always amaze me. I am also deeply grateful to my committee members: Dr. Roel Snieder, Dr. Yaoguo Li, Dr. Mark Lusk, and Dr. Andrzej Szymczak, for the diversity of their insights and numerous suggestions, which helped my research and course work tremendously.

Finding a Ph.D. thesis topic is not necessarily easy, and it could be the very first obstacle in the long journey. A special thanks goes to Dr. Zhaobo (Joe) Meng, who inspired and encouraged me to start the topic of this thesis when I took my first internship with ConocoPhillips in Summer 2009. Special thanks also go to Dr. Yunqing Shen, Dr. Jianxin (Jerry) Yuan and Dr. Bin Gong who guided and mentored me on the projects in another two internships with ConocoPhillips in 2010 and 2011. This thesis work is sponsored by the research agreement (SST-20090254-SRA) between ConocoPhillips Company and Colorado School of Mines.

Doing research is only one part of the entire Ph.D. study, and taking wonderful classes is the other big part. Dr. Ilya Tsvankin introduced me to seismology; I learned what the word “migration” means in seismic imaging from Dr. Paul Sava; Dr. Roel
Snieder and Dr. Ken Larner taught me the art of being a scientist; Dr. Dave Hale’s sharpness in computing made me believe the importance of good skills in geophysical computing.

I received tremendous help from the wonderful staff at CWP and the department. Pam, Michelle and Dawn handled all the administration; John helped in fixing all kinds of problems in my Linux workstation; Shingo and Barbara are great communication and publication specialists; Diane has been teaching me how to improve my technical writing since my first semester at Mines.

Many friends in and out of CWP have helped and encouraged me along the way. The senior students have all offered me valuable advice which made my life at CWP and Mines much easier. I specially thank the members of cTeam (Luming Liang, Ran Xuan, Simon Luo, Farhad Bazargani, Andrew Muños) and the members of iTeam (Jia Yan, Tongning Yang, Francesco Perrone) for inspirational discussions.

Finally, I want to thank my family. When I was lost, hesitated, or even thought about quitting at some moment, they are the sources of all my courage to calm down and carry on. Without them, I simply could not survive and accomplish the Ph.D. study.
The ultimate goal of exploration seismology is to estimate a quantitatively accurate model of the subsurface from measured seismic data. Among many parameters of the subsurface, a high-resolution seismic velocity model is an emerging demand in various aspects of exploration geophysics, from processing raw seismic data to monitoring reservoir production.

Seismic migration has evolved from producing a simple structural image (Claerbout, 1985) of the subsurface to rendering an image with correct amplitudes and other attributes that can describe the interior of the earth. To accomplish this, pre-stack depth migration, e.g., reverse time migration (RTM), depends more on velocity information than do other seismic processing steps (Etgen et al., 2009). The accuracy of seismic depth imaging depends greatly on the accuracy of the velocity model, especially in complex media. For instance, the estimation of an accurate velocity model with high resolution in and below a salt body is essential for sub-salt imaging (Jiao et al., 2008; Oezsen, 2004; Wang et al., 2008). High resolution velocity models also benefit seismic pore pressure prediction (Batzle & Wang, 1992), which helps geoscientists improve their understanding of hydrocarbon reservoirs, identify potential drilling hazards, and improve well positioning.

1.1 Methods for Velocity Estimation

In general, retrieving earth’s velocity models from recorded seismic data requires solving an inverse problem, as illustrated by Figure 1.1. Seismic migration can extract and locate reflectivities of the subsurface by producing a structural image. However, a structural image alone is not sufficient to fully provide and interpret properties of
the subsurface. Among the many reasons for this insufficiency, the most obvious and intrinsic one is that seismic migration behaves as the adjoint operator rather than the inverse operator. In other words, a migrated image only approximates the solution of the inverse problem.

\[
\text{Forward problem}
\]

\[
\begin{align*}
\text{Model} & \quad m \\
F & \quad d \\
F^{-1} & \quad \text{Data}
\end{align*}
\]

\[
\text{Inverse problem}
\]

Figure 1.1: Schematic of the relationship between a model \( m \) and data \( d \). Simulating data in a model with a forward operator \( F \) illustrates a forward problem \( d = F(m) \). Retrieving model parameters or properties from recorded data with an inverse operator \( F^{-1} \) is a corresponding inverse problem, \( m = F^{-1}(d) \). In general, the operator \( F \) is nonlinear, so that the data \( d \) depend non linearly on the model \( m \).

Various inversion methodologies have been developed in past decades to build seismic velocity models of the subsurface. These methods can be generally divided into two categories:

- image-domain methods: migration velocity analysis (MVA) (Yilmaz & Chambers, 1984) and wave-equation MVA (WEMVA) (Sava & Biondi, 2004; Shen,
This thesis focuses on the data-domain approaches.

1.2 Traveltime Inversion and Challenges

Traveltime inversion begins with extraction of refraction or reflection traveltime from large sets of recorded data. An inverse problem is then posed with the objective to match traveltime measurements obtained from recorded seismic data and corresponding synthetic seismic data simulated in a trial model. One can implement traveltime inversion by using ray tracing (Bishop et al., 1985; Stork, 1988) or solving wave equations (Luo & Schuster, 1991; Woodward, 1992). In this thesis, I choose the latter wave-equation based traveltime inversion, which can handle complex media where ray-based methods may fail. Traveltime inversion is suitable for estimating the large-scale background of a velocity model, and it has been successful in applications of cross-well data (Luo & Schuster, 1991) and refraction data (Zelt & Barton, 1998). Nevertheless, for reflection data, traveltime inversion is still plagued by the following two problems.

1.2.1 Traveltime Shift Estimation

In cross-well or refraction traveltime inversion, first arrivals are used to measure the traveltime, and in these cases, traveltime shifts between recorded and synthetic data are normally considered time invariant. In reflection traveltime inversion, however, one should measure traveltime shifts that vary with time. Manually picking reflection events is the most reliable way to do this but is also time consuming, especially with today’s huge amounts of data. More automatic methods for estimating traveltime shifts are commonly based on a crosscorrelation of the recorded data and the synthetic data (Luo & Schuster, 1991; van Leeuwen & Mulder, 2010). However, when traveltime shifts change rapidly, crosscorrelation-based methods may
suffer from the cycle-skipping problem, which causes errors in estimated traveltime shifts and subsequently makes reflection traveltime inversion problematic.

1.2.2 Velocity-Depth Ambiguity

Reflection traveltime inversion aims to determine both the large-scale velocity background and the small-scale reflector positions (Bishop et al., 1985). However, reflection traveltime inversion may encounter an ambiguity between velocity and reflector depth (Stork & Clayton, 1986), especially when the offset-to-depth ratio is small (Bube et al., 2002). Consider a horizontal reflector embedded in a constant background velocity. If in the initial model the background velocity is too small and the reflector is too shallow, the reflection traveltime can still be the same as that in the true model. In order to obtain the true model, one must increase the background velocity and meanwhile push down the reflector. To cope with this ambiguity, Snieder et al. (1989) and Clément et al. (2001) reparameterize velocity models to be functions of vertical traveltime instead of depth.

1.3 Full Waveform Inversion and Challenges

The objective of FWI is to estimate subsurface model by matching the synthetic data \( F(\mathbf{m}) \) and recorded data \( \mathbf{d} \) in a comprehensive way, such that all information in waveforms (e.g., traveltimes, amplitudes, converted waves, multiples) is accounted for in the data misfit function \( E(\mathbf{m}) \). A commonly used form of \( E(\mathbf{m}) \) is a least-squares function \( E(\mathbf{m}) = \frac{1}{2} \| \mathbf{F}(\mathbf{m}) - \mathbf{d} \|^2 \).

FWI is nowadays considered to be superior over traveltime inversion and MVA largely because it in principle can estimate a subsurface model with generally higher resolution (Operto et al., 2004) than can traveltime inversion and MVA. Although FWI has a long history and definite benefits, we have not seen its widespread applications in realistic cases mainly because of two major obstacles — high computational
cost and non-unique solutions.

1.3.1 High Computational Cost

One often solves FWI using gradient-based methods, which normally requires multiple iterations to minimize the data misfit, and then computational cost is proportional to the number of required iterations. To reduce iterations, one may reduce the size of the model space (number of parameters in inversion). Various compression methods are employed for this purpose, such as Fourier transform (Fichtner & Trampert, 2011b), wavelet transform (Loris et al., 2007; Meng & Scales, 1996; Simons et al., 2011), curvelet transform, etc. However, such methods do not account for geological structures in the subsurface, and so these methods may yield models that are geologically implausible.

1.3.2 Non-unique Solutions

The second obstacle is that the inverse problem posed by FWI has no unique solution. This is intrinsically because in FWI, as in other underdetermined inversions, we always aim to solve a nonlinear ill-posed problem with insufficient data. Therefore, many different models may yield synthetic data that match recorded data within a reasonable tolerance.

The forward operator $F$ is generally a nonlinear function of the model $m$, thereby generating local minima in the data misfit function $\frac{1}{2} \| F(m) - d \|^2$, as illustrated in Figure 1.2. The presence of local minima is one source of the nonuniqueness. That is, if the initial model is too far away from the true model, iterative methods may converge to a local minimum. To eliminate local minima, one can linearize the inverse problem by linearizing the forward operator $F$ (Tarantola, 1984). As a consequence, the data misfit function becomes $\frac{1}{2} \| Fm - d \|^2$, which is a purely quadratic function of $m$ and contains only a global minimum. However, this linearization approach is
Figure 1.2: Schematic of the local-minima problem in FWI. The data misfit has spurious local minima because of the nonlinearity in forward modeling $F(m) = d$. If FWI starts from an initial model (starting model 1) that is too far way from the true model, FWI may converge to a local minimum. If FWI starts from an initial model (starting model 2) that is close to the true model, FWI may converge to the global minimum.
valid only if the initial model is in the vicinity of the true model, and this assumption is rarely true in real applications. In addition, the nonlinearity becomes more severe in FWI with reflection data (Snieder, 1998).

Cycle skipping also causes nonunique solutions to FWI. As shown in Figure 1.3, cycle skipping occurs if the phase difference (traveltime shift) between synthetic and recorded data is larger than half a period of the dominant wavelet. In practice, the cycle-skipping problem appears because it is usually difficult to obtain an initial model with adequate low-wavenumber components or to record data with sufficient low frequencies. To mitigate cycle skipping, multiscale approaches (Bunks, 1995; Sirgue & Pratt, 2004) have been proposed. Such methods recursively add higher-wavenumber details to models first computed from lower-frequency data; however, unfortunately, these methods depend fundamentally on the fidelity of low frequencies in the recorded data. In practice, the low frequencies required to bootstrap multiscale FWI may be unavailable because low-frequency data are often contaminated by noise.

1.4 Thesis Overview

The objective of this thesis is to provide solutions to overcome these problems in previous discussions, so that in reflection seismology, one can estimate velocity models of the subsurface effectively and efficiently.

In Chapter 2, I propose image-guided sparse-model FWI (IGFWI), which aims to improve the convergence of FWI as well as to generate geologically plausible inversion results. Unlike conventional FWI, this method inverts for much fewer parameters in a sparse model space, while constraining the sparse inversion with structural features of the subsurface. In this thesis, the structural constraints are extracted from migrated seismic images, where the structural features could be apparent. This new IGFWI approach is implemented with a modified conjugate-gradient method that uses an image-guided gradient.
Figure 1.3: Two synthetic monochromatic seismograms with a period $T$ in (a) and (c) are compared with the recorded seismogram in (b) to illustrate the cycle-skipping problem in FWI. The synthetic monochromatic seismogram in (a) has a time delay larger than half a period, and in this case FWI will update the model such that the $(n+1)$th cycle in (a) matches the $n$th cycle in (b). Therefore, FWI produces an inaccurate model. The seismogram in (c) has a time delay smaller than half a period, and in this case FWI will update the model to make the $n$th cycle in (c) match the $n$th cycle in (b), leading to a correct model. Image courtesy of Virieux & Operto (2009).
In Chapter 3, I present a new quasi-Newton method to solve realistic-size FWI by explicitly constructing a projected Hessian matrix and its inverse matrix. This method is based on the idea of sparse-model inversion introduced in Chapter 2. With respect to fewer parameters in the sparse model, I can avoid the commonly used “limit-memory” approximation and use a projected BFGS (P-BFGS) method to explicitly compute and store a projected Hessian matrix and its inverse. A projected Hessian computed in this way saves both computational time and required memory and can be directly used to solve FWI with a quasi-Newton method, which converges faster in fewer iterations than conjugate-gradient methods.

In Chapter 4, I propose a new wave-equation reflection traveltime inversion (WERTI) to recover low-wavenumber components of velocity models, which help mitigate the cycle-skipping problem in reflection FWI with insufficient low frequencies. In WERTI, traveltime shifts between the recorded data and the synthetic data are estimated with dynamic warping, which solves a global optimization problem so that it can avoid cycle skipping. However, conventional methods, such as crosscorrelation, still suffer from cycle skipping in estimating traveltime shifts. I use WERTI to update the low-wavenumber background of the velocity model while using FWI to recover the high-wavenumber details (reflectors). Through an alternating combination of WERTI and FWI, this new approach can mitigate the velocity-depth ambiguity and therefore can successfully recover both the low- and high-wavenumber components with only high-frequency reflection data.

In Chapter 5, I apply IGFWI to a 2D ocean-bottom cable data set in order to test the methodology proposed in Chapter 2. Compared to conventional FWI, IGFWI with reflection data generates velocity models that make better geologic sense. With IGFWI-estimated velocities, RTM images show more interpretable coherent structures in geologically complex areas where, for instance, gas cloud is known to
Finally, I summarize the thesis and suggest some future work in \textbf{Chapter 6}.

Chapters 2–5 of the thesis have been published in, submitted to, or soon will be submitted to Geophysics:

- \textbf{Ma, Y.} and D. Hale, 2012, Qausi-Newton full-waveform inversion with a projected Hessian matrix: \textit{Geophysics}, 77, no. 5, R207-R216 (Chapter 3)
- \textbf{Ma, Y.} and D. Hale, 2013, Wave-equation reflection traveltime inversion with dynamic warping: to be submitted to \textit{Geophysics}, (Chapter 4)
- \textbf{Ma, Y.} and J. Yuan, Y. Shen, and B. Gong, Application of image-guided full waveform inversion to a 2D ocean-bottom cable data set: to be submitted to \textit{Geophysics} (Chapter 5)

I organize these articles in the thesis with slight modifications. Although this may create some redundancies, it improves readability of the thesis, so that one can read each chapter independently.

In addition to the publications listed above, my Ph.D. work also contributes to the following publications in journals and conventions:


• **Ma, Y.** and P. Sava, 2009, The effects of multi-scale heterogeneities on wave-equation migration: *Journal of Seismic Exploration*, 18, 357-383

• **Ma, Y.** and P. Sava, 2009, Effects of multi-scale velocity heterogeneities on wave-equation migration: *SEG Technical Program Expanded Abstracts*, 28, 2732-2736
2.1 Summary

Multiple problems, including high computational cost, spurious local minima, and solutions with no geologic sense, have prevented widespread application of full waveform inversion (FWI), especially FWI of seismic reflections. These problems are fundamentally related to a large number of model parameters and to the absence of low frequencies in recorded seismograms. Instead of inverting for all the parameters in a dense model, image-guided full waveform inversion (IGFWI) inverts for a sparse model space that contains far fewer parameters. In this chapter, we represent a model with a sparse set of values, and from these values, we use image-guided interpolation (IGI) and its adjoint operator to compute finely- and uniformly-sampled models that can fit recorded data in FWI. Because of this sparse representation, image-guided FWI updates more blocky models, and this blockiness in the model space mitigates the absence of low frequencies in recorded data. Moreover, IGI honors imaged structures, so image-guided FWI built in this way yields models that are geologically sensible.

2.2 Introduction

Full waveform inversion (FWI) (Tarantola, 1984) uses recorded seismic data \( d \) to estimate parameters of a subsurface model \( m \), by minimizing the difference between
recorded data $d$ and synthetic data $F(m)$, where $F$ is a forward operator that synthesizes data. All information in recorded seismic waveforms should, in principle, be taken into account in minimizing this difference. In other words, FWI comprehensively minimizes differences in traveltimes, amplitudes, converted waves, multiples, etc. between recorded and synthetic data. This all-or-nothing approach distinguishes FWI from other methods, such as traveltime tomography (Bishop et al., 1985; Luo & Schuster, 1991; Stork, 1988; Zelt & Barton, 1998), which focuses on only traveltime differences.

FWI is attractive in its capability to estimate a subsurface model with generally higher resolution (Operto et al., 2004) than traveltime tomography and migration velocity analysis (MVA) (Sava & Biondi, 2004; Yilmaz & Chambers, 1984). Another advantage of FWI over traveltime tomography or MVA is that FWI can estimate multiple parameters: density (Forgues & Lambaré, 1997), attenuation (Liao & McMechan, 1996), elasticity (Shi et al., 2007), anisotropy (Barnes et al., 2008), etc. Although FWI has a long history and promising benefits, two major obstacles – high computational cost and nonunique solutions – have prevented its widespread application in exploration seismology.

FWI requires large numbers of seismic wavefield simulations and reconstructions, and computational cost is proportional to the number of sources. FWI also requires multiple iterations to minimize data misfit, and computational cost is proportional to the number of required iterations. Therefore, various methods have been applied to reduce computational cost. One such method is to apply phase-encoding techniques (Krebs et al., 2009) that combine all shots together to form a simultaneous source. The cost of FWI using encoding techniques is thereby reduced by a factor ideally equal to the number of encoded shots divided by the number of recorded shots. To reduce the number of required iterations, one may use a sparse representation.
of a model space and reduce the number of model parameters. The velocity model estimated in FWI is densely sampled for simulating wave propagation. The number of dense samples is far beyond the number of samples necessary to geologically explain the model. In inversion, one therefore should choose as few samples as possible to construct a sparse model, while still maintaining as many geological features as possible. The wavelet transform is a representative technique used in inverse problems to reduce the number of parameters (Meng & Scales, 1996).

Because FWI is a typical underdetermined problem, many different models may yield synthetic data that match recorded data within a reasonable tolerance. This nonuniqueness problem is caused mainly by local minima in a data misfit function, and the presence of local minima is due to the fact that the forward operator $F$ is generally a nonlinear function of the model $m$. Strong nonlinearity in reflection FWI makes this local-minima problem more severe (Snieder et al., 1989). Cycle-skipping also causes nonunique solutions in FWI. Cycle-skipping occurs if the phase difference (time delay) between synthetic and recorded data is larger than half a period of the dominant wavelet. In practice, the cycle-skipping problem typically appears because it can be difficult to obtain an adequate initial model that is consistent with unrecorded low frequencies.

Both local-minima and cycle-skipping problems lead to models that poorly approximate the subsurface. To mitigate such problems, multiscale approaches (Boonyasiriwat et al., 2009; Bunks, 1995; Sirgue & Pratt, 2004) have been proposed. These methods recursively add higher-frequency details to models first computed from lower-frequency data. The fidelity of multiscale techniques depends fundamentally on the fidelity of low-frequency content in recorded data. In practice, the low frequencies required to bootstrap the multiscale approach may be unavailable. Other methods for addressing these problems have been proposed as well, e.g., inverting
high-wavenumber and low-wavenumber components separately (Hicks & Pratt, 2001; Snieder et al., 1989).

We solve these problems in a different way. Following Meng (2009), who proposes to use subsurface dips to constrain the inversion, we investigate the image-guided gradient (Ma et al., 2010) to complement low frequencies that are usually unavailable in recorded data. In this chapter, we propose image-guided sparse FWI, which aims to make FWI more efficient and more stable and to generate geologically sensible results. In image-guided FWI (IGFWI), we use image-guided interpolation (IGI) (Hale, 2009a) and its adjoint operator in order to apply structural constrains derived from migrated images. We first reformulate FWI in a sparse model space, by efficiently choosing sample points. We then solve the sparse FWI by using a modified image-guided conjugate-gradient method. This image-guided sparse FWI is tested on the Marmousi II model and with realistically high-frequency data.

2.3 Sparse-Model Full Waveform Inversion

Because the forward operator $F$ has no inverse $F^{-1}$ for almost any geophysical inverse problem, we cannot simply invert the model from the data using $m = F^{-1}(d)$. Instead, FWI is usually formulated as a least-squares optimization problem, in which we compute a model $m$ that minimizes the data misfit function

$$E(m) = \frac{1}{2}\|d - F(m)\|^2,$$

where $\| \cdot \|$ denotes an L2 norm.

We begin with an initial model $m_0$, which can be found using other inversion methods (e.g., traveltime tomography or migration velocity analysis); then we iteratively reduce the data misfit $E(m)$ by applying Newton-like methods. In the $i^{th}$ iteration, the Taylor series expansion of equation 3.1 about the model $m_i$ is
where $g_i \equiv g(m_i) = \frac{\partial E}{\partial m_i}$ denotes the gradient of the data misfit $E$ evaluated at $m_i$ and $H_i$ denotes the Hessian matrix comprised of the second-order partial derivatives of $E(m)$, again evaluated at $m_i$. If we ignore any term higher than the second order in equation 2.2, this Taylor series approximation is quadratic in the model perturbation $\delta m_i$, and we can minimize the data misfit $E(m_i)$ by solving a set of linear equations:

$$
H_i \delta m_i = -g_i 
$$

with a Newton method (Pratt et al., 1998) solution

$$
\delta m_i = -H_i^{-1}g_i .
$$

Unfortunately, the large size of the Hessian matrix $H_i$, which is directly determined by the number of parameters, prevents the application of Newton or Newton-like methods in realistic cases. Moreover, FWI is usually ill-posed in practice due to a typically large condition number of the Hessian matrix (Tarantola, 2005). A large condition number tends to appear especially when an inverse problem has a large number of model parameters in $m$. If the change of a model parameter in $m$ does not cause significant change in the data misfit function $E(m)$, the Hessian matrix $H$ will have a small (or nearly zero) eigenvalue. As a consequence, the condition number of the Hessian matrix will be large enough that the gradient-descent method (Vigh & Starr, 2008) used to solve an FWI problem converges slowly.

### 2.3.1 Inverse Problem in Sparse Model Space

Inspired by the intuitive relationship between the convergence rate and the number of model parameters, we pose an FWI problem that inverts for only a few model
parameters, to which the data misfit function is sensitive. We then reduce the condition number of the Hessian matrix and thereby the number of required iterations.

Following an approach similar to subspace methods (Kennett et al., 1988; Oldenburg et al., 1993) and the point collocation scheme (Pratt et al., 1998, Appendix A), we reconstruct a finely- and uniformly-sampled (dense) model $m$ from a sparse model $s$ that contains a much smaller number of model parameters than does the dense model $m$:

$$m = Rs,$$

(2.5)

where $R$ denotes a linear operator that interpolates model parameters from the sparse model to the dense model.

Differentiating both sides of equation 2.5, we have

$$\delta m = R \delta s.$$  

(2.6)

Then, substituting equation 2.6 into equation 2.4, we can reformulate the inverse problem posed in equation 2.4, with respect to a smaller number of model parameters in the sparse model $s$, as

$$H_i R \delta s_i = -g_i.$$  

(2.7)

Because $R$ is not a square matrix, equation 2.7 is different from conventional preconditioning (Benzi, 2002).

2.3.2 Solution in Sparse Model

One cannot solve equation 2.7 with a solution like $\delta s_i = -(H_i R)^{-1} g_i$ in the sparse domain $s$ because equation 2.7 is overdetermined; i.e., there are more equations than parameters. Therefore, we modify equation 2.7 to be

$$R^T H_i R \delta s_i = -R^T g_i,$$

(2.8)

and thereby obtain a least-squares solution for equation 2.7 in the sparse domain $s$:  

18
\[ \delta s_i = - \left( R^T H_i R \right)^{-1} R^T g_i , \]

where \( R^T \) is the adjoint operator of \( R \). This adjoint operator projects model parameters from the dense model \( m \) to the sparse model \( s \). Unfortunately, equation 2.9 is usually hard to implement because in practice the Hessian matrix is extremely expensive to compute and store.

Alternatively, the model update \( \delta s \) can be iteratively approximated by replacing the inverse of the projected Hessian matrix \( (R^T H_i R) \) with a scalar step length \( \alpha_i \):

\[ s_{i+1} = s_i - \alpha_i h_s^i , \]

where the conjugate direction \( h_s^i \) (Ma et al., 2010) is determined by

\[
\begin{align*}
    h_0^s &= R^T g_0 , \\
    \beta_i &= \frac{\left( R^T g_i \right)^T \left( R^T g_i - R^T g_{i-1} \right)}{\left( R^T g_{i-1} \right)^T R^T g_{i-1}} , \\
    h_s^i &= R^T g_i + \beta_i h_s^{i-1} .
\end{align*}
\]

In equation 2.10, the step length \( \alpha_i \) can be found with a line-search method (Nocedal & Wright, 2000). Equation 2.11 employs \( R^T g_i \) instead of \( g_i \), implying that equation 2.10 provides a solution for the FWI problem in the sparse domain \( s \). Because of fewer model parameters involved, the projected Hessian matrix \( (R^T H_i R) \) can become better-conditioned and thus equation 2.10 can require fewer iterations to converge to a solution model \( s \).

In reality, we need a dense update \( \delta m \) to compute synthetic data \( F(m) \) and to fit recorded data \( d \). For this reason, we apply the linear operator \( R \) to both sides of equation 2.10 and thereby interpolate the sparse model update \( \delta s_i \) to obtain the dense model update \( \delta m_i \):

\[ m_{i+1} = m_i - \alpha_i h_m^i , \]
where we compute the search direction $h^m_i$ by projecting the sparse conjugate direction $h^s_i$ to the dense domain:

$$h^m_0 = Rh^s_0 = RR^T g_0,$$

$$\beta_i = \frac{(R^T g_i)^T (R^T g_i - R^T g_{i-1})}{(R^T g_{i-1})^T R^T g_{i-1}},$$

$$h^m_i = RR^T g_i + \beta_i h^m_{i-1}. \quad (2.13)$$

Equation 2.13 provides a solution $m$ for FWI in the dense model space with the advantages derived from solving for $s$ in the sparse model space.

### 2.3.3 Implementation of Sparse-Model FWI

An implementation of sparse FWI based on conjugate gradients consists of five steps performed iteratively, beginning with an initial model $m_0$:

(i) compute the data difference $d - F(m_i)$;

(ii) compute the gradient $g_i, R^T g_i, RR^T g_i$, and the update direction $h^m_i$;

(iii) search for a step length $\alpha_i$;

(iv) update the model with $m_{i+1} = m_i + \delta m_i$.

(v) remigrate with the updated model and reselect the sparse model based on the remigrated image.

Each iteration of this sparse FWI is more expensive than one iteration of conventional FWI as steps (iv) and (v) bring additional cost. Fortunately, the cost of applying the interpolation and its adjoint operator or reselecting a sparse model is negligible compared to the cost of forward modeling or reverse time migration. Also, it is not necessary to reselect the sparse model in every iteration, and therefore by applying the last step only in selected iterations, one can further reduce the additional cost.
2.4 Choice of R

The operator $R$ can take different forms, including Fourier transform, wavelet transform, cubic splines, etc. In this chapter, we implement $R$ with image-guided interpolation (IGI) (Hale, 2009a) specifically because IGI accounts for imaged sub-surface structure. IGI uses structure tensors (Fehmers & Höcker, 2003; van Vliet & Verbeek, 1995) to guide interpolation of a few spatially scattered values, thereby making the interpolant conform to structure in images.

2.4.1 Image-Guided Interpolation

The input of IGI is a set of scattered data, a set

$$\mathcal{F} = \{f_1, f_2, ..., f_K\}$$

of $K$ known sample values $f_k \in \mathbb{R}$ that correspond to a set

$$\chi = \{x_1, x_2, ..., x_K\}$$

of $K$ known sample points $x_k \in \mathbb{R}^n$. Combining these two sets forms a space (e.g., our sparse model $s$), in which $\mathcal{F}$ and $\chi$ denote sample values and coordinates, respectively. The result of the interpolation is a function $q(x) : \mathbb{R}^n \rightarrow \mathbb{R}$, such that $q(x_k) = f_k$. Here, the dense model $m$ consists of all interpolation points $x$ and values $q(x)$.

Image-guided interpolation is a two-step process (Hale, 2009a):

$$R = QP,$$

where $P$ and $Q$ denote nearest neighbor interpolation and blending of nearest neighbors, respectively. Examples of applying IGI can be found in Hale (2009a). Appendix A describes in more detail the operators $P$ and $Q$. Intuitively, we can describe these two operators as:
1. \( P \): *scatters* values \( f_k \) from nearest sample points \( x_k \) to obtain uniformly sampled interpolated values;

2. \( Q \): *smoothes* the uniformly sampled nearest neighbor interpolant.

### 2.4.2 Adjoint Image-Guided Interpolation

Because \( Q^T = Q \) (Appendix A), we can configure the adjoint of image-guided interpolation as

\[
R^T = P^T Q^T = P^T Q. \tag{2.15}
\]

The adjoint operator \( R^T \) is again a two-step process:

1. \( Q^T = Q \): *smoothes* uniformly sampled values;

2. \( P^T \): *gathers* uniformly sampled values from nearest neighbors \( x \) to the scattered sample points \( x_k \).

In equation 2.13, we sequentially apply the IGI operator \( R \) and its adjoint operator \( R^T \) to produce the image-guided gradient \( RR^T g_i \). The adjoint operator \( R^T \) first gathers information to sample points from nearest neighborhoods that conform to structural features in the gradient \( g_i \). The IGI operator \( R \) then scatters the gathered information back to the same neighborhoods. Through this *gather-scatter* process, the image-guided gradient \( RR^T g_i \) generates low wavenumbers in models \( m \).

### 2.5 Structurally Constrained Sample Selection

A set of properly chosen locations for scattered samples is essential for implementing image-guided sparse FWI. The samples, which constitute a sparse model, should be representative, i.e., the chosen samples should cover all major structural features
and contain as little redundancy as possible. With samples that satisfy these two conditions, image-guided interpolation can reconstruct an accurate dense model $m$ from a sparse model $s$. Therefore, in general, one must:

- **locate samples between reflectors.** We put samples between reflectors rather than on reflectors, so that the gather-scatter process ($RR^T$) can produce low wavenumbers to fill areas between reflectors. The low wavenumbers will play an important role in image-guided FWI. Moreover, the sparse model should not miss important structures and therefore we need to put samples between every major reflector.

- **locate samples along geological features.** To reduce redundancy and better honor structural features, we put dense samples in structurally complex areas and sparse samples in structurally simple or coherent areas.

Figure 2.1(a) and Figure 2.1(b) show examples of uniform sample and pseudo-random sample selections, respectively. The uniform sample selection and the pseudo-random selection are easy to implement, however, neither of them can satisfy both of the above criteria. Given a fixed number of samples, the uniform samples fail to follow structural structures; many samples lie on reflectors, as shown in Figure 2.1(a), and those samples are undesirable. The pseudo-random selection has the same problem and creates samples that are too close, as shown in Figure 2.1(b). In this chapter, we investigate and then employ a structurally constrained sample selection scheme, which satisfies both criteria.

### 2.5.1 Distance Transform

A migrated image $I(x)$ (Figure 2.2(a)) can be considered as a combination of two parts: reflectors and areas delimited by reflectors. To choose samples between reflectors, we must first distinguish reflectors from the areas between them. For this
Figure 2.1: A uniform sample selection (a) and a pseudo-random sample selection (b). A total of 165 samples (white dots) are chosen from the densely sampled model space.
Figure 2.2: A migrated image $I(x)$ (a) and a distance map $d(x)$ of the migrated image (b). Zero distance indicates reflectors, and nonzero values indicate areas between reflectors.
purpose, we use a distance transform (DT) (Fabbri et al., 2008), which was first introduced in computer vision and image processing.

A distance transform computes for each pixel of an image the smallest distance to a given subset pixels. This given subset is a region of interest in the DT. Appendix B describes the distance transform in detail. For our sample selection problem, we treat reflectors as regions of interest and compute the distance from each sample in the migrated image to the nearest reflectors. Therefore, identifying reflectors is necessary before apply the distance transform to the migrated image. We determine reflectors by comparing pixel amplitudes of the migrated image with a percentile threshold. If the amplitude of a pixel is larger than the threshold, we assume that this pixel is on a reflector. Due to uneven amplitudes in the migrated image, this amplitude-thresholding scheme is performed within a window and then repeated for the entire image. Of course, horizon-picking algorithms (Bondár, 1992; Faraklioti & Petrou, 2004) can also be useful for identifying the reflectors.

Figure 2.2(b) shows a distance map \( d(\mathbf{x}) \), which illustrates, for a migrated image, how far each point is from the nearest reflector. As we can see from Figure 2.2(b), a large part of the distance map is nearly zero, which indicates reflectors. The remainder of the distance map shows areas between the reflectors, and a larger value implies a larger distance from reflectors.

If we choose only samples that have nonzero distances in Figure 2.2(b), the selection result will satisfy our first criteria. Normally, we choose the first sample at the location with the largest distance. To keep samples sparse, we avoid placing another sample in a nearby area surrounding the first sample. We refer to this area as an exclusion region, where no sample can be chosen.
2.5.2 Structurally Constrained Selection

The exclusion region can take different shapes. Figure 2.3 compares three types of rejection regions: a rectangle, a circle and a structure-constrained region. Neither the rectangle or the circle follows the structural features in the geological layers. These two types of exclusion regions cross the reflectors, and as a consequence it risks missing possible samples. To make the sample selection follow the second criteria, we use image structures to construct a structurally constrained exclusion region, which is shown as the read area in Figure 2.3. This region follows the structure and does not cross imaged reflectors.

Figure 2.3: A chosen sample (the white dot) in geological layers and three different exclusion regions: a rectangular, a circle, and a structurally constrained region (black). An exclusion region is where no a second sample appears.
To construct a structurally constrained exclusion region, we use a tensor field $D(x)$, as we do in the image-guided interpolation and its adjoint operator. Pseudo code for implementing the structurally constrained sample selection:

```
while $d(x) > 0$ for some $x$ do
    pick the sample $x_k$ with the largest $d(x)$
    solve equation A.1 for $t(x)$ and $t(x_k) = 0$
    find the structurally constrained region where $t(x) \leq t_0$
    exclude every other sample in that region by setting $d(x) = 0$, if $t(x) \leq t_0$
end while
```

Figure 2.4(a) shows tensor fields $D(x)$ computed for the migrated image, and Figure 2.4(b) shows a total of 165 samples picked with this structurally constrained selection procedure. According to our previously mentioned two criteria, Figure 2.4(b) shows a better distribution of samples than do the uniform and pseudo-random selections shown in Figure 2.1.

2.6 Example

To illustrate the feasibility of image-guided sparse-model FWI (IGFWI), we test the algorithm using the Marmousi II model (Bourgeois et al., 1991) and compare IGFWI results with conventional FWI results. We refer to the model in Figure 2.5(a) as the true model $m$. Figure 2.5(b) displays the initial model $m_0$ that we used for inversion; except for the water layer, which is the same, $m_0$ is a highly smoothed version of the true model $m$. In this example, we employ 11 evenly distributed shots on the surface, and a Ricker wavelet with a peak frequency of 12 Hz is used as the source for simulating wavefields. The source and receiver intervals are 0.76 km and 0.024 km, respectively; the maximal time is 2.9 s, with a maximal offset of about 4.2 km. The effective frequency band that we can invert in the data is approxi-
Figure 2.4: A metric tensor field $D(x)$ illustrated by ellipses (a) and structurally constrained sample selection (b). A total of 165 samples (white dots) are chosen for sparse-model FWI.
mately 3 – 27 Hz, and the low and high ends are 20 dB below the peak frequency in the spectrum, respectively. In either conventional FWI or image-guided FWI, we apply time-domain approach with no preconditioning, which is equivalent to invert the entire frequency band simultaneously. In this example, the same nonlinear conjugate-gradient method and the same quadratic line-search algorithm are used in both conventional FWI and the image-guided FWI. The difference is that the nonlinear conjugate-gradient method (equation 13) is adopted to fit the sparse model in the image-guided FWI.

We first create data \( d = F(m) \) using the true model \( m \). Henceforth, for consistency with the previous discussion, we refer to these data as the recorded data, even though we compute these noise-free data using a finite-difference solution to a 2D acoustic constant-density wave equation. For example, Figure 2.6(a) shows a common-shot gather for shot number 1 of the recorded data \( d \). Because the maximal offset is about 4.2 km, no significant refraction energy appears in the data. For this example, we therefore primarily use reflection data in the inversion. Figure 2.6(b) shows the corresponding synthetic data \( F(m_0) \) computed for the initial model \( m_0 \). Figure 2.6(c) displays the data residual \( d - F(m_0) \), which is a part of the data that cannot be explained by the initial model. In the four steps of image-guided FWI, computation of this data residual is step (i).

In step (ii), we compute the gradient of the data misfit. As discussed in Tarantola & Valette (1982), this gradient is equal to the output of reverse time migration (RTM) applied to the data residual, using the current model \( m_0 \). This implementation of the gradient is also referred to as the adjoint-state method (Tromp et al., 2005). Figure 2.7(a) shows the gradient \( g_0 \) computed in this way for the first iteration.

Also in step (ii), we compute the image-guided gradient. To obtain this gradient, one must know the tensor field \( D(x) \) that corresponds to the structure of the subsur-
Figure 2.5: The Marmousi II velocity model (a) and the initial velocity (b) for FWI, which is a highly smoothed version of the true velocity.
face. We obtain this metric tensor field \( \mathbf{D} (\mathbf{x}) \) from the migrated image. Figure 2.4(a) displays ellipses that illustrate tensors for a migrated image, which is a RTM result of recorded data \( \mathbf{d} \) with the initial model \( \mathbf{m}_0 \). We must also choose a set of sparse sample points. In this example, we employ the structurally constrained sample selection scheme and automatically pick a total of 165 samples, as depicted by the dots in Figure 2.4(b). Figure 2.7(b) shows the image-guided gradient \( \mathbf{RR}^T \mathbf{g}_0 \) computed in this way for the 1st iteration of image-guided FWI.

In step (iii), we use a quadratic line-search algorithm to compute a step length \( \alpha_0 \). The search direction \( \mathbf{h}_0^m \) is computed using conjugate gradients in equation 2.13, but for this first iteration is simply the image-guided gradient.

Finally, in step (iv), we update the current velocity model according to equation 2.12. Figure 2.8(b) shows the updated velocity after 10 iterations of image-guided FWI. For comparison, Figure 2.8(a) shows the updated velocity after 10 iterations of conventional FWI without the application of the image-guided technique.

### 2.7 Discussions

Due to the gather-scatter operation \( \mathbf{RR}^T \), the image-guided gradient \( \mathbf{RR}^T \mathbf{g}_0 \) shown in Figure 2.7(b) contains significantly more low-wavenumber contents than the original gradient \( \mathbf{g}_0 \) in Figure 2.7(a). In addition to low-wavenumber components, the image-guided gradient preserves the structural features in the model. Therefore, the velocity estimated by image-guided FWI shows coherent structures and makes better geological sense, as shown in Figure 2.8(b). Compared to the conventional FWI estimation (Figure 2.8(a)), the image-guided FWI (Figure 2.8(b)) shows less crossing artifacts that make no geological sense. Meanwhile, with more accurate velocity values and reservoir depth, the image-guided FWI better recovers the Marmousi model, especially in the deep part of the model.
Figure 2.6: A shot gather from the recorded data $d$ (a), a shot gather from the synthetic data $F(m_0)$ (b), and the initial data residual $d - F(m_0)$ (c).
Figure 2.7: The gradient $g_0$ (a) and the image-guided gradient $\mathbf{R}^T \mathbf{R} g_0$ (b).
Figure 2.8: Estimated velocity models with the conventional FWI (a) and the image-guided FWI (b), both in the 10th iteration.
Image-guided FWI, with the application of the image-guided gradient, also converges faster than does conventional FWI. After 10 iterations, the data residual for image-guided FWI is significantly smaller than the residual for conventional FWI, as shown in Figure 2.9.

Moreover, Figure 2.10(a) compares the data misfit functions of image-guided FWI and conventional FWI, and image-guided FWI converges in fewer iterations. Although one never know the true model in practice, for the synthetic study in this chapter, it is worthwhile to compare the model misfit function (an L2 norm of the difference between the true model and the estimated model) as well. Figure 2.10(b) shows the model misfit as a function of iteration number. Interestingly, but unsurprisingly, the model misfit function of image-guided FWI decreases much faster than conventional FWI.

Due to the lack of low frequencies in the recorded data, conventional FWI in this Marmousi example cannot reduce the model misfit significantly, even though the data misfit decreases monotonically. In contrast, the model misfit of IGFWI decreases significantly because the lack of low frequencies is mitigated. To demonstrate this mitigation, we analyze and compare four vertical velocity profiles, which are extracted from the true model (Figure 2.5(a)), the initial model (Figure 2.5(b)), the FWI model (Figure 2.8(a)) and the IGFWI model (Figure 2.8(b)), respectively, all at the horizontal distance of 2.5 km. Figure 2.11(a) plots these velocity profiles; Figure 2.11(b) compares the same profiles in the wavenumber domain ranging from 1 km$^{-1}$ to 6 km$^{-1}$, which is far below the Nyquist wavenumber (62.5 km$^{-1}$). Figure 2.11(b) does not compare these velocity profiles below 1 km$^{-1}$ because, between 0 and 1 km$^{-1}$, the smooth background model dominates and therefore all four curves look similar. Within this low-wavenumber range between 1 km$^{-1}$ and 6 km$^{-1}$, the FWI velocity, as shown in Figure 2.11(b), does not add significant information to the
initial velocity. On the other hand, IGFWI gains significantly and therefore approximates the true velocity much better. For example, notice the significant change cause by IGFWI at 2.5 km$^{-1}$. For a velocity of 2.5 km/s, 2.5 km$^{-1}$ corresponds to 1 Hz, which is a significantly low frequency compared to the peak frequency (12 Hz). In other words, the lack of low frequencies in the data domain is mitigated by IGFWI in the space domain with additional low wavenumber components. This mitigation is more meaningful when the multi-scale approach is inapplicable; this inapplicability commonly occurs because low frequencies are often contaminated by noise and therefore unavailable in the recorded data.

![Data residual for one shot after 10 iterations of conventional FWI (a) and image-guided FWI (b).](image)

Figure 2.9: The data residual for one shot after 10 iterations of conventional FWI (a) and image-guided FWI (b).

Sparse sources are partially responsible for artifacts in the shallow parts of the model in conventional FWI (Figure 2.8(a)). However, in the deep reservoir area (for example, 2.5 – 3 km in depth, 4 – 7 km in distance), artifacts are not primarily due to the sparse source locations but should be caused by velocity errors. These crossing artifacts in the reservoir are geologically nonsensible and therefore are misleading in interpretation. In IGFWI (Figure 2.8(b)), these geologically nonsensible artifacts
Figure 2.10: Convergence of between image-guided FWI and conventional FWI: the data misfit function (a) and the model misfit function (b).
Figure 2.11: Comparison of vertical velocity profiles at distance = 2.5 km (a) and the corresponding wavenumber-domain representation (b).
are removed. While eliminating geologically nonsensible artifacts, IGFWI provides a slightly lower resolution result compared with conventional FWI because, in fact, both IGI and its adjoint operators partially perform as smoothing filters. This smoothness may weaken some interesting sharp features in the estimated model, for example, the fault planes in Figure 2.8(b). To solve this problem, we feed the IGFWI velocity in Figure 2.8(b) as an initial model into conventional FWI, and we then obtain a new velocity model as shown in Figure 2.12, which preserves the sharp fault planes and still makes good geologic sense.

Figure 2.12: Using conventional FWI to enhance sharp boundaries such as fault planes. The IGFWI velocity in Figure 2.8b is used as an initial model in conventional FWI to produce this result.

In more realistic 3D cases, model parameters generally vary more gently in the inline–crossline direction than the depth–inline direction, which provides more sparsity in 3D than 2D. As a result, IGFWI, which actually is a sparse model inversion, can be extended to 3D with potentially more significant advantages.
Recorded data is better described by an elastic model. Therefore, it is difficult to accurately match the amplitudes of synthetic acoustic data with recorded elastic data, especially at far offsets, where elastic effects may not be negligible (Mulder & Plessix, 2008). However, because recorded elastic data in general is dominated by P-waves, it is a common approach to apply acoustic inversion to real elastic data (Vigh et al., 2009), in order to avoid prohibitive costs of 3D elastic modeling and imaging. When acoustic IGFWI is applied to real elastic data, problems related to the mismatched amplitudes may arise. Therefore, future development of elastic IGFWI may be necessary.

2.8 Conclusions

In this chapter, we have proposed image-guided sparse FWI, which inverts for subsurface parameters in a sparse model space. Image-guided sparse FWI is implemented with a modified conjugate-gradient method that employs an image-guided gradient. We also proposed an efficient way to select sparse model sample locations in a structurally constrained fashion.

We test our method on the Marmousi model. By using image-guided interpolation and its adjoint operator, we construct an image-guided gradient that mitigates the lack of low frequencies in the recorded data, and thereby improve both inversion speed and quality. Because structural features in images are considered, models updated by image-guided FWI are more sensible geologically.
3.1 Summary

We present a method, in realistic-size full waveform inversion (FWI), to explicitly construct a projected Hessian matrix and its inverse matrix, which we subsequently use to solve FWI with a quasi-Newton method. Newton’s method is practically unfeasible in solving realistic-size FWI problems, simply because of the prohibitive cost (both computing time and memory consumption) of calculating the Hessian matrix and the inverse Hessian. Therefore, the Gauss-Newton method and various quasi-Newton methods are proposed to approximate a Hessian matrix. Particularly, current quasi-Newton FWI (QNFWI) commonly uses the limited-memory BFGS (L-BFGS) method, which however only implicitly approximates an inverse Hessian. In this chapter, we repose FWI as a sparse optimization problem in a sparse model space, which contains substantially fewer model parameters that are constrained by structures of the model. With respect to fewer parameters in the sparse model, we can avoid the “limited-memory” approximation and are able to explicitly compute and store a projected Hessian matrix that saves both the computational time and required memory. We construct such a projected Hessian matrix by adapting the classic BFGS method to a projected BFGS (P-BFGS) method in the sparse space. Using the

---

1 Center for Wave Phenomena, Colorado School of Mines
2 Corresponding author
projected Hessian matrix and its inverse, we can apply the P-BFGS method to solve FWI with a quasi-Newton method. In QNFWI with P-BFGS, because we invert for a sparse model with much fewer parameters, the memory required to compute the projected Hessian is negligible compared to either forward modeling or gradient calculation. QNFWI with P-BFGS converges in fewer iterations than conjugate-gradient based methods and QNFWI with L-BFGS.

3.2 Introduction

Full waveform inversion (FWI) is usually formulated as an optimization problem (Tarantola, 1984), in which we consider the minimization of a multivariate nonlinear objective function \( E: \mathbb{R}^N \rightarrow \mathbb{R} \),

\[
\min_{m \in \mathbb{R}^N} E(m),
\]

where the function variable \( m \) denotes a model space that consists of \( N \) parameters, such as seismic velocities, for an earth model. The objective function for FWI can have various forms (Crase et al., 1990), e.g., L2 norm (Tarantola, 1984), L1 norm (Loris et al., 2007), and Huber norm (Guitton & Symes, 2003). In this chapter, we choose a least-squares form: \( E(m) = \frac{1}{2} \| d - F(m) \|^2 \), where \( \| . \| \) denotes an L2 norm, \( d \) denotes the recorded data, and \( F \) is a forward data-synthesizing operator, for example, a wave equation, which generally is a nonlinear function of the model \( m \).

Various iterative methods (Nocedal & Wright, 2000; Tarantola, 2005) for solving this least-squares optimization problem include the conjugate gradient method, Newton’s method, the Gauss-Newton method, and various quasi-Newton methods. Compared to the conjugate gradient method, Newton-like methods generally converge faster in fewer iterations (Pratt et al., 1998). However, Newton’s method is rarely used to solve realistic seismic full waveform problems, because for such large problems
it is costly. In fact, conventional Newton’s method is suitable only for solving small-
or medium-size optimization problems (Pratt et al., 1998; Sheen et al., 2006; Virieux & Operto, 2009), in which the number \( N \) of model parameters ranges from hundreds to thousands.

Newton’s method must compute the gradient \( \mathbf{g}(\mathbf{m}) \equiv \frac{\partial E}{\partial \mathbf{m}} \) and the Hessian \( \mathbf{H}(\mathbf{m}) \equiv \frac{\partial^2 E}{\partial \mathbf{m}^2} \). The Hessian matrix, in particular, consists of a large number \( \mathcal{O}(N^2) \) of second-order derivatives of the objective function \( E(\mathbf{m}) \) with respect to model parameters in \( \mathbf{m} \). To compute each second-order derivative as the definition, one must solve one forward problem \( \mathbf{F}(\mathbf{m}) \). Therefore, the Hessian matrix computed in this way requires the solution of \( \mathcal{O}(N^2) \) forward problems. Pratt et al. (1998) show a more efficient method that reduces the number of required forward solutions from \( \mathcal{O}(N^2) \) to \( \mathcal{O}(N) \), which, however, is still too costly in realistic large-size FWI problems. Moreover, the Hessian matrix consumes large amounts of memory. For a \( N \)-parameter model, a Hessian matrix stored in single precision requires \( 4N^2 \) bytes of memory. In addition, one must, in each iteration of Newton’s method, find an update direction by solving a linear set of \( N \) equations, which requires an inverse Hessian is an \( \mathcal{O}(N^3) \) operation.

Instead of computing the Hessian \( \mathbf{H} \) explicitly, Santosa & Symes (1988) and Fichtner & Trampert (2011a) extend the adjoint state methods (Tromp et al., 2005) to compute the action of the Hessian on an arbitrary vector, e.g., \( \mathbf{H}\delta\mathbf{m} \), where \( \delta\mathbf{m} \) is a model update. Fichtner & Trampert (2011a) point out that the cost for computing Hessian kernels (the volume density of \( \mathbf{H}\delta\mathbf{m} \)) is only twice as much as that for Fréchet kernels, which is substantially lower than the cost of explicitly computing a Hessian. However, Fichtner & Trampert (2011a) also mention that computing \( \mathbf{H}\delta\mathbf{m} \) is still costly (\( \mathcal{O}(N^3) \)) and the high cost makes the Hessian kernels less applicable in practice. Moreover, an inverse Hessian is required in Newton-like methods, and the idea of Hessian kernels cannot help.
Because of the high cost of Newton’s method for large problems, various methodologies have been proposed to approximate the Hessian matrix. The Gauss-Newton method, for example, ignores the second-order terms that account for nonlinearity in the function $F(m)$. This approximation is valid only when the data-synthesizing operator $F$ is a linear or quasi-linear function of $m$. If the initial model in FWI is far from the true model and nonlinearity is significant, the Gauss-Newton method may only converge to local minima.

Quasi-Newton methods do not compute the true Hessian matrix directly, but instead iteratively update Hessian approximations. One can apply various methods (Nocedal & Wright, 2000), such as the BFGS method, the DFP method, the SR1 method, and the Broyden class, to approximate a Hessian matrix. The BFGS method, which is named after the four inventors Broyden (1970), Fletcher (1970), Goldfarb (1970), and Shanno (1970), is an efficient and the “most popular” (Nocedal & Wright, 2000) way to iteratively approximate the Hessian matrix. However, although the BFGS method reduces the computation time required to approximate a Hessian matrix, it does not decrease the amount of memory required to store Hessian approximations, nor does it reduce the computation time required to use the Hessian matrix to update model parameters.

In practical applications of quasi-Newton FWI (QNFWI), one must reduce both computation time and memory consumption significantly. The limited-memory BFGS (L-BFGS) (Liu & Nocedal, 1989; Nocedal, 1980) algorithm, for instance, does not explicitly compute or store any type of Hessian matrix. Instead, L-BFGS only stores information, such as model changes and gradient changes, from a limited number $M$ (often $M \leq 10$) of previous iterations. L-BFGS then uses the stored information to implicitly form an inverse of the approximated Hessian and obtain a search direction for the next iteration. In this way, memory required in L-BFGS is much less than
Examples of QNFWI with L-BFGS can be found in Brossier et al. (2010); Guitton et al. (2012); Virieux & Operto (2009). L-BFGS maintains a history of changes in the model and the gradient; although this history is often less than 10 iterations, L-BFGS can still get expensive in 3D. Eigen-decomposition of a Hessian matrix provides eigenvalues and eigenvectors, which are useful for sensitivity analysis in FWI (Fichtner & Trampert, 2011b). Unfortunately, L-BFGS never explicitly forms any Hessian matrix, and hence, in QNFWI with L-BFGS, we are not able to perform the sensitivity analysis by applying the eigen-decomposition to the Hessian.

In this chapter, we aim to explicitly construct an approximated Hessian matrix for realistic-size FWI problems. We pose FWI as a sparse optimization problem with respect to fewer parameters in a sparse model space and then compute a projected Hessian matrix in the sparse space. Various bases (e.g., Fourier bases, wavelet bases, and splines) have been used, typically in tomographic studies, to obtain some types of sparse models. Among those methods, the representative one is the wavelet transform (Loris et al., 2007; Meng & Scales, 1996; Simons et al., 2011). In this chapter, we construct a sparse space in a different way by considering the inherent structural sparsity of the model. In a model estimated with FWI, the number of dense samples is far beyond the number of necessary samples to geologically soundly explain the model. In FWI, one therefore should invert for a sparse model that contains as few samples as possible, while still maintaining as good geological sense as possible. Structural or geological information can also be introduced as regularization or precondition in tomography (Clapp et al., 2004) and FWI (Guitton et al., 2012). In the sparse model space, a projected Hessian matrix with respect to only a few parameters can significantly reduce both computation time and memory consumption. As a result, we need not to use a “limited-memory” approximation (e.g., L-BFGS) of the BFGS method. Instead, we directly obtain the projected Hessian matrix by apply-
ing a projected BFGS (P-BFGS) method, a projected version of the original BFGS method in the sparse model space. Unlike the L-BFGS method, the P-BFGS method computes and stores the projected Hessian explicitly. Therefore we can analyze the eigen-decomposition of a projection Hessian and can further explore the sensitivity analysis, which reveals the relationship between eigenvectors and the model that needs to be recovered by FWI.

This chapter is organized as follows. We first briefly review various Newton-like methods. We then explain how to construct a projected Hessian in a sparse model space. We finally apply the projected Hessian in time-domain QNFWI. Tests of our projected Hessian matrix on the Marmousi II model (Martin et al., 2006) suggest that quasi-Newton methods like ours may be promising in practical applications of FWI.

3.3 Newton-Like Methods

We consider only iterative methods for solutions of the nonlinear FWI optimization problem posed in equation 3.1. In the \( i \)th iteration of such methods, we update the model \( \mathbf{m} \) in the direction of a vector \( \mathbf{p}_i \):

\[
\mathbf{m}_{i+1} = \mathbf{m}_i + \alpha_i \mathbf{p}_i , \tag{3.2}
\]

for some step length \( \alpha_i \). Newton’s method, the Gauss-Newton method, and quasi-Newton methods differ in the way in which they compute and use the update vector \( \mathbf{p}_i \).

3.3.1 Newton’s Method

In Newton’s method, the update vector \( \mathbf{p}_i \) is determined by first ignoring the higher-order (> 2) terms in a Taylor series approximation of the objective function \( E(\mathbf{m}) \):
\[ E(m_i + \delta m_i) \approx E(m_i) + \delta m_i^T g_i + \frac{1}{2} \delta m_i^T H_i \delta m_i , \] 

(3.3)

where \( E(m_i) \) denotes the objective function evaluated at \( m_i \), \( g_i = g(m_i) \), and \( H_i = H(m_i) \). The update direction in Newton’s method then is (Pratt et al., 1998)

\[ p_i = -H_i^{-1} g_i , \] 

(3.4)

and the step length \( \alpha_i = 1 \). The Newton method, in practice, is often adapted to a damping version, written as \( p_i = -(H_i + \lambda I)^{-1} g_i \), where \( \lambda \) is a damping factor. However, for models with large numbers of parameters, high costs for computing the Hessian \( H_i \) and its inverse \( H_i^{-1} \) prevent the application of Newton’s method.

### 3.3.2 Gauss-Newton Method

One can explicitly write each element of the Hessian as

\[ H_{jk} = H_{kj} = J_j^T J_k + \left( \frac{\partial^2 F(m)}{\partial m_j \partial m_k} \right)^T (d - F(m)) , \] 

(3.5)

where \( J_j \) and \( J_k \) are the \( j \)th and \( k \)th columns in the Jacobian matrix \( \{ \frac{\partial F(m)}{\partial m_1}, \frac{\partial F(m)}{\partial m_2}, ..., \frac{\partial F(m)}{\partial m_N} \} \), respectively. Discarding the second-order term in equation 3.5, which accounts for the nonlinearity in wave propagation and the nonlinearity in the objective function, we get an approximated Hessian \( H_a \), with elements

\[ H_{jk} \approx J_j^T J_k . \] 

(3.6)

Using this approximation we obtain the Gauss-Newton method with update direction

\[ p_i = -H_a^{-1} g_i . \] 

(3.7)

Pratt et al. (1998) show an approach for calculating this approximated Hessian \( H_a \), with a cost of solving \( N \) forward problems \( F(m) \). However, by ignoring the second-order term, the approximation \( H_a \) in effect assumes that the function \( F(m_i + \delta m_i) \) is linear with respect to the difference \( \delta m_i \) between the true model \( m \) and the current model \( m_i \). In practice this assumption is seldom realistic.
3.3.3 Quasi-Newton Method

A quasi-Newton method iteratively approximates the Hessian matrix. The BFGS method is currently considered the “most popular” (Nocedal & Wright, 2000) quasi-Newton update formula. The basic idea of BFGS is to update the Hessian using changes in both the model \( m_i \) and the gradient \( g_i \), from one iteration to the next:

\[
H_{i+1} = H_i + \frac{y_i y_i^T}{y_i^T \delta m_i} - \frac{H_i \delta m_i (H_i \delta m_i)^T}{\delta m_i^T H_i \delta m_i},
\]

(3.8)

where \( y_i = g_{i+1} - g_i \), \( \delta m_i = m_{i+1} - m_i \).

Pseudo-code for implementing the BFGS method is as follows (Nocedal & Wright, 2000):

```plaintext
given \( m_0, \ g_0 = \frac{\partial E}{\partial m_0} \), and \( H_0 = I \)

for \( i = 0, 1, 2, \ldots \), until convergence do

\[ p_i = -H_i^{-1}g_i \]

search for \( \alpha_i, \delta m_i = \alpha_i p_i \)

\[ m_{i+1} = m_i + \delta m_i \]

\[ g_{i+1} = \nabla E(m_{i+1}) \]

\[ y_i = g_{i+1} - g_i \]

\[ H_{i+1} = H_i + \frac{y_i y_i^T}{y_i^T \delta m_i} - \frac{H_i \delta m_i (H_i \delta m_i)^T}{\delta m_i^T H_i \delta m_i} \]

end for
```

The BFGS method is an efficient way to compute Hessian approximations. In each iteration, BFGS performs a line search to update the current model \( m_i \), which ideally costs only 1 forward modeling \( F(m) \). In addition, BFGS requires computation of a gradient \( g_i \), with a cost of only 2 forward modelings \( F(m) \). Unlike the Gauss-Newton method, BFGS does not simply ignore the second-order terms in the Hessian. Instead, the BFGS method uses differences in models and gradients between iterations.
to construct a complete Hessian approximation.

Unfortunately, the BFGS method does not reduce the $O(N^2)$ memory required to store the Hessian approximation, nor does it reduce the $O(N^3)$ cost of solving the set of linear equations (equation 3.4) for the update vector $p_i$.

### 3.4 Projected Hessian Matrix

The only way to simultaneously reduce both computational time and memory for the Hessian computation is to reduce the number $N$ of model parameters in the dense model $m$. For this purpose, we reformulate FWI as a sparse optimization problem in a sparse model space $s$, which contains a much smaller number $n << N$ of model parameters. With respect to the $n$ parameters in the sparse model, we compute a projected Hessian matrix.

#### 3.4.1 Sparse Optimization

In an approach similar to that used in subspace methods (Kennett et al., 1988; Oldenburg et al., 1993) and alternative parameterizations (Pratt et al., 1998, Appendix A), we construct a finely- and uniformly-sampled (dense) model $m$ from a sparse model $s$:

$$m = Rs$$  \hspace{1cm} (3.9)

where the linear operator $R$ can be regarded as an interpolation operator. FWI is then reformulated as a new sparse optimization problem, in which we minimize a new nonlinear objective function $E: \mathbb{R}^n \rightarrow \mathbb{R}$,

$$\min_{s \in \mathbb{R}^n} E(s) .$$  \hspace{1cm} (3.10)

We can rewrite the objective function in equation 3.3, with respect to parameters in the sparse model $s_i$, as

$$E(s_i + \delta s_i) \approx E(s_i) + \delta s_i^T \tilde{g}_i + \frac{1}{2} \delta s_i^T \tilde{H}_i \delta s_i ,$$  \hspace{1cm} (3.11)
where we refer to \( \tilde{g}_i \equiv R^T g_i \) as a *projected gradient*, and \( \tilde{H}_i \equiv R^T H_i R \) as a *projected Hessian* matrix (Gill *et al.*, 1981). Here \( R^T \) denotes the adjoint of the interpolation operator \( R \). Therefore, in Newton’s method the update direction vector becomes

\[
\tilde{p}_i = -\tilde{H}_i^{-1} \tilde{g}_i ,
\]

for a step length \( \alpha_i = 1 \).

The projected Hessian \( R^T H_i R \) is a \( n \times n \) symmetric matrix, where memory consumption is reduced from \( O(N^2) \) to \( O(n^2) \). Computation in equation 3.12 is likewise reduced from \( O(N^3) \) to \( O(n^3) \). Figure 3.1 describes this projected Hessian matrix schematically.

![Figure 3.1: Schematic for computing a projected Hessian \( \tilde{H}_i \). The projected Hessian reduces computation time and memory consumption.](image)

Through the optimization problem posed in equation 3.10, model parameters in the dense model \( m \) are not determined independently as in equation 3.1. Instead, the \( n \) parameters in the sparse model \( s \) constrain the other \( N - n \) parameters in \( m \). This constraint is imposed by the operator \( R \). Therefore, the optimization problem in equation 3.10 is equivalent to a linearly constrained optimization problem in the
dense model space \( \mathbf{m} \), where the number of constraints is \( N - n \). In the context of constrained optimization, the projected Hessian matrix is suggested by Gill et al. (1981) and several other authors.

### 3.4.2 \( \mathbf{R} \) and \( \mathbf{R}^T \)

In tomographic studies, methods such as splines, Fourier transform (Fichtner & Trampert, 2011b), and the wavelet transform (Loris et al., 2007; Meng & Scales, 1996; Simons et al., 2011) are used to perform sparse inversions. Nevertheless, these methods are not good options for our purpose of constructing the projected Hessian in a sparse optimization. All of them simply reparameterize a dense model in different bases, without introducing any constraints to make the sparse inversion be a constrained optimization problem, which, however, is the origin of the projected Hessian. In particular, none of those methods takes the structural information of the subsurface into consideration, and as a result, those methods may generate results that make little geologic sense.

We implement the operator \( \mathbf{R} \), in this study, with image-guided interpolation (IGI) (Hale, 2009a) because IGI uses structural information as constraints between the sparse model and the dense model. IGI includes two sequential steps: \( \mathbf{R} = \mathbf{Q}\mathbf{P} \), where \( \mathbf{P} \) and \( \mathbf{Q} \) are nearest neighbor interpolation and blended neighbor interpolation, respectively. Intuitively, we can describe these two steps as:

1. **\( \mathbf{P} \):** interpolates values from nearest sample points in a sparse model \( \mathbf{s} \) to obtain uniformly sampled values in a dense model \( \mathbf{m} \);
2. **\( \mathbf{Q} \):** smooths the uniformly sampled nearest neighbor interpolant.

The simplest nearest neighbor interpolation sets the value of an interpolation point equal to the value of a known sample point, which is the nearest in the Eu-
clidian distance. In IGI, Hale (2009a) uses a non-Euclidean distance, which honors the structural features in the interpolation area. The non-Euclidean distance says that the distance is small in structurally coherent directions and is large in directions where structural features change rapidly. The blended neighbor interpolation then smoothes the interpolant more in structurally coherent directions and less in structurally incoherent directions. Therefore, IGI interpolates (scatters) values of known samples in a sparse model $s$ to a uniformly sampled dense model $m$, and the interpolant can make good geological sense because it accounts for structures in the model (Hale, 2010).

Because the two steps of IGI can be explicitly written as two sequential matrix-vector operations (Hale, 2009a), the adjoint operator $R^T = P^TQ^T$ can be directly obtained with simple linear algebra. Different from the scattering action in $R$, the adjoint operator $R^T$ gathers values from the dense model $m$ to the known samples in the sparse model $s$. Appendix A explains the IGI operator $R$ and its adjoint operator $R^T$ in detail.

### 3.4.3 Projected BFGS Method

A projected Hessian matrix can be achieved iteratively with an approach similar to the classic BFGS method. Apply the interpolation operator $R$ and its adjoint operator $R^T$ to both sides of equation 3.8 to obtain

$$
\tilde{H}_{i+1} = \tilde{H}_i + \frac{\tilde{y}_i \tilde{y}_i^T}{\bar{y}_i^T \delta s_i} - \frac{\tilde{H}_i \delta s_i \left( \tilde{H}_i \delta s_i \right)^T}{\delta s_i^T \tilde{H}_i \delta s_i},
$$

where $\tilde{y}_i \equiv R^T y_i = \tilde{g}_{i+1} - \tilde{g}_i$. We further simplify equation 3.13 with $\delta s_i = \alpha_i \tilde{p}_i$ and $\tilde{H}_i \delta s_i = -\alpha_i \tilde{g}_i$ (Gill et al., 1981):

$$
\bar{H}_{i+1} = \bar{H}_i + \frac{\tilde{y}_i \tilde{y}_i^T}{\bar{y}_i^T \tilde{p}_i} + \frac{\tilde{g}_i \tilde{g}_i^T}{\bar{p}_i \tilde{g}_i}.
$$

54
Equation 3.13 has the same structure as equation 3.8, so we refer to this formula for updating the projected Hessian as the projected BFGS (P-BFGS) method, a version of the classic BFGS method in the sparse model space. Pseudo-code for implementing this P-BFGS method is similar to the code for BFGS, as follows:

```
given \( s_0, \tilde{g}_0 = R^T g_0 \), and \( \tilde{H}_0 = I \)

\( \text{for } i = 0, 1, 2, ..., \text{ until convergence } \text{ do} \)

\( \tilde{p}_i = -\tilde{H}_i^{-1} \tilde{g}_i \)

search for \( \alpha_i, \delta s_i = \alpha_i \tilde{p}_i \)

\( m_{i+1} = m_i + R \delta s_i \)

\( g_{i+1} = \nabla E (m_{i+1}), \tilde{g}_{i+1} = R^T g_{i+1} \)

\( \tilde{y}_i = \tilde{g}_{i+1} - \tilde{g}_i \)

\( \tilde{H}_{i+1} = \tilde{H}_i + \frac{\tilde{y}_i \tilde{y}_i^T}{\alpha_i \tilde{y}_i^T \tilde{p}_i} + \frac{\tilde{g}_{i+1} \tilde{g}_{i+1}^T}{\tilde{p}_i^T \tilde{g}_i} \)

end for
```

Notice that in every iteration of the P-BFGS method, we not only construct a projected Hessian, but also update the model. Therefore, the P-BFGS method can be directly used in FWI as a quasi-Newton method that employs the projected Hessian and its inverse Hessian. In FWI, a dense model \( m \) is necessary for simulating wave propagation. Therefore, the outlined P-BFGS algorithm interpolates the sparse model update \( \delta s \) and updates a dense model with \( m_{i+1} = m_i + R \delta s_i \). Although \( m_{i+1} \) gives a solution to QNFWI in the dense model space, it in fact attains advantages of the sparse optimization.

### 3.4.4 Wolfe Conditions

The projected Hessian matrix \( \tilde{H} \) is symmetric positive-definite (SPD) if \( \tilde{y}_i^T \delta s_i > 0 \) in equation 3.13 (Nocedal & Wright, 2000). This SPD property can be achieved if the step length \( \alpha_i \) satisfies the sufficient descent condition
\[ E(s_i + \alpha_i \tilde{p}_i) \equiv E(\alpha_i) \leq E(0) + c_1 E'(0) \alpha_i , \]  

(3.15)

and the curvature condition

\[ E'(\alpha_i) \geq c_2 E'(0) , \]  

(3.16)

where \( E'(0) = \tilde{g}_T \tilde{p}_i \) and \( E'(\alpha_i) = \tilde{g}_{i+1}^T \tilde{p}_i \) are directional derivatives. This SPD property is important for an optimization problem, because it guarantees that a quasi-Newton method that uses this projected Hessian converges to (possibly local) minimum. In equations 3.15 and 3.16, \( c_1 \) and \( c_2 \) are constants \( \in (0, 1) \). Suggested values for \( c_1 \) and \( c_2 \) are 0.0001 and 0.9 (Nocedal & Wright, 2000), respectively.

Together, equations 3.15 and 3.16 are referred to as the Wolfe conditions (Nocedal & Wright, 2000). However, as also stated by Nocedal & Wright (2000), a step length \( \alpha_i \) that satisfies the Wolfe conditions may not be particularly close to a minimizer of the objective function \( E \). If the curvature condition in equation 3.16 is replaced by

\[ |E'(\alpha_i)| \leq c_2 |E'(0)| , \]  

(3.17)

then the Wolfe conditions become the strong Wolfe conditions (Moré & Thuente, 1994) used in this study. The strong Wolfe conditions can exclude step lengths that are far from stationary points of \( E \).

3.4.5 Cost of Projected BFGS

In each iteration of the P-BFGS method, one performs at least one forward calculation \( F(m) \) in the line search and compute at least one gradients \( g \). Provided proper \( c_1 \) and \( c_2 \), the first trial of step length \( \alpha_i \) can satisfy the strong Wolfe conditions, and therefore, in this case, the cost of each iteration of the P-BFGS method is that of three forward calculations \( F(m) \). In general, the line search that obeys the strong Wolfe conditions may need to examine more than one step length. In contrast, a quadratic line search, which is commonly used in conjugate-gradient methods, needs
one gradient $g$ and must test at least three trials of step length. Therefore, one iteration of conjugate-gradient method requires at least five forward calculations $F(m)$, 66% more than the P-BFGS method.

Compared to conjugate-gradient methods, the P-BFGS method requires additional memory for the projected Hessian matrix. Because we compute the projected Hessian matrix with respect to a sparse model space, the memory consumption is negligible compared to the memory used for forward modeling or gradient calculation. The P-BFGS method only needs the information from the last iteration, and it needs only $O(n^2) + 2O(n)$ memory. In the contrast, the L-BFGS method requires $(2M + 1) \cdot O(N)$ memory (Nocedal, 1980) to store the history model and gradient changes in the previous $M$ iterations; this can still be expensive in 3D. By considering the structural sparsity, we commonly construct a $n$-parameter sparse model with little redundancy, such that $n^2 < N$ as in the following example. Therefore, the P-BFGS method can save more memory than the L-BFGS method.

3.5 Example

We test the P-BFGS method in the time-domain FWI of the Marmousi II model. Figure 3.2(a) shows the true model $m$, and Figure 3.2(b) shows the initial model $m_0$, a highly smoothed version of the true model. We use 11 shots uniformly distributed on the surface, and a 12 Hz Ricker wavelet as the source for simulating wavefields. The effective frequency band that we can invert in the data is approximately 3–27 Hz; we implement the inversion in the time domain, which is equivalent to inverting the entire frequency band simultaneously. The source and receiver intervals are 0.76 km and 0.024 km, respectively. In this example, the dense model $m$ has $N = 391 \times 1151$ parameters. Therefore, computation or storage of the true Hessian matrix for this example is infeasible.
Figure 3.2: The Marmousi II velocity model (a) and an initial velocity model (b).
3.5.1 Sparse Model

Generally, model parameters are homogeneous within the same geological layers, where model structures keep coherent or do not change rapidly. This coherency between model parameters and model structures indicates that a model, in fact, is inherently sparse if one considers the structures in the model as \textit{a priori} constraints, which are useful in building a sparse model. Figure 3.3(a) shows an image of the Marmousi II data migrated using the initial model in Figure 3.2(b). On the top of the migrated image, white ellipses represent metric tensor fields (Hale, 2009a), which indicate the structural features (e.g., coherency, orientation, et al.,) of the image and imply the structures of the model as well. Given the tensors fields, we construct a sparse model $s$ using a structurally constrained sample selection scheme (Ma \textit{et al.}, 2012). Although a multi-scale approach may be available for FWI, migration should generally be performed with higher frequencies, such that migrated images can provide high-resolution structural features to construct a sparse model.

This selection method recursively picks samples based on a priority distance map (Appendix B), which measures how far each pixel in the migrated image is from the corresponding nearest reflector. We choose the first sample at the location with the largest distance. To keep samples sparse, we avoid picking another sample in a nearby area surrounding the first sample. We refer to this area as an exclusion region. The exclusion region can take different shapes, such as a rectangle or a circle, but they do not follow the structural features in the geological layers. Instead, we use the tensors again to design a structurally constrained exclusion region (see Chapter 2), which conforms to model structures. As a result, we less likely miss possible samples in adjacent layers. We repeatedly run this procedure until no more sample is available for picking. Figure 3.3(b) shows a total of 165 scattered sample locations. The scattered locations together with corresponding values at the locations comprise
Figure 3.3: Tensors that indicate structural characteristics of the model (a) and 165 sample locations of a sparse model space (b).
a sparse model space $s$ that will be used in the P-BFGS method. This set of scattered samples are representative, because in general

(i) they lie between (not on) reflectors;

(ii) they are distributed along structural features;

(iii) there are more samples in structurally complex areas and fewer samples in simple areas.

### 3.5.2 Projected Hessian and Its Inverse

Figure 3.4(a) shows the initial projected Hessian $\tilde{H}_0$, an identity matrix. Figure 3.4(b) shows the updated projected Hessian $\tilde{H}_1$ after one iteration of the P-BFGS method. The update $\tilde{H}_1 - \tilde{H}_0$ to the Hessian in this first iteration is shown separately in Figure 3.4(c). As we can see, the P-BFGS method adds significant off-diagonal components to the initial Hessian $H_0$. Because our line search satisfies the strong Wolfe conditions, the inverse of the projected Hessian matrix exists. Figure 3.4(d) shows the inverse Hessian $\tilde{H}_1^{-1}$. The inverse Hessian is not dominated by the diagonal elements, thereby indicating that, at this condition, it is inadequate to only use the diagonal term as preconditioning in FWI.

### 3.5.3 Eigenvalues and Eigenvectors

Because the projection Hessian has a small size so that we get a chance to apply eigendecomposition to it and analyze the eigenvalues and eigenvectors. Note that the last two terms of the right-hand side of equation 3.13 are two rank-one matrices, so each iteration of the P-BFGS method is a rank-two update (Nocedal & Wright, 2000). Figure 3.5(a), Figure 3.5(b), Figure 3.5(c) and Figure 3.5(d) show the eigenvalues of the projected Hessian $\tilde{H}_i$ in the 1st, 4th, 7th and 10th iterations, respectively. As
Figure 3.4: The initial Hessian $\tilde{H}_0$ (a), the Hessian matrix $\tilde{H}_1$ after the 1st iteration (b), the Hessian update $\tilde{H}_1 - \tilde{H}_0$ in the 1st iteration (c), and the inverse Hessian $\tilde{H}_1^{-1}$ (d).
suggested by the rank-two update, the projected BFGS method changes only two eigenvalues in each iteration: one, the smallest, and the other, the largest.

![Graphs showing eigenvalues](image)

Figure 3.5: Eigenvalues of projected Hessian in 1st (a), 4th (b), 7th (c), and 10th (d) iterations.

As suggested by Fichtner & Trampert (2011b), the Hessian matrix is useful for resolution analysis in FWI, as the Hessian directly measures the change of the objective function with respect to model perturbation. Figure 3.6(a) and Figure 3.6(b) show two eigenvectors of the projected Hessian $\tilde{H}_1$, corresponding to the largest and the smallest eigenvalues, respectively. Consider the objective function $E(m)$ as a multidimensional parabolic surface (Thacker, 1989), and we can find that the
eigenvectors have geometric meaning. The eigenvector associated with the largest eigenvalue points in the direction of greatest curvature on that surface. Updating the model $m$ along the eigenvector direction shown in Figure 3.6(a), we get the largest rate of change in the objective function $E(m)$. This eigenvector indicates the part of the model $m$ that can be best determined in FWI, and this part is in the shallow area of the model. Likewise, the eigenvector associated with the smallest eigenvalue indicates the direction of least curvature. Updating the model $m$ along the eigenvector direction shown in Figure 3.6(b) yields the smallest rate of change of $E(m)$. This eigenvector indicates the component of the model $m$ that is close to the null space of FWI and therefore can be least well resolved in FWI.

### 3.5.4 Application in Quasi-Newton FWI

The P-BFGS method updates the model in each iteration, and therefore the P-BFGS method can be directly used to obtain quasi-Newton solutions to FWI. The key difference between QNFWI with the P-BFGS method and other solutions, such as the conjugate gradient (CG) and the L-BFGS methods, is the update direction. Table 3.1 compares the update directions used in the CG method, the image-guided CG method (Chapter 2), the quasi-Newton method with L-BFGS, and our quasi-Newton method with P-BFGS. In the quasi-Newton method with L-BFGS, we never compute a Hessian matrix $H_i$ or its inverse $H_i^{-1}$; we instead use stored information from previous 5 iterations to compute $-H_i^{-1}g_i$ in a recursive way (Nocedal, 1980). In the quasi-Newton method with P-BFGS, we first compute a projected Hessian matrix $\tilde{H}_i$ and its inverse $\tilde{H}_i^{-1}$, and we then directly obtain the update direction $\tilde{p}_i$ with a matrix multiplication $-\tilde{H}_i^{-1}\tilde{g}_i$.

As previously suggested, the IGI operator $R$ and its adjoint operator $R^T$ can be characterized by the action of scattering and gathering, respectively. Therefore, the joint operator $RR^T$ in image-guided CG method implies a gather-scatter process.
Figure 3.6: Eigenvectors of $\tilde{\mathbf{H}}_1$ corresponding to the largest (a) and the smallest (b) eigenvalues. Geometrically, an eigenvector is a direction on the multidimensional surface of an objective function $E(\mathbf{m})$, and the associated eigenvalue determines the curvature on the surface. If the model is updated in the eigenvector direction in (a), the rate of change of $E(\mathbf{m})$ is largest. If the model is updated in the eigenvector direction in (b), the rate of change of $E(\mathbf{m})$ is smallest.
The direct consequence of using this gather-scatter process is that one can generate geologically plausible low wavenumbers, which, on the other hand, are not present in the simple conjugate gradient due to the lack of low frequencies in recorded data. The update directions $p_i$ of the CG method in Figure 3.7(a) and the image-guided conjugate gradient (Figure 3.7(b)) illustrate this difference.

Figure 3.8(a) and Figure 3.8(b) show inversion results estimated by the CG method and the image-guided CG method after 10 iterations, respectively. We can see that both the conventional CG method and the image-guided CG method recover the shallow part of the model better than the deep part. This phenomena is due to the fact that both the conjugate gradient (Figure 3.7(a)) and the image-guided conjugate gradient (Figure 3.7(b)) have unbalanced amplitudes, high in the shallow part and low in the deeper part. This is common because, in recorded data, later reflection arrivals have much weaker energy than early arrivals, which mainly represent shallow
Figure 3.7: Update directions $p_1$ in FWI (a) and image-guided FWI (b).
Figure 3.8: Velocity models estimated with FWI (a) and image-guided FWI (b). Both in the 10th iteration.
parts of the model.

Figure 3.9(a) shows the update direction $p_1$ in QNFWI with L-BFGS, which uses a history of previous 10 iterations to approximate the Hessian and then generates the update direction. We can observe the change of amplitudes between the L-BFGS update direction (Figure 3.9(a)) and the CG update direction (Figure 3.7(a)). This change is not very significant in Figure 3.9(a) because L-BFGS only has information from one previous iteration to compute $p_1$. As the iteration increases, Figure 3.10(a) shows the inversion result estimated in the 10th iteration of QNFWI with L-BFGS. Compared with the conventional FWI result (Figure 3.8(a)), the QNFWI with L-BFGS generates a model (Figure 3.10(a)) that presents more balanced amplitudes in deep areas of the model. However, the L-BFGS method also boosts geologically nonsensible artifacts, which are geologically misleading in the reservoir area. Moreover, many deep reflectors are misplaced at wrong positions by conventional FWI in Figure 3.8(a) and QNFWI with L-BFGS can not correct them in Figure 3.10(a).

In the case of QNFWI with P-BFGS, the update direction $p_1$ shown in Figure 3.9(b) not only contains significant low wavenumbers, as does the image-guided conjugate gradient in Figure 3.7(b), but shows more balanced amplitudes as well. This improvement is due to the use of the inverse projected Hessian matrix $\tilde{H}_i^{-1}$, which works as a spatially varying filter applied to the gradient. Figure 3.10(b) shows the model estimated in the 10th iteration of QNFWI with P-BFGS. As we can see, QNFWI with P-BFGS updates the deeper part of the model better than either the conventional CG method or the image-guided CG method. Because we construct the projected Hessian while taking subsurface structures as constraints, QNFWI with P-BFGS eliminates those misleading artifacts shown in Figure 3.10(a), which is estimated by QNFWI with L-BFGS. Moreover, because of the additional low wavenumbers, QNFWI with P-BFGS, like image-guided FWI, places reflectors
Figure 3.9: Update directions $p_1$ in QNFWI with L-BFGS (a) and QNFWI with P-BFGS (b).
Figure 3.10: Velocity models estimated with QNFWI with L-BFGS (a) and QNFWI with P-BFGS (b). Both in the 10th iteration.
at more accurate positions than conventional FWI and QNFWI with L-BFGS.

Figure 3.11 shows the convergence rates of this example with a data misfit function and a model misfit function (an L2 norm of the difference between the true model and the estimated model). Compared with the conventional FWI with the CG method, in Figure 3.11(a), image-guided FWI, QNFWI with L-BFGS, and our QNFWI with P-BFGS all decrease the data misfit faster. Image-guided FWI and QNFWI with P-BFGS eventually approach a similar level in the data misfit function because they both well recover the shallow part of the model, where the transmitted and reflected energies dominate the data misfit. In Figure 3.11(b), QNFWI with L-BFGS reduces the model misfit curve faster than conventional FWI because it updates the model more in the deep part. As previously mentioned, both image-guided FWI and QNFWI with P-BFGS generate significant low wavenumbers to update the model, and therefore, they both decrease the model misfit faster than QNFWI with L-BFGS. Because our QNFWI with P-BFGS also recovers the deep part of the model better, it reduces the model misfit in Figure 3.11(b) most rapidly among all four methods. Overall, QNFWI with P-BFGS shows a faster convergence.

Two different line-search methods are used in this example. In QNFWI with L-BFGS and QNFWI with P-BFGS, we employ the same line-search method that satisfies the strong Wolfe conditions; in CG-based conventional FWI and image-guided FWI, we use a quadratic line search. In the 10 iterations shown in Figure 3.11, both the L-BFGS and P-BFGS methods only need to test one step length to meet the strong Wolfe conditions. That is, in each iteration, both QNFWI with L-BFGS and QNFWI with P-BFGS are less expensive than either conventional FWI or image-guided FWI because as previously suggested, the quadratic line search is 66% more expensive than the line search obeying the strong Wolfe conditions. If we set the same computing time for all these methods, FWI and image-guided FWI will perform less
Figure 3.11: Convergence of FWI, image-guided FWI, QNFWI with L-BFGS, and QNFWI with P-BFGS: the data misfit function (a) and the model misfit function (b).
iterations than both quasi-Newton methods. Of course, a fairer comparison can be achieved if we use the same line-search method that satisfies the strong Wolfe conditions for all the inversion schemes. Nevertheless, because both conventional FWI and image-guided FWI are based on conjugate gradients, a quadratic line search is practically more suitable.

In this example, an identity matrix is used as the initial Hessian matrix in both the L-BFGS and the P-BFGS methods. We can replace the identity matrix with an illumination map (Xie et al., 2006), and in this way, we can expect faster convergence in both QNFWI with L-BFGS and QNFWI with P-BFGS. The same illumination map can also be used in FWI and image-guided FWI to compensate the update in deep parts of the model.

3.6 Conclusions

A P-BFGS method iteratively approximates the Hessian matrix in FWI with respect to fewer parameters in a sparse model, thereby reducing both computation time and required memory. As a consequence, the P-BFGS method enables realistic applications in QNFWI. As demonstrated by the Marmousi II example, QNFWI with P-BFGS converges faster in fewer iterations than gradient-based methods and the L-BFGS method. Because structural features are used to construct a sparse model and to constrain the sparse optimization, the model estimated by QNFWI with P-BFGS makes better geological sense than the result generated by the conventional FWI and QNFWI with P-BFGS, which do not consider structural constraints.

The P-BFGS method and the consequent QNFWI can be extended to 3D cases, where the advantage of projected Hessian may be more obvious. The structural sparsity in 3D models is generally more significant because the earth model turns to be more structurally coherent in the inline-crossline direction. The sparse model in this chapter is constructed basing on the structural sparsity in the model. In possible
cases where models do not depend on structures (e.g., velocity anomalies caused by overpressure), other priori information such as well log measurements and geologic interpretations, can be used as additional constraints in designing the operators $\mathbf{R}$ and $\mathbf{R}^T$. 
4.1 Summary

In reflection seismology, full waveform inversion (FWI) can generate high-wavenumber subsurface velocity models but often suffers from an objective function with local minima caused mainly by the absence of low frequencies in seismograms. These local minima cause cycle skipping when the low-wavenumber component in the initial velocity model for FWI is far from the true model. To avoid cycle skipping, we propose a new wave-equation reflection traveltime inversion (WERTI) to update the low-wavenumber components of the velocity model, while using FWI to only update high-wavenumber details in the model. We implement the low- and high-wavenumber inversions in an alternating way. In WERTI, we use dynamic image warping (DIW) to estimate the time shifts between recorded data and synthetic data. When compared with correlation-based techniques often used in traveltime inversion, DIW can avoid cycle skipping and thereby reduce errors in the estimated time shifts. The alternating combination of WERTI and FWI mitigates the velocity-depth ambiguity and can recover subsurface velocities using only high-frequency reflection data.
4.2 Introduction

Traveltime inversion (Bishop et al., 1985; Luo & Schuster, 1991; Stork, 1988) and full waveform inversion (FWI) (Pratt, 1999; Tarantola, 1984) are two different methods that geophysicists use to estimate subsurface velocity models. Both methods have advantages and disadvantages in estimating a subsurface model $m = m(x)$, a function of spatial coordinates. In this study, we consider that the model $m$ is composed of a smooth low-wavenumber background $m^s$ and a rough high-wavenumber component $m^r$ (Clément et al., 2001):

$$m = m^s + m^r,$$  \hspace{1cm} (4.1)

where $m^s$ corresponds to the kinematics of wave propagation and $m^r$ corresponds to reflections in seismograms.

Traveltime inversion minimizes only traveltime differences between observed (recorded) data $p_o(x_r, t; x_s)$ and computed (synthetic) data $p_c(x_r, t; x_s)$, where $x_s$ and $x_r$ denote source and receiver locations, respectively. The synthetic data satisfy a wave equation. For the purpose of this paper, we choose a constant-density acoustic wave equation:

$$\left( m(x) \frac{\partial^2}{\partial t^2} - \nabla^2 \right) p(x, t; x_s) = f(t; x_s),$$  \hspace{1cm} (4.2)

where $m(x)$ denotes the slowness square $v(x)^{-2}$, and $f(t; x_s)$ denotes a source function. Because the traveltime shift, compared with the data difference, is more sensitive and more linearly related to the long-wavelength model perturbation (Cara & Lévéque, 1987; Fichtner et al., 2008), traveltime inversion better resolves the low-wavenumber background model $m^s$, which can be subsequently used as an initial model in FWI for the high-wavenumber details $m^r$.

Traveltime inversion has succeeded in applications to cross-well data (Luo & Schuster, 1991; van Leeuwen & Mulder, 2010) and refraction data (Zelt & Barton, 1998),
where commonly only first arrivals are used and traveltime shifts are assumed to be
time-invariant. However, reflection traveltime inversion (RTI) is more difficult. One
difficulty is that traveltime shifts vary with time. Manually picking arrivals in seismo-
grams is the most reliable way to estimate these shifts but is time-consuming. More
automatic approaches compute crosscorrelations of seismograms to estimate time
shifts (Luo & Schuster, 1991; van Leeuwen & Mulder, 2010). Unfortunately, when
time shifts change rapidly, correlation-based methods suffer from the cycle-skipping
problem. A second difficulty in RTI is the ambiguity between the low-wavenumber
background velocities and the depths of high-wavenumber reflectors (Stork & Clayton,
1986).

FWI, on the other hand, estimates the subsurface model \( \mathbf{m} \) by minimizing a data-
difference objective function

\[
E_F (\mathbf{m}) = \frac{1}{2} \| p_c (x_r, t; x_s) - p_o (x_r, t; x_s) \|^2 .
\]

This data-difference objective function comprehensively measures differences between
the recorded data and the synthetic data including differences in traveltimes, ampli-
tudes, converted waves, multiples, etc. Gradient-based methods are often used to
solve the FWI optimization problem. The gradient of the objective function in FWI
can be computed by reverse time migration (Tarantola, 1984) of the data difference

\[
\Delta p (x_r, t; x_s) = p_c (x_r, t; x_s) - p_o (x_r, t; x_s)
\]

If the data lack low frequencies, the gradient in FWI lacks low wavenumbers. Therefore,
FWI can update and may recover the high-wavenumber component of the model
\( \mathbf{m}^r \), but it cannot estimate the low-wavenumber background model \( \mathbf{m}^s \) (Snieder et al.,
1989).

The ability of FWI to recover \( \mathbf{m}^r \) depends on an accurate background model \( \mathbf{m}^s \).
If the initial background model is far way from the true model and the recorded data
do not have sufficient low frequencies, FWI cannot recover the low-wavenumber background and therefore suffers from problems such as local minima and cycle skipping (Virieux & Operto, 2009). In the presence of cycle skipping, the high-wavenumber details $m^r$ estimated with FWI are mispositioned. Various methods have been proposed to solve the cycle-skipping problem. Multiscale approaches (Bunks, 1995; Sirgue & Pratt, 2004) recursively add higher-wavenumber details to models first computed from lower-frequency data, but they are useful only if sufficient low frequencies are available. Snieder et al. (1989) and Clément et al. (2001) reparameterize $m^r$ as a function of vertical traveltime instead of depth to mitigate cycle skipping. Sparse-model inversions (Ma & Hale, 2012; Ma et al., 2012) mitigate the lack of low frequencies by generating so-called image-guided low-wavenumber gradients.

Adapting an alternating inversion strategy (Clément et al., 2001; Snieder et al., 1989; Xu et al., 2012), we propose a new wave-equation reflection traveltime inversion (WERTI) to update the low-wavenumber $m^s$, while using conventional FWI to generate the high-wavenumber $m^r$. During application of WERTI, the high-wavenumber model $m^r$ is fixed; likewise, when performing FWI, the background $m^s$ is fixed. In WERTI, we use dynamic image warping (DIW) (Hale, 2012) to estimate the time-varying traveltime misfit. DIW avoids cycle skipping inherent in crosscorrelation methods when traveltime shifts change rapidly.

In the following sections, we first introduce DIW to estimate traveltime shift. Then, we analyze and compute the gradient in WERTI using the methods introduced by Luo & Schuster (1991) and Xu et al. (2012). We then test our method for a high-velocity-anomaly model and a part of the Sigsbee2A model (Paffenholz et al., 2002). Results demonstrate that by combining WERTI and FWI, we can use only high-frequency reflection data to recover both low- and high-wavenumber components of velocity models.
4.3 Wave-Equation Reflection Traveltime Inversion

Common methods for estimating time shifts are based on crosscorrelation (Luo & Schuster, 1991; van Leeuwen & Mulder, 2010). Crosscorrelation-based methods, however, may suffer from cycle skipping when shifts change rapidly. In this paper, we choose dynamic image warping (DIW; Hale, 2012) to overcome the cycle-skipping problem in estimating time shifts. After posing an inverse problem for WERTI, we investigate the gradient for WERTI to update the low-wavenumber component of velocity models and propose an alternating inversion strategy.

4.3.1 Traveltime Shifts

DIW obtains the traveltime misfit $u \equiv u(x_r, t; x_s)$ between the recorded data $p_o(x_r, t; x_s)$ and the synthetic data $p_c(x_r, t; x_s)$ by finding a shift field $u(x_r, t; x_s)$ that is the solution to the following constrained optimization problem:

$$ u(x_r, t; x_s) = \arg \min_{l(x_r, t; x_s)} D(l(x_r, t; x_s)) , $$

where

$$ D(l(x_r, t; x_s)) = \frac{1}{2} \| p_c(x_r, t; x_s) - p_o(x_r, t + l(x_r, t; x_s); x_s) \|^2 , $$

subject to the constraints

$$ \left| \frac{\partial u}{\partial t} \right| \leq \delta_t, \quad \left| \frac{\partial u}{\partial x_r} \right| \leq \delta_r, \quad \text{and} \quad \left| \frac{\partial u}{\partial x_s} \right| \leq \delta_s . $$

The constraints in equation 4.7 ensure that the time shifts $u(x_r, t; x_s)$ neither decrease nor increase too rapidly in time or space (Hale, 2012). In this paper, we choose $\delta_t = 0.5, \delta_r = 0.1$ and $\delta_s = \infty$. The infinite bound $\delta_s$ means that we do not constrain the shifts in the direction of $x_s$; in other words, we estimate these time shifts for each shot, independently. If necessary, we could instead specify a finite $\delta_s$ to constrain changes in time shifts with respect to shot coordinates.
The DIW optimization problem implied by equation 4.6 is similar to the FWI problem in equation 4.3. FWI must find a model $\mathbf{m}$ to minimize the objective function (equation 4.3); nevertheless, DIW aims to find time shifts $\mathbf{u}$ to minimize the objective function (equation 4.6). The inherent relationship between the model $\mathbf{m}$ and time shifts $\mathbf{u}$ will be used below to compute the gradient for WERTI.

### 4.3.2 Inverse Problem

After we obtain the traveltime misfit $u(\mathbf{x}_r, t; \mathbf{x}_s)$ between the recorded data $p_o(\mathbf{x}_r, t; \mathbf{x}_s)$ and the synthetic data $p_c(\mathbf{x}_r, t; \mathbf{x}_s)$, we can use this misfit in WERTI. To do this, we must formulate an objective function that is minimized (or maximized) when the misfit $u(\mathbf{x}_r, t; \mathbf{x}_s)$ is zero, indicating the reflections in the recorded data $p_o(\mathbf{x}_r, t; \mathbf{x}_s)$ and in the synthetic data $p_c(\mathbf{x}_r, t; \mathbf{x}_s)$ are well-aligned. The most straightforward way of using the misfit $u(\mathbf{x}_r, t; \mathbf{x}_s)$ is analogous to conventional traveltime tomography, where we aim to solve a least-squares inverse problem:

$$E_T(\mathbf{m}) = \frac{1}{2} \| u(\mathbf{x}_r, t; \mathbf{x}_s) \|^2.$$  \hspace{1cm} (4.8)

### 4.3.3 Gradient Calculation

Various methods, including conjugate-gradient methods and quasi-Newton methods, can be used to solve the inverse problem implied by equation 4.8. All of the methods require computing the gradient of the objective function with respect to the model. In WERTI, the gradient is

$$\mathbf{g}_T(\mathbf{m}) \equiv \frac{\partial E_T}{\partial \mathbf{m}} = \left( \frac{\partial u^T}{\partial \mathbf{m}} \right) \mathbf{u} = \left( \frac{\partial u_1}{\partial \mathbf{m}} \frac{\partial u_2}{\partial \mathbf{m}} \cdots \frac{\partial u_n}{\partial \mathbf{m}} \right) \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix},$$ \hspace{1cm} (4.9)

where $u_i$ is the $i$th element of $u(\mathbf{x}_r, t; \mathbf{x}_s)$.

We can compute the $i$th column of $\frac{\partial \mathbf{u}^T}{\partial \mathbf{m}}$ by using the concept of a connective function (Luo & Schuster, 1991) and the rule for the derivative of an implicit function:
\[
\frac{\partial u_i}{\partial m} = -\frac{\partial F_i}{\partial m} / \frac{\partial F_i}{\partial u_i}, \tag{4.10}
\]

where the implicit connective function \( F_i(u_i, m) = 0 \) and \( \frac{\partial F_i}{\partial u_i} \neq 0 \).

In dynamic warping we choose the shifts \( u_i \) to minimize the sum

\[
D(u_1, u_2, ..., u_n) = \frac{1}{2} \sum_i (p_c[i] - p_o[i + u_i])^2 \tag{4.11}
\]

subject to the constraints that the shifts \( u_i \) vary smoothly. Therefore, a suitable choice for the implicit function is \( F_i \equiv \frac{\partial D}{\partial u_i} = 0 \). This partial derivative \( \frac{\partial D}{\partial u_i} \) must be zero because if it were not zero, then we could decrease \( D \) by changing \( u_i \) slightly, which contradicts the fact that \( D \) is minimized in dynamic warping. Subsequently, the denominator and numerator of equation 4.10 are, respectively,

\[
\frac{\partial F_i}{\partial u_i} = \frac{\partial^2 D}{\partial u_i^2} = \frac{\partial p_o[i + u_i]^2 - \partial p_o[i + u_i]}{\partial p_c[i] - \partial p_o[i + u_i]} \tag{4.12}
\]

and

\[
\frac{\partial F_i}{\partial m} = \frac{\partial p_o[i + u_i]}{\partial m} . \tag{4.13}
\]

Using equations 4.10, 4.12 and 4.13, we can rewrite the gradient in equation 4.9 as an integration:

\[
g_T(m) = -\sum_s \sum_r \int dt \left( \frac{\partial p_c(x_r, t; x_s)}{\partial m} \right) A(x_r, t; x_s) , \tag{4.14}
\]

where \( s \) and \( r \) are source and receiver indices, and the Frechét derivative \( \frac{\partial p_c(x_r, t; x_s)}{\partial m} \) measures the wavefield perturbation cause by the model perturbation. In equation 4.14, \( A(x_r, t; x_s) \) can be referred to as the adjoint source in the adjoint-state method (Plessix, 2006):

\[
A(x_r, t; x_s) = \frac{\dot{p}_o(x_r, t + u(x_r, t; x_s); x_s)}{c(x_r, t; x_s)} u(x_r, t; x_s) , \tag{4.15}
\]

where
\[
c (x_r; t; x_s) = \dot{p}_0 (x_r, t + u(x_r, t; x_s); x_s)^2
- \dot{p}_0 (x_r, t + u(x_r, t; x_s); x_s) (pc (x_r, t; x_s) - p_o (x_r, t + u(x_r, t; x_s); x_s)) .
\]  
(4.16)

### 4.3.4 Which Gradient?

As described in the introduction, the model \( m \) contains both the smooth low-wavenumber background \( m^s \) and the rough high-wavenumber component \( m^r \). For a fixed background \( m^s \), to invert for \( m^r \), we need a gradient with respect to \( m^r \). For simplicity, hereafter we omit the sum over source and receiver indices \( s \) and \( r \), and can rewrite equation 4.14 as

\[
g_T (m^r) = - \int dt \left( \frac{\partial pc (x_r, t; x_s)}{\partial m^r} \right) A (x_r, t; x_s) .
\]  
(4.17)

Appendix C.1 shows that the Frechét derivative is

\[
\frac{\partial pc (x_r, t; x_s)}{\partial m^r} = - \dot{G} (x_r, t; m^s, x) * \dot{p} (x, t; x_s) ,
\]  
(4.18)

where \( G (x, t; m^s, x') \) is the Green's function for equation 4.2 (for the smooth model \( m^s \)), and * denotes time convolution. Substituting equation 4.18 into equation 4.17, we obtain

\[
g_T (m^r) = \int dt \left( \dot{G} (x_r, t; m^s, x) * \dot{p} (x, t; x_s) \right) A (x_r, t; x_s) .
\]  
(4.19)

Using the identity \( \int dt [f(t) * g(t)]h(t) = \int dt g(t) [f(-t) * h(t)] \) (Tarantola, 1984) and the reciprocity \( G (x, t; m^s, x_r) = G (x_r, t; m^s, x) \), we can rewrite equation 4.19 as

\[
g_T (m^r) = \int dt \dot{p} (x, t; x_s) \left( \dot{G} (x, -t; m^s, x_r) * A (x_r, t; x_s) \right) ,
\]  
(4.20)

which implies reverse time migration (RTM), a zero-lag crosscorrelation of the source wavefield \( \dot{p} (x, t; x_s) \) and the back-propagated adjoint wavefield \( \dot{G} (x, -t; m^s, x_r) * A (x_r, t; x_s) \).
The gradient $\mathbf{g}^T (\mathbf{m}^r)$ supplies mainly high wavenumbers near reflectors; it cannot be used in WERTI to recover the low-wavenumber background model $\mathbf{m}^s$. Therefore, we should instead compute the gradient with respect to $\mathbf{m}^s$, $\mathbf{g}^T (\mathbf{m}^s) \equiv \frac{\partial E^T}{\partial \mathbf{m}^s}$. Given a fixed $\mathbf{m}^r$, equation 4.14 becomes

$$ \mathbf{g}^T (\mathbf{m}^s) = - \int dt \left( \frac{\partial p_c (x_r, t; x_s)}{\partial \mathbf{m}^s} \right) \mathcal{A} (x_r, t; x_s) . \quad (4.21) $$

We follow Xu et al. (2012) to compute the required Frechét derivative $\frac{\partial p_c (x_r, t; x_s)}{\partial \mathbf{m}^s}$ (Appendix C.2):

$$ \frac{\partial p_c (x_r, t; x_s)}{\partial \mathbf{m}^s} = - \dot{\mathcal{G}} (x_r, t; \mathbf{m}^s, x) * \dot{p} (x, t; x_s) $$

$$ - \delta \mathcal{G} (x_r, t; \delta \mathbf{m}^r, x) * \dot{p} (x, t; x_s) , \quad (4.22) $$

where $\delta \mathcal{G} (x, t; \delta \mathbf{m}^r, x')$ is the Green’s function perturbation caused by a rough-model perturbation $\delta \mathbf{m}^r$, and accordingly $\delta p (x, t; x_s) = \delta \mathcal{G} (x, t; \delta \mathbf{m}^r, x) * f (t; x_s)$ is the wavefield perturbation due to $\delta \mathbf{m}^r$ generated by the source $f (t; x_s)$.

Substituting equation 4.22 into equation 4.21, we obtain

$$ \mathbf{g}^T (\mathbf{m}^s) = \int dt \left( \dot{\mathcal{G}} (x_r, t; \mathbf{m}^s, x) * \dot{p} (x, t; x_s) \right) \mathcal{A} (x_r, t; x_s) $$

$$ + \int dt \left( \delta \dot{\mathcal{G}} (x_r, t; \delta \mathbf{m}^r, x) * \dot{p} (x, t; x_s) \right) \mathcal{A} (x_r, t; x_s) . \quad (4.23) $$

Similarly, we rewrite equation 4.23 as

$$ \mathbf{g}^T (\mathbf{m}^s) = \left\{ \int dt \delta p (x, t; x_s) \left( \dot{\mathcal{G}} (x; -t; \mathbf{m}^s, x_r) * \mathcal{A} (x_r, t; x_s) \right) \text{SR} \right\} $$

$$ + \left\{ \int dt \dot{p} (x, t; x_s) \left( \delta \dot{\mathcal{G}} (x; -t; \delta \mathbf{m}^r, x_r) * \mathcal{A} (x_r, t; x_s) \right) \text{RS} \right\} . \quad (4.24) $$

The first term $\text{SR}$ on the right side of equation 4.24 represents the zero-lag crosscorrelation of a forward source-side wavefield perturbation $\delta p (x, t; x_s)$ and a backward
wavefield excited by $\mathcal{A}(x_r, t; x_s)$; likewise, the second term $RS$ is the crosscorrelation of a forward wavefield $\hat{\mathcal{p}}(x, t; x_s)$ and a backward receiver-side wavefield perturbation.

In equation 4.24 the fixed high-wavenumber component $m^r$ scatters both the source wavefield and the back-propagating receiver wavefield. In other words, every point in $m^r$ can be considered a secondary source that contributes to the scattered wavefield. With these secondary sources, the calculation of $g_T(m^s)$ is analogous to that in a transmission-type experiment, as in cross-well examples (Luo & Schuster, 1991; van Leeuwen & Mulder, 2010), where we can expect low-wavenumber gradients for inversion. With the gradient $g_T(m^s)$, WERTI iteratively updates the low-wavenumber background model $m^s$ with standard conjugate-gradient methods.

<table>
<thead>
<tr>
<th>WERTI</th>
<th>FWI</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E(m)$</td>
<td>$\frac{1}{2}|u(x_r, t; x_s)|^2$</td>
</tr>
<tr>
<td>adjoint source</td>
<td>$\mathcal{A}(x_r, t; x_s)$</td>
</tr>
<tr>
<td>$g(m^r)$</td>
<td>$-\int dt \left( \frac{\partial p_c(x_r, t; x_s)}{\partial m^r} \right) \mathcal{A}(x_r, t; x_s)$</td>
</tr>
<tr>
<td>$g(m^s)$</td>
<td>$-\int dt \left( \frac{\partial p_c(x_r, t; x_s)}{\partial m^s} \right) \mathcal{A}(x_r, t; x_s)$</td>
</tr>
</tbody>
</table>

Table 4.1: Comparison between WERTI and FWI.

### 4.3.5 Compared with FWI

FWI employs the different objective function $E_F$, which includes (directly) amplitude differences and (indirectly) traveltime differences. As in the derivation above of gradients in WERTI, we can also get two gradients for FWI:

$$g_F(m^r) = \int dt \left( \frac{\partial p_c(x_r, t; x_s)}{\partial m^r} \right) \Delta p(x_r, t; x_s)$$

(4.25)
and
\[
g_F(m^s) = \int dt \left( \frac{\partial p_c(x_r, t; x_s)}{\partial m^s} \right) \Delta p(x_r, t; x_s),
\]
where the data difference \( \Delta p(x_r, t; x_s) \) is simply the adjoint source defined in equation 4.4.

Table 4.1 compares WERTI with FWI by summarizing the objective functions, the adjoint sources, and gradients. This table highlights the fact that both WERTI and FWI share the same method for computing gradients; the only differences are in the definitions of the objective function and the consequent adjoint sources.

Although FWI can also use the gradient \( g_F(m^s) \) to update the low-wavenumber background model \( m^s \) (Xu et al., 2012), it still suffers from local minima, because FWI in this case still employs the simple data-difference objective function \( E_F(m) \) defined in equation 4.3.

### 4.4 Inversion Procedure

Our inversion methodology is the following combination of WERTI and FWI:
given initial smooth model \( m_0^s \)

1) with \( m_0^s \), apply one iteration of FWI to obtain \( \delta m_0^r \)

2) for \( i = 1, 2, \ldots \) until traveltime objective function \( E_T \) converges, do
   
   fix \( \delta m_{i-1}^r \) and apply WERTI to update \( m_i^s \)
   
   fix \( m_i^s \) and apply FWI to obtain \( \delta m_i^r \)
   
end for

initialize model \( m^s = m_i^s, m_0^r = 0, m_0 = m^s \)

3) for \( j = 1, 2, \ldots \) until data objective function \( E_F \) converges, do
   
   apply FWI to obtain \( \delta m_j^r \)
   
   update model \( m_j^r = m_{j-1}^r + \delta m_j^r, m_j = m^s + m_j^r \)
   
end for

In practice, inversion often begins with a smooth model \( m_0^s \) that would create no reflections in synthetic seismograms. However, WERTI requires reflections in the synthetic data, so that DIW can estimate time shifts between these reflections and those in the recorded data. Therefore, before WERTI, we employ one iteration of FWI in step 1 to obtain the rough-model perturbation \( \delta m_0^r \), which we use to synthesize reflections in seismograms.

Then, WERTI in step 2 extracts traveltime shifts from the synthetic seismic reflections and updates the low-wavenumber component \( m^s \). Also in step 2, we apply again one iteration of FWI after every WERTI to obtain a new rough-model perturbation \( \delta m_i^r \), which we use to replace \( \delta m_{i-1}^r \) and to generate new seismic reflections; this is necessary because reflectors in \( \delta m^r \) will move after WERTI updates the low-wavenumber component \( m^s \). Through the alternation in step 2, we can eliminate the ambiguity between the low-wavenumber velocities and the depths of reflectors.
After we resolve the low-wavenumber component \( m^s \) in step 2, we begin iterations of conventional FWI in step 3 to iteratively update the high-wavenumber component of the model \( m^r \), which refines the model \( m \) with details.

When performing WERTI in step 2, we apply a spatial low-pass filter to the gradient \( g_T(m^s) \), so that we can fix the rough-model perturbation \( \delta m^r \) and update only the low-wavenumber component \( m^s \). Likewise, also in step 2, when performing FWI after every WERTI, we apply a spatial high-pass filter to the gradient \( g_F(m^r) \) to update only the rough-model perturbation \( \delta m^r \).

### 4.5 Examples

To demonstrate the application of WERTI, we test the algorithm using a simple Gaussian high-velocity anomaly model and a part of the Sigsbee2A model. In both tests, we use only reflection data; we mute any direct and refracted arrivals.

#### 4.5.1 A Simple Velocity Anomaly

We refer to the model \( m \) in Figure 4.1(a) that contains a Gaussian high-velocity anomaly as the true model. Figure 4.1(b) displays the initial constant model \( m_0^s \) from which we start the inversion. In this example, we employ 24 evenly distributed shots on the surface, and a Ricker wavelet with a peak frequency of 15 Hz is used as the source for simulating wavefields. The source and receiver intervals are 0.16 km and 0.016 km, respectively. In this example, the maximum time is 2 s and the maximum offset is 4 km.

Figure 4.2(a) and Figure 4.2(b) show the FWI gradient \( g_T(m^r) \) and the updated model \( (m_0^s + \delta m_0^r) \), respectively, for only one iteration of conventional FWI. As expected, conventional FWI mispositions the reflector and fails to recover the Gaussian anomaly above the reflector. However, beginning with this misplaced reflector, WERTI can recover the anomaly.
Figure 4.1: True velocity model (a) that contains a high-velocity Gaussian anomaly and an initial model (b) for inversion.
Figure 4.2: Gradient $\mathbf{g}_F(\mathbf{m}^r)$ (a) and updated velocity (b) in the 1st iteration of conventional FWI, which fails to estimated the anomaly.
WERTI first synthesizes reflection data using the model shown in Figure 4.2(b), and one of the common shot gathers is displayed in Figure 4.3(b). Figure 4.3(c) illustrates the difference between the true data (Figure 4.3(a)) and the synthetic data (Figure 4.3(b)), and highlights the cycle skipping, especially for offsets > 2 km.

More compelling evidence of cycle skipping is in the time shifts $u(x_r, t; x_s)$ estimated from the recorded and synthetic data. Figure 4.4(a) displays the time shifts estimated with dynamic warping, and these shifts are normalized by the dominant period of the data. In Figure 4.4(a) shifts greater than half a cycle indicate the presence of cycle skipping.

After obtaining the time-varying time shifts $u(x_r, t; x_s)$, we calculate the objective function $E_T$ and compute the gradient $g_T(m^s)$. The adjoint-state method is an efficient way to compute the gradient, and it uses the adjoint source previously defined as $A(x_r, t; x_s)$. Figure 4.4(b) shows an example of the adjoint source for one shot. Figure 4.5(a) depicts the gradient $g_T(m^s)$ in the first iteration of WERTI, clearly showing a low-wavenumber feature near the high-velocity anomaly. After 10 iterations of WERTI, the estimated model in Figure 4.5(b) recovers the anomaly reasonably well and, moreover, locates the reflector in the correct position.

As suggested by Xu et al. (2012), we could instead modify conventional FWI by using the new gradient $g_F(m^s)$ to estimate the high-velocity anomaly in this example. Figure 4.6(a) displays the gradient $g_F(m^s)$ in the first iteration of such a modified FWI, and Figure 4.6(b) shows the updated model after 10 iterations. In general, this modified FWI approach changes the model in the right direction. However, due to cycle skipping at far offsets (Figure 4.3(c)) and the resulting strong side lobes in the gradient, convergence for this method is significantly slower than for WERTI.
Figure 4.3: Common shots gathers simulated (a) in the true model (Figure 4.1a) and (b) in the FWI model (Figure 4.2b). The data difference between (a) and (b) is displayed in (c), where cycle skipping appears in distance greater than 2 km.
Figure 4.4: (a) Normalized traveltime misfit estimated with dynamic warping between the recorded data (Figure 4.3a) and the synthetic data (Figure 4.3b). The maximum shift is greater than 0.5, indicating an existence of cycle skipping. (b) One example of the adjoint source $A(x_r, t; x_s)$ in WERTI.

### 4.5.2 Sigsbee Model

Figure 4.7(a) displays the true velocity model that is part of the Sigsbee2A model created by SMAART JV (Paffenholz et al., 2002); Figure 4.7(b) shows an initial velocity, which increases as a function of depth. Except for the water layer, which is the same as in the true model, the initial model in Figure 4.7(b) is a scaled (by 0.95) and smoothed version of the horizontal average of the true model. In this example, except for the maximum offset (3 km), other survey parameters are the same as in the previous example. Figure 4.8 displays two common shot gathers simulated in the true model with a 15 Hz Ricker wavelet as the source, and only reflection data will be used for inversion.

Figure 4.9(a) is the RTM image migrated with the true model; accordingly Figure 4.9(b) is the migration image for the initial model. Because the initial model significantly deviates from the true model, all the reflectors in Figure 4.9(b), except
Figure 4.5: Gradient $g_T(m^s)$ (a) in the first iteration and updated velocity (b) in the 10th iteration of WERTI.
Figure 4.6: Gradient $\mathbf{g}_F (\mathbf{m}^*)$ (a) in the first iteration and updated velocity (b) in the 10th iteration of Xu et al. (2012)’s modified FWI. Due to strong side lobes in the gradient, this method converges very slowly.
Figure 4.7: True velocity model (a) that is part of the Sigsbee2A model and the initial model (b) for inversion.
Figure 4.8: Two common shot gathers simulated in the true model (Figure 4.7a). The maximum offset is about 3 km.

for the water bottom, are located at inaccurate positions, and some deep reflectors are misplaced by more than the dominant wavelength. In addition, two major faults and two diffractors are poorly imaged in the RTM with the incorrect smooth initial model.

In this example, conventional FWI can only add some details in the scale of one wavelength or less, as indicated by the updated model shown in Figure 4.10(a), which is the inversion result after 10 iterations of conventional FWI. Because of the lack of low wavenumbers in the model (Figure 4.10(a)), the corresponding RTM image shown in Figure 4.10(b) does not provide any noticeable improvement over the initial result in Figure 4.9(b). Figure 4.11(a) and Figure 4.11(b) are two shot gathers simulated in the updated model shown in Figure 4.10(a). Compared to the true gathers in Figure 4.8, we observe that the synthetic data matches the true data well, but only for small offsets. As illustrated in Figure 4.11(c) and Figure 4.11(d), traveltime shifts estimated by dynamic warping between the true data and synthetic data indicate large
Figure 4.9: RTM images produced with the true model (a) and the initial model (b).
misalignments at far offsets. Without low-wavenumber updates, conventional FWI cannot avoid cycle skipping, even after 10 iterations.

However, after only one iteration of conventional FWI (step 1), which adds reflectors to the initial model at possibly incorrect positions, WERTI (step 2) begins to update the low-wavenumber background model and to correct the reflector positions. These two actions take place in the same WERTI iteration (step 2) but in an alternating fashion. To ensure that WERTI updates only the background model, we apply a spatial low-pass filter to the gradient $g_T(m^*)$ to remove any components or artifacts that are close to or less than the dominant wavelength.

Figure 4.12(a) shows the updated low-wavenumber background model after 10 iterations of WERTI (in step 2). With this background model, we can significantly improve the RTM image quality, as illustrated by Figure 4.12(b). Compared to the initial migration image in Figure 4.9(b), Figure 4.12(b) shows that two major faults and diffractors are better resolved, and reflectors are now close to the true positions shown in Figure 4.9(a). We plot the convergence of WERTI in this example in Figure 4.13. The traveltime misfit function (Figure 4.13(a)) decreases rapidly to nearly zero after WERTI updates the low-wavenumber background and corrects reflector positions. Since this is a synthetic example, we are able to also plot a model misfit curve, which also decreases significantly with low-wavenumber updates.

In step 3, we can further improve the WERTI result in Figure 4.12(a) by using it as a new initial model and begin iterations of conventional FWI. In this way, we can refine the low-wavenumber WERTI model with high-wavenumber details. Figure 4.14 shows the final model after 10 iterations of FWI. With the addition of high-wavenumber details, the final model represents most of the major features including faults and diffractors. We once again simulate synthetic data in the final model and show two gathers in Figure 4.15(a) and Figure 4.15(b). Compared to
Figure 4.10: Estimated velocity model (a) with 10 iterations of conventional FWI and the corresponding RTM image (b).
Figure 4.11: (a) - (b): Two common shot gathers simulated in the FWI model (Figure 4.10a). (c) - (d): Corresponding traveltime shifts to the recorded data in Figure 4.8 estimated by dynamic warping. Large time shift indicates the failure of conventional FWI in recovering the background velocity.
Figure 4.12: Estimating velocity model (a) with 10 iterations of WERTI and the corresponding RTM image (b).
Figure 4.13: Convergence of WERTI: traveltime misfit function (a) and model misfit function (b).
the true data in Figure 4.8(a) and Figure 4.8(b), seismograms are well aligned at both near and far offsets. The small traveltime shifts displayed in Figure 4.15(c) and Figure 4.15(d) illustrate this alignment in a quantitatively compelling way. Finally, we compare the convergence of FWI after WERTI with conventional FWI in Figure 4.16. FWI can decrease the data misfit function, but due to cycle skipping, the model misfit function in fact does not decrease. On the other hand, FWI after WERTI decrease both the data misfit and model misfit curves more rapidly.

Figure 4.14: Using the velocity model in Figure 4.12a as the new initial model, conventional FWI refines the velocity model with high-wavenumber details.

4.6 Conclusions

In this paper, we have proposed WERTI to estimate the low-wavenumber components of subsurface velocity models using seismic reflections. WERTI requires no picking and it uses dynamic warping to detect time-varying time shifts, without suffering cycle skipping problems that often plague correlation-based methods. By minimizing only traveltime errors, WERTI can overcome local minima that cannot be avoided in reflection FWI with insufficient low frequencies in recorded data.
Figure 4.15: (a) - (b): Two common shot gathers simulated in the final model (Figure 4.14). (c) - (d): Corresponding travelt ime shifts to the recorded data in Figure 4.8 estimated by dynamic warping.
Figure 4.16: Data misfit (a) and model misfit (b) in conventional FWI and FWI after WERTI.
We design an inversion process by combining WERTI and FWI to estimate the low- and high-wavenumber components of the velocity model alternately. Tests on two synthetic models demonstrate that this process can effectively recover subsurface velocity models using only high-frequency reflection data.

4.7 Acknowledgments

We are grateful to Yi Luo (Saudi Aramco) for insightful discussions and suggestions.
CHAPTER 5
APPLICATION OF IMAGE-GUIDED FULL WAVEFORM INVERSION TO A 2D OCEAN-BOTTOM CABLE DATA SET

A paper to be submitted to Geophysics
Yong Ma\textsuperscript{1,2}, Jerry Yuan\textsuperscript{3}, Yunqing Shen\textsuperscript{3}, and Bin Gong\textsuperscript{4}

5.1 Summary

Image-guided full waveform inversion (IGFWI) was proposed and synthetically tested in Chapter 2. Compared with conventional FWI, IGFWI yields subsurface velocity models with better geologic sense because it takes into consideration of structures of the subsurface, which can be apparent in migrated seismic images. Moreover, IGFWI converges faster in fewer iterations, especially when reflection energy is used to invert for high-wavenumber details in the model. In this chapter, we test IGFWI using a 2D ocean-bottom cable (OBC) data set. In this example, we first employ refraction energy to update the low-wavenumber component of the model and then proceed to invert for high-wavenumber details with reflection data. In the reflection stage, we incorporate structure constraints into inversion and as a result the estimated velocity model and the corresponding RTM image show plausible geologic sense.

5.2 Introduction

Full waveform inversion (FWI) (Tarantola, 1984) uses recorded seismic data $\mathbf{d}$ to estimate parameters of a subsurface model $\mathbf{m}$, by minimizing the difference between recorded data $\mathbf{d}$ and synthetic data $\mathbf{F}(\mathbf{m})$, where $\mathbf{F}$ is a forward operator.

\textsuperscript{1}Center for Wave Phenomena, Colorado School of Mines
\textsuperscript{2}Corresponding author
\textsuperscript{3}ConocoPhillips Company
\textsuperscript{4}Formerly ConocoPhillips Company, presently In-Depth Geophysical Inc.
that synthesizes data. In FWI, the objective function often takes an L2 norm:

$$E(m) = \frac{1}{2}\|d - F(m)\|^2.$$  

A typical implementation of FWI based on conjugate gradients consists of three steps performed iteratively, beginning with an initial model $m_0$:

(i) compute the gradient $g_i$ of the objective function using the adjoint-state method (Tromp et al., 2005);

(ii) search for a step length $\alpha_i$ in the update direction $h_i$;

(iii) update the model with $m_{i+1} = m_i - \alpha_i h_i$.

In each iteration, one needs to find a proper step length $\alpha_i$ that decreases the objective function. A line-search algorithm (Nocedal & Wright, 2000), such as the quadratic line-search method, can be used to find this step length, and the search direction $h_i$ is defined by the conjugate gradient as

$$h_0 = g_0,$$

$$\beta_i = \frac{g_i^T (g_i - g_{i-1})}{g_{i-1}^T g_{i-1}},$$

$$h_i = g_i + \beta_i h_{i-1}. \quad (5.1)$$

FWI is a computationally intensive tool. It requires multiple iterations to minimize data misfit; in each iteration, the cost of both the gradient calculation and the line search is equivalent to the cost of several seismic wavefield simulations and reconstructions, which are especially expensive in 3D.

In addition to its computational cost, FWI also suffers from the nonunique problem. In other words, many different models may yield synthetic data that match recorded data within a reasonable tolerance. This nonuniqueness is caused mainly by local minima in the data misfit function, and the presence of local minima is due to the nonlinearity in the forward operation $F(m)$ and the cycle skipping. Strong
nonlinearity in reflection FWI makes this local-minima problem more severe (Snieder et al., 1989). Cycle skipping occurs if the time delay between synthetic and recorded data is larger than half a period of the dominating wavelet. In practice, the cycle-skipping problem occurs typically because it can be difficult to obtain an adequate initial model that is consistent with unrecorded low frequencies.

Both local-minima and cycle-skipping problems lead to models that poorly approximate the subsurface. To mitigate such problems, multiscale approaches (Boonyasiriwat et al., 2009; Bunks, 1995; Sirgue & Pratt, 2004) have been proposed. The fidelity of multiscale techniques depends fundamentally on the fidelity of low-frequency content in recorded data. In practice, the low frequencies required in the multiscale approach may be unavailable because of noise.

Ma et al. (2012) propose image-guided FWI (IGFWI) to deal with these problems in a sparse model space, where conventional FWI is reposed as a sparse inverse problem. The model \( m \) estimated in conventional FWI is densely sampled for simulating wave propagation. However, the number of dense samples is far beyond the number of necessary samples needed to geologically explain the model. To reduce the number of required iterations in FWI, one may use a sparse representation of the model and reduce the number of model parameters. Therefore, in inversion, one should choose as few samples as possible to construct a sparse model \( s \), while still maintaining as many geological features as possible.

Between the dense model \( m \) and the sparse model \( s \), we define a linear relationship \( m = Rs \), where \( R \) can take different forms. Various bases (e.g., Fourier bases, wavelet bases, and splines) have been used, typically in tomographic studies, to obtain some types of sparse models. Among those methods, the representative one is the wavelet transform (Loris et al., 2007; Meng & Scales, 1996; Simons et al., 2011). However, these types of methods for model parameter reduction do not consider the
structures of the subsurface, and as a result, it risks in generating geologically non-plausible inversion results. In IGFWI, we implement $R$ and $R^T$ with the image-guided interpolation (IGI) (Hale, 2009a) and its adjoint operator (Appendix A), respectively, in order to apply structural constrains derived from migrated images.

In the sparse model space $s$, we represent FWI as a sparse inverse problem, in which we minimize a new objective function: $E(s) = \frac{1}{2} \| d - F(Rs) \|^2$. The nonlinear conjugate-gradient method is still valid to solve for the sparse model $s$ iteratively. The line search for this case is performed in a new update direction $h^s_i$, which is defined as

$$
\begin{align*}
    h^s_0 &= R^Tg_0, \\
    \beta_i &= \frac{(R^Tg_i)^T(R^Tg_i - R^Tg_{i-1})}{(R^Tg_{i-1})^T R^Tg_{i-1}}, \\
    h^s_i &= R^Tg_i + \beta_i h^s_{i-1}. 
\end{align*}
$$

(5.2)

In reality, one needs a dense model $m$ to compute synthetic data $F(m)$ and to fit recorded data $d$. For this reason, we apply the interpolation operator $R$ to both sides of equation 5.2 in IGFWI and thereby interpolate the update direction $h^s_i$ in equation 5.2 to obtain an image-guided update direction $h^m_i$, which corresponds to the dense model space:

$$
\begin{align*}
    h^m_0 &= RR^Tg_0, \\
    \beta_i &= \frac{(R^Tg_i)^T(R^Tg_i - R^Tg_{i-1})}{(R^Tg_{i-1})^T R^Tg_{i-1}}, \\
    h^m_i &= RR^Tg_i + \beta_i h^m_{i-1}. 
\end{align*}
$$

(5.3)

We then can use the image-guided conjugate-gradient method in equation 5.3 to update the dense model.
An implementation of IGFWI based on conjugate gradients consists of four steps performed iteratively, beginning with an initial model $m_0$:

(i) compute the gradient $g_i, R^T g_i, RR^T g_i$;

(ii) search for a step length $\alpha_i$ in the direction $h^m_i$;

(iii) update the model with $m_{i+1} = m_i - \alpha_i h^m_i$;

(iv) remigrate with the updated model and reselect the sparse model based on the remigrated image.

Although the four-step procedure finally gives a dense model solution, it in fact maintains the advantages of sparse inversion. In the sparse inversion, each parameter in the sparse model space $s$ represents an area in the dense model space $m$. This sparse representation corresponds to more blocky updates to the model and the blockiness can mitigate the absence of low frequencies in field data. Because the structure features of the subsurface are taken into consideration as constraints in the sparse inversion, IGFWI generates models that make better geological sense than conventional FWI does. Ma et al. (2012) test IGFWI in the synthetic Marmousi II model. In this chapter, we test IGFWI on a real 2D ocean-bottom cable (OBC) data set.

5.3 Field Data and Source Wavelet

Figure 5.1 shows the source line and the receiver line of this 2D data set in a survey map. The entire 2D data set consists of 24 shots and 235 receivers. Shot spacing and receiver spacing are about 500 m and 50 m, respectively. Sources are 5 m below the water surface; receivers are on the sea floor (70 m beneath the water surface). In this OBC survey, both primary and converted waves are collected; however, we only use the P-wave component of the data set in inversion.
Figure 5.1: One 2D survey line extracted from a 3D OBC survey. The red dot indicates the location of a shot; white dots are receivers spanning about 12 km; green dots are common-depth point locations.
5.3.1 Field Data

Figure 5.2 shows one P-wave shot gather for the 2D data set, which has a largest offset of about 12 km. Due to the large offset of the survey, this data set contains significant refraction energy, especially in the far offset. As illustrated in Figure 5.2, the red line indicates the traveltime of the direct wave; refraction energy arrives earlier than the direct wave. Because of the shallow water depth (70 m), strong surface-related multiples are present in the original data set and no processing is done to remove those surface-related multiples. The original data set contains a very wide frequency band. A bandpass filter is applied to the original data set to obtain the data between 4 Hz and 20 Hz, as shown in Figure 5.2.

5.3.2 Source Wavelet

Estimation of a source wavelet is an essential step in FWI. In this field data test, the source wavelet for inversion is obtained from a source signature measured in the laboratory. Figure 5.3(a) shows the source signature after de-bubbling; its spectrum, as shown in Figure 5.3(b), has a much wider range than the effective frequency band (4 – 20 Hz) of the data set. We apply a minimum phase bandpass filter to the source signature and obtain a source wavelet (Figure 5.4(a)), whose spectrum, shown in Figure 5.4(b), matches the frequency band of the data set.

5.4 Methodology

The survey area is known to have strong vertical transparent isotropy (VTI). Figure 5.5 shows the initial P-wave velocity model that is produced by traveltime tomography and will be used in full waveform inversion. Figure 5.6(a) and Figure 5.6(b) show 1D profiles of the Thomsen parameters \( \epsilon \) and \( \delta \), respectively. These anisotropic parameters are not exact because in fact, they are borrowed from another field that is nearby. In inversion, we fix Thomsen parameters and update only the
Figure 5.2: One typical shot gather for the 2D OBC data set. This shot gather shows an offset of about 12 km. Because of this large offset, significant refraction energy appears at far offset. The red line indicates the traveltime of the direct arrival and energy that arrives earlier than this line is refraction.
Figure 5.3: Source signature (a) that is measured in laboratory and its spectrum (b).
Figure 5.4: Source wavelet (a) that is used for inversion and its spectrum (b). This source wavelet is obtained by applying a bandpass filter to the source signature in (a).
P-wave velocity.

5.4.1 Synthetic Data

VTI has a strong influence on traveltime especially at far offsets. To honor the strong VTI effect of this field in wave propagation, we solve an acoustic VTI wave equation (Chu et al., 2011), which uses the pseudo-acoustic approximation (Alkhalifah, 2000), to simulate the synthetic data. One example of the 2D acoustic VTI wave equation contains two coupled partial differential equations (Du et al., 2008):

$$\frac{\partial^2 p}{\partial t^2} = v_{pz}^2 \frac{\partial^2 p}{\partial x^2} + v_{pz}^2 \frac{\partial^2 q}{\partial z^2}, \quad (5.4)$$

and

$$\frac{\partial^2 q}{\partial t^2} = v_{pn}^2 \frac{\partial^2 p}{\partial x^2} + v_{pz}^2 \frac{\partial^2 q}{\partial z^2}, \quad (5.5)$$

where $p \equiv p(x, z, t)$ and $q \equiv q(x, z, t)$ are P- and S-waves, respectively; $v_{pz} \equiv v_{pz}(x, z)$ is the P-wave velocity (Figure 5.5) along the symmetry axis; $v_{pz} = v_{pz} \sqrt{1 + 2\epsilon}$, and $v_{pn} = v_{pz} \sqrt{1 + 2\delta}$. 

![Figure 5.5: Initial P-wave velocity model.](image)
Figure 5.6: 1D profiles of anisotropy parameters: $\epsilon$ (a) and $\delta$ (b).
In this study, we focus only on the P-wave. Figure 5.7 shows the synthetic P-wave data simulated with the initial velocity and the Thomsen parameters. If we ignore the VTI anisotropy by using an isotropic acoustic wave equation, the synthetic data would have more significant traveltime error in the far offset.

5.4.2 Refraction FWI + Reflection FWI

Because significant refraction energy is available in the far offset, we can first use the refraction energy to update the velocity model. In this stage of FWI, we only match seismograms that arrive earlier than the direct wave, which is indicated by the red line in Figure 5.2. After updating low-wavenumber components of the velocity with refractions, we then use the reflection information in the data set to update high-wavenumber details of the model. In this study, the reflection data that we use for inversion is within the red triangle region, where the maximal zero-offset time is about 3 s.

Conventional FWI is performed to implement the inversion with refraction data. When proceeding to inversion with reflection data, we use IGFWI and compare IGFWI results with conventional FWI results. Both conventional FWI and IGFWI are implemented in the time domain. The time-domain approach is equivalent to invert the entire frequency band (4 – 20 Hz) simultaneously in the frequency domain.

5.5 Inversion with Refraction Data

Figure 5.8(a) displays the estimated velocity model after 5 iterations of FWI with refraction data. Because FWI uses only the refraction energy, it only updates the velocity model with low-wavenumber components, which is indicated by the accumulated velocity update in these 5 iterations, as shown in Figure 5.8(b).
Figure 5.7: Synthetic data simulated with the initial velocity model and anisotropic parameters.
Figure 5.8: Estimated velocity (a) after 5 iterations of refraction FWI and the difference (b) between this estimation and the initial velocity.
5.6 Inversion with Reflection Data

In order to let FWI update the velocity model with high-wavenumber details, we need to use reflection data. In this stage of reflection FWI, we employ the refraction-FWI-updated velocity model (Figure 5.8(a)) as the initial velocity model. Both the conventional FWI and IGFWI are tested using the reflection data.

5.6.1 Conventional FWI

We first compute the gradient of the objective function using the adjoint-state method. In other words, the gradient is achieved by performing a reverse-time migration (RTM) of the data residual \( d - F(m_i) \) with the current velocity model \( m_i \). Figure 5.9 shows an example of the gradient of the objective function in the reflection FWI. This gradient is computed with the velocity in Figure 5.8(a), which is estimated by FWI with the refraction data. We then use a quadratic line search method to update the velocity model. Figure 5.10(a) shows the velocity updated by the conventional FWI, which employs the reflection data in 5 iterations. The difference between the reflection-FWI-updated velocity and the initial velocity is shown in Figure 5.10(b). Different from the refraction inversion (Figure 5.8(b)), the inversion with reflection data updates the velocity model with more details by generating high-wavenumber components, as illustrated by Figure 5.10(b). However, the high-wavenumber details are contaminated by many artifacts that are not geologically sensible.

5.6.2 Image-Guided FWI

The procedure of image-guided FWI (IGFWI) is similar as that of conventional FWI because the nonlinear conjugate-gradient method and the quadratic line search algorithm are the same in both IGFWI and FWI. The difference is that, instead of computing the gradient \( g_i \), IGFWI computes the projected gradient \( R^Tg_i \) and the
Figure 5.9: Gradient $g$ of the objective function in reflection FWI. The velocity for computing this gradient is the refraction-FWI updated model in (a).

The image-guided gradient $RR^T g_i$. In order to do so, we must first know the structure information of the subsurface and then construct a sparse model according to the structure features.

Figure 5.11(a) displays a RTM image of the 2D OBC data, which is migrated with the initial velocity model. On top of the migrated image, ellipses illustrate the structure features of the subsurface, e.g., coherence, orientation, etc.. For this field data example, the subsurface structure turns horizontally coherent in most areas. Figure 5.11(b) shows a sparse model space that is constructed with the structure-constrained selection scheme (Ma et al., 2012).

Using the structure constraints and the sparse model, we compute the image-guided gradient $RR^T g_i$, as shown in Figure 5.12. This image-guided gradient is achieved in two steps. We first apply the adjoint image-guided interpolation operator $R^T$ to the gradient $g_i$ and obtain the projected gradient $R^T g_i$. This step projects (gathers) the gradient information to the chosen sparse model space. We then apply the image-guided interpolation operator $R$ to the projected gradient $R^T g_i$ and get
Figure 5.10: Estimated velocity (a) after 5 iterations of reflection FWI and the difference (b) between this estimation and the initial velocity.
Figure 5.11: Structure features (a) indicated by tensor fields (ellipses) and a sparse model represented by dots (b).
the image-guided gradient $\mathbf{R} \mathbf{R}^T \mathbf{g}_i$. This step interpolates (scatters) the projected gradient from the sparse model space to the dense model space. Compared to the regular gradient (Figure 5.9), the image-guided gradient (Figure 5.12) contains less artifacts and better honors the structure features. Moreover, the gather-scatter process produces low wavenumber components in the image-guided gradient.

![Figure 5.12: Image-guided gradient $\mathbf{R} \mathbf{R}^T \mathbf{g}$ of the objective function in reflection FWI. The velocity for computing this gradient is the refraction-FWI updated model in (a).](image)

The nonlinear conjugate-gradient method (equation 5.3) takes the projected gradient $\mathbf{R}^T \mathbf{g}_i$ and the image-guided gradient $\mathbf{R} \mathbf{R}^T \mathbf{g}_i$ to compute the update direction $\mathbf{h}_i^m$, which is employed in the subsequent quadratic line search. Figure 5.13(a) shows the velocity updated by the image-guided FWI, which uses the reflection data in 5 iterations. The difference between the reflection-IGFWI-updated velocity and the initial velocity is shown in Figure 5.13(b).

### 5.6.3 Discussions

Because of limited data is used in inversion, velocity update mainly occurs in the area above 3 km. Figure 5.14(a), Figure 5.14(b), Figure 5.15(a) and Figure 5.15(b)
Figure 5.13: Estimated velocity (a) after 5 iterations of image-guided reflection FWI and the difference (b) between this estimation and the initial velocity.
compare the initial velocity, refraction-FWI velocity, conventional reflection-FWI velocity, and the image-guided reflection-FWI velocity. The velocity in Figure 5.15(b), which is obtained after image-guided reflection FWI, shows significantly improvement by removing geologically non-interpretable artifacts from the conventional reflection-FWI model (Figure 5.15(a)). Meanwhile, image-guided reflection FWI maintains the high-wavenumber details.

Figure 5.14: Zoom-in views of the initial model (a) and the refraction updated model (b).
Figure 5.15: Zoom-in views of the reflection updated model (a) and the image-guided reflection updated model (b).

In this field data test, the synthetic data takes into account of VTI anisotropy, but the anisotropic parameters may not be exact because they are estimated for another nearby field. Therefore, the estimated velocity model may have errors to trade off the ambiguity between the velocity itself and the anisotropy parameters.

RTM is used to test the inversion results. Figure 5.16(a), Figure 5.16(b), Figure 5.17(a) and Figure 5.17(b) display the RTM images of the 2D OBC data set, which
is migrated with the initial velocity, refraction-FWI velocity, conventional reflection-FWI velocity, and image-guided reflection-FWI velocity, respectively. In the circle-highlighted area of these RTM images have gas clouds been discovered. Despite of the existence of gas clouds, rock layers should maintain the structural continuity. However, broken structures are observed in the highlighted area of Figure 5.16(a) and Figure 5.16(b). In contrast, Figure 5.17(a) and Figure 5.17(b) show more interpretable coherent structures, especially in the highlighted gas cloud area. Moreover, the migrated image in Figure 5.17(b), which is done with the image-guided reflection-FWI velocity, contains less artifacts than Figure 5.17(a).

Each iteration of the IGFWI is more expensive than one iteration of conventional FWI as steps (i) and (iv) bring additional cost. Fortunately, the cost of applying the interpolation and its adjoint operator or reselecting a sparse model is negligible compared to the cost of forward modeling or reverse time migration. Also, it is not necessary to reselect the sparse model in every iteration, and therefore by applying the last step only in selected iterations, we can further reduce the additional cost.

5.7 Conclusions

In this chapter we have demonstrated the capability of IGFWI in estimating velocity models with a 2D OBC data set. The refraction data and the reflection data are sequentially used to update the velocity model. Compared to conventional reflection FWI, IGFWI with reflection data, which essentially solves a sparse inverse problem, generates velocity models that make better geological sense. This improvement is caused by the fact that IGFWI uses the subsurface structures, which are extracted from the migrated seismic image in this study, to constrain the inversion in the sparse model space. We compared RTM images of the 2D OBC data using velocity models that are updated by conventional FWI and IGFWI, respectively. With the IGFWI-estimated velocity, RTM images show more interpretable coherent structures in the
Figure 5.16: RTM with the initial model (a) and the refraction-FWI model (b).
Figure 5.17: RTM with the reflection-FWI model (a) and the image-guided reflection-FWI model (d).
complex area.

5.8 Acknowledgments

Thank ConocoPhillips for the permission to publish this work, which was finished during Yong Ma’s internship with the Seismic Technology group at ConocoPhillips in summer 2011. We want to thank Yanqing Zeng for his generous help to make RTM images in this report. We are grateful to Yi Wang and Chunlei Chu for the reference to the acoustic VTI modeling that is employed in this study. We also want to thank Kirk Wallace for his help to generate common-image gathers, which are used intermediately to QC the inversion.
6.1 Main Results

In this thesis, I develop efficient and effective methods for estimating subsurface velocity models. The main results of this thesis are summarized as follows.

6.1.1 Structure-Constrained Inversion

In Chapter 2, I propose to use structural information of the subsurface in FWI as \textit{a priori} constraints in order to yield geologically plausible inversion results. By using the migrated seismic image as the source of structural constraints, I design image-guided FWI (IGFWI). I test IGFWI using both the synthetic Marmousi II data (Chapter 2) and a real 2D OBC data (Chapter 5). Compared to conventional unconstrained FWI, IGFWI provides geologically plausible velocity models that contains fewer geologically unreasonable artifacts, thereby making geologic interpretation easier.

6.1.2 Sparse-Model Inversion

I design a structurally-constrained sample selection method in Chapter 2 in order to construct a sparse model that contains significantly fewer parameters than does the original model. With fewer parameters in the sparse model, I reformulate conventional FWI as a linearly-constrained sparse-model FWI, which can converge in fewer iterations than conventional FWI. In addition, this sparse representation yields low-wavenumber blocky models, which mitigate the lack of the low frequencies in recorded data and improve convergence. I solve the sparse-model FWI with an image-guided conjugate-gradient method and test this sparse inversion with both synthetic and field data in Chapters 2 and 5, respectively.
6.1.3 Projected Hessian and Efficient Quasi-Newton Method

In Chapter 3, I solve the sparse-model FWI problem by using a quasi-Newton method, which requires a projected Hessian and its inverse matrix. In conventional FWI, the Hessian and its inverse are too expensive to compute, and many methods can only implicitly approximate the Hessian or its inverse. In Chapter 3, I adapt the classic L-BFGS method to a projected BFGS (P-BFGS) method, which, with respect to the sparse model, can reduce computation time and memory consumption substantially. As a result, one can explicitly compute a projected Hessian or its inverse. The quasi-Newton FWI in Chapter 3 shows a faster convergence than the previous conjugate-gradient methods in Chapter 2.

6.1.4 Integration of Traveltime Inversion and FWI

In Chapter 4, I investigate wave-equation reflection traveltime inversion (WERTI) to resolve low-wavenumber components of velocity models. In WERTI, I estimate traveltime shifts with dynamic warping, which provides more accurate time-varying traveltime shifts than conventional crosscorrelation-based approaches do. I design an alternating inversion strategy that combines WERTI and FWI, in order to estimate the low- and high-wavenumber components. This strategy solves both the cycle-skipping problem in conventional FWI and the velocity-depth ambiguity in reflection traveltime inversion. Using this alternating approach, moreover, one can successfully resolve both the low- and high-wavenumber components of velocity models with only high-frequency reflection data.

6.2 Suggestions

In this thesis, I mainly investigate 2D velocity inversion using seismic reflection data. I also note some potential research directions for future work.
6.2.1 Different Constraints: From Structure to Geology

In this thesis, I use imaged subsurface structures as constraints to generate geologically plausible models. However, these structural constraints rely on the quality of migrated images, which, due to complex geology, may fail to provide reliable structural information to constrain inversion. In addition, model parameters do not necessarily coincide with structures, and therefore in some cases (e.g., velocity anomaly caused by overpressure), structural features are not useful in constraining inversion. When structural constraints are unavailable or not helpful, other priori information such as well-log measurements and even geologic interpretations can be employed as additional constraints, so that structure-constrained FWI can evolve to truly geology-constrained FWI.

6.2.2 Multiparameter and 3D Inversion in Sparse Model

This thesis focuses on acoustic constant-density FWI, which therefore inverts only for the P-wave velocity of the subsurface. In fact, the subsurface is more accurately characterized by an elastic, anisotropic and dispersive model with varying densities. In principle, FWI can invert for multiple parameters, such as P- and S-wave velocities, density, Thomsen parameters and attenuation. However, inverting for multiple parameters makes FWI even more computationally intensive. By using the ideas of a sparse model and sparse-model FWI proposed in Chapter 2, one can reduce the model space for each parameter and therefore can reduce the cost of FWI.

Large-scale 3D FWI is increasingly attractive but is probably still limited by the computation burden. The 2D sparse model in this thesis is constructed based on the structural sparsity, which is generally more significant in 3D models because subsurface geology is often highly correlated in both inline and crossline directions. Therefore, the sparse-model FWI can be extended to 3D cases, where the advantages
can be more significant.

6.2.3 Structure-Constrained Joint Inversion

Seismic data and nonseismic data (e.g., gravity, magnetic, electromagnetic) can provide different estimates of physical properties of the earth. In many cases, different geophysical data are complimentary. On the one hand, seismic data, for instance, can reveal the subsurface structure with a high-resolution image, which merely indicates reflectivity but cannot differentiate between the hydrocarbon reservoir and water. On the other hand, electromagnetic data can distinguish hydrocarbon from water but cannot provide high-resolution information. Joint inversion of different geophysical data can provide an integrated understanding of the subsurface. In particular, structure-constrained joint inversion can use high-resolution subsurface structures provided by seismic data to constrain the inversion of nonseismic data. In this way, structure-constrained joint inversion can decrease risk in hydrocarbon exploration.
REFERENCES CITED


A.1 Image-Guided Interpolation

We follow the steps in Hale (2009a) to describe the details of nearest neighbor interpolation $P$ and blended neighbor interpolation $Q$:

1. $P$: solve

$$\nabla t(x) \cdot D(x) \cdot \nabla t(x) = 1, x \notin \chi ;$$

$$t(x) = 0, x \in \chi$$ \hspace{1cm} (A.1)

for

$t(x)$: the minimum time from $x$ to the nearest known sample point $x_k$, and

$p(x)$: the nearest neighbor interpolant corresponding to $f_k$, the value of the sample point $x_k$ nearest to the point $x$.

2. $Q$: for a specified constant $e \geq 2$ ($e = 4$ in this paper), solve

$$q(x) - \frac{1}{e} \nabla \cdot t^2(x) D(x) \cdot \nabla q(x) = p(x)$$ \hspace{1cm} (A.2)

for the blended neighbor interpolant $q(x)$.

In equation A.1, the metric tensor field $D(x)$ (Fehmers & Höcker, 2003; van Vliet & Verbeek, 1995) represents structural features of the subsurface, such as structural orientation, coherence, and dimensionality. In $n$ dimensions, each metric tensor field $D$ is a symmetric positive-definite $n \times n$ matrix (Hale, 2009a). Here, the minimum time $t(x)$ is a non-Euclidean distance between a sample point $x_k$ and an interpolation point $x$. By this measure of distance, we say that a sample point $x_k$ is nearest to a point $x$ if the time $t(x)$ to $x_k$ is less than $t(x)$ to any other sample point.
A.2 Adjoint Image-Guided Interpolation

Letting $p$ and $q$ denote vectors that contain all values in $p(x)$ and $q(x)$, respectively, we can rewrite equation A.2 in a matrix-vector form:

$$(I + B^TDB)q = p,$$  \hspace{1cm} (A.3)

where $B$ corresponds to a finite-difference approximation of the gradient operator (Hale, 2009b). Therefore, $q = Qp$, where

$$Q = (I + B^TDB)^{-1},$$ \hspace{1cm} (A.4)

and this inverse can be efficiently approximated by conjugate-gradient iterations because $I + B^TDB$ is symmetric and positive-definite (SPD).

Note that $Q^T = Q$, so we can write the adjoint image-guided interpolation as

$$R^T = P^TQ^T = P^TQ.$$ \hspace{1cm} (A.5)
APPENDIX B - EUCLIDEAN DISTANCE TRANSFORM

The distance transform (DT) (Fabbri et al., 2008) computes the distance of each pixel of an image to a given subset of pixels. Let \( I : \Omega \subset \mathbb{Z}^2 \rightarrow \{0, 1\} \) represent a binary image (e.g., Figure B.1(a)) where the domain \( \Omega = \{1, 2, \ldots, n_1\} \times \{1, 2, \ldots, n_2\} \). In image processing, 1 is associated with white, and 0 with black. Hence, two sets can be defined in the following way:

\[
\mathcal{O} = \{ x \in \Omega \mid I(x) = 1 \}, \quad (B.1)
\]

and its complement

\[
\mathcal{O}^c = \{ x \in \Omega \mid I(x) = 0 \}. \quad (B.2)
\]

In image processing literature, the set \( \mathcal{O} \) is referred to as object or foreground and can consist of any subset of the image domain \( \Omega \). The set of black pixels \( \mathcal{O}^c \) is called background. In the DT, \( \mathcal{O}^c \) is the given subset.

DT is, thereby, the transformation that produces a map \( d(x) \), which shows the smallest distance from this pixel \( x \) to \( \mathcal{O}^c \):

\[
d(x) := \min \{ l(x, y) \mid y \in \mathcal{O}^c \}, \quad (B.3)
\]

where the DT kernel \( l(x, y) \) can take different forms, but \( l(x, y) \) is usually the Euclidean distance, defined as \( l(x, y) = \|x - y\|_2 \). In this article, for simplicity we employ distance square \( l(x, y) = (\|x - y\|_2)^2 \). Figure B.1(b) shows the distance square map computed for the binary image in Figure B.1(a).
Figure B.1: A binary image (a) and its distance transform (b).
APPENDIX C - FRECHÉT DERIVATIVES

C.1 With respect to $m^r$: $\frac{\partial p_c(x_r, t; x_s)}{\partial m^r}$

We follow the method in the appendices of Luo & Schuster (1991) and Tarantola (1984) to obtain the Frechét derivative $\frac{\partial p_c(x_r, t; x_s)}{\partial m^r}$.

$$\left( m^s \frac{\partial^2}{\partial t^2} + \nabla^2 \right) p(x, t; x_s) = f(t; x_s) , \quad (C.1)$$

The solution to equation C.1 is

$$p(x, t; x_s) = G(x, t; m^s, x_s) \ast f(t; x_s) , \quad (C.2)$$

where $G(x, t; m^s, x_s)$ is the Green’s function for equation C.1 and $\ast$ denotes the time convolution.

A rough-model perturbation $\delta m^r$ generates a perturbation $\delta p(x, t; x_s)$:

$$\left( (m^s + \delta m^r) \frac{\partial^2}{\partial t^2} + \nabla^2 \right) (p(x, t; x_s) + \delta p(x, t; x_s)) = f(t; x_s) . \quad (C.3)$$

Subtracting equation C.1 from equation C.3, we obtain

$$\left( m^s \frac{\partial^2}{\partial t^2} + \nabla^2 \right) \delta p(x, t; x_s) = -\delta m^r \ddot{p}(x, t; x_s) . \quad (C.4)$$

Using the Green’s function, the solution to equation C.4 can be written as

$$\delta p_c(x_r, t; x_s) = -G(x_r, t; m^s, x) \ast \delta m^r(x) \ddot{p}(x, t; x_s) . \quad (C.5)$$

Dividing both sides of equation C.5 with $\delta m^r$, we obtain the Frechét derivative:

$$\frac{\partial p_c(x, t; x_s)}{\partial m^r} = -G(x_r, t; m^s, x) \ast \ddot{p}(x, t; x_s) \quad (C.6)$$

$$= -\dot{G}(x_r, t; m^s, x) \ast \dot{p}(x, t; x_s) ,$$

which is equivalent to equation A-6 in Luo & Schuster (1991).
C.2 With respect to $m^s$: \[
\frac{\partial p_c(x_r, t; x_s)}{\partial m^s}
\]

For simplicity, we first rewrite equation C.5 in the frequency domain:

\[
\delta p_c(x_r, \omega; x_s) = \omega^2 G(x_r, \omega; m^s, y) \delta m^r(y) G(y, \omega; m^s, x_s) f(\omega; x_s) = \delta G(x_r, \omega; \delta m^r, x_s) f(\omega; x_s),
\]

where $\delta G(x_r, \omega; \delta m^r, x_s) = \omega^2 G(x_r, \omega; m^s, y) \delta m^r(y) G(y, \omega; m^s, x_s)$ is the perturbation of the Green’s function $G(x_r, \omega; m^s, x_s)$ caused by $\delta m^r$.

Then the Frechet derivative $\frac{\partial p_c(x_r, \omega; x_s)}{\partial m^s}$ can be written as

\[
\frac{\partial p_c(x_r, \omega; x_s)}{\partial m^s} = \underbrace{\omega^2 \frac{\partial G(x_r, \omega; m^s, y)}{\partial m^s} \delta m^r(y) G(y, \omega; m^s, x_s) f(\omega; x_s)}_{R_1} + \underbrace{\omega^2 G(x_r, \omega; m^s, y) \frac{\partial G(y, \omega; m^s, x_s)}{\partial m^s} f(\omega; x_s)}_{R_2}.
\]

Xu et al. (2012) introduce the Born approximation to the perturbation of Green’s function $G(x_r, \omega; m^s, x_s)$ caused by the model perturbation $\delta m^s$:

\[
\delta G(x_r, \omega; \delta m^s, x_s) = \omega^2 G(x_r, \omega; m^s, x) \delta m^s(x) G(x, t; m^s, x_s),
\]

Dividing both sides of equation C.9 with $\delta m^s$, we obtain

\[
\frac{\partial G(x_r, \omega; m^s, x_s)}{\partial m^s} = \omega^2 G(x_r, \omega; m^s, x) G(x, t; m^s, x_s).
\]

Similarly,

\[
\frac{\partial G(x_r, \omega; m^s, y)}{\partial m^s} = \omega^2 G(x_r, \omega; m^s, x) G(x, t; m^s, y)
\]

and

\[
\frac{\partial G(y, \omega; m^s, x_s)}{\partial m^s} = \omega^2 G(y, \omega; m^s, x) G(x, t; m^s, x_s).
\]

Substituting equations C.11 and C.12 into equation C.8, we have
\[ R1 = \omega^4 G (x_r, \omega; m^s, x) G (x, t; m^s, y) \delta m^r (y) G (y, \omega; m^s, x_s) f (\omega; x_s) \tag{C.13} \]

and

\[ R2 = \omega^4 G (x_r, \omega; m^s, y) \delta m^r (y) G (y, \omega; m^s, x) G (x, t; m^s, x_s) f (\omega; x_s) \tag{C.14} \]

Using equation C.7, we can simplify equation C.13 as

\[ R1 = \omega^2 G (x_r, \omega; m^s, x) \delta p (x, t; x_s) \tag{C.15} \]

Likewise, using \( \delta G (x_r, \omega; \delta m^r, x) = \omega^2 G (x_r, \omega; m^s, y) \delta m^r (y) G (y, \omega; m^s, x) \) and \( p (x, t; x_s) = G (x, t; m^s, x_s) f (\omega; x_s) \), equation C.14 becomes

\[ R2 = \omega^2 \delta G (x_r, \omega; \delta m^r, x) p (x, t; x_s) \tag{C.16} \]

Finally, the Frechét derivative \( \frac{\partial p_c (x_r, \omega; x_s)}{\partial m^s} \) is

\[
\frac{\partial p_c (x_r, \omega; x_s)}{\partial m^s} = \omega^2 G (x_r, \omega; m^s, x) \delta p (x, \omega; x_s) \\
+ \omega^2 \delta G (x_r, \omega; \delta m^r, x) p (x, \omega; x_s) \tag{C.17}
\]

The time-domain expression of the Frechét derivative \( \frac{\partial p_c (x_r, t; x_s)}{\partial m^s} \) becomes

\[
\frac{\partial p_c (x_r, t; x_s)}{\partial m^s} = -\dot{G} (x_r, t; m^s, x) * \delta p (x, t; x_s) \\
- \delta G (x_r, t; \delta m^r, x) * \dot{p} (x, t; x_s) \tag{C.18}
\]