UNIVERSITY OF CALGARY

Convergence of a full waveform inversion scheme based on PSPI migration and forward modeling-free approximation: procedure and validation

by

Marcelo Guarido de Andrade

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN GEOLOGY AND GEOPHYSICS

CALGARY, ALBERTA

August, 2017

Abstract

Full waveform inversion is a least squares technique that estimates rocks parameters by finding the model that reduces the difference between synthetic and acquired seismic data. This project is focused on acoustic inversion. The main goal is to understand and develop an FWI algorithm that is cheap and provides a higher quality inversion. The steepest-descent method is shown to be robust with some level of guarantee to converge, when the geology is simple, and when the starting model is close to the global minimum. Understanding that the gradient can be estimated by any pre-stack depth migration of the residuals, the RTM (reverse-time migration) is replaced by a PSPI (phase shift plus interpolation) migration, to reduce cost. Even though the method was shown to be expensive, mostly by the number of synthetic shots required and the pre-stack migration. By using a monochromatic averaged gradient combined with a conjugate gradient algorithm, inversion is possible for more complex geology, like a simulation on a 2D acoustic Marmousi model. However, the cost to run the process increased significantly, as each frequency on a selected band is migrated separately to form a pseudo-gradient, and are weighted averaged for the update. Inversion is simplified by applying an impedance inversion on the reflection coefficients based gradient using a band-limited impedance inversion (BLIMP) algorithm. Cost is reduced, being comparable to the standard steepest-descent, and the inverted model resolution is kept similar to the one using the monochromatic averaged gradient. Later, we came with a new interpretation of the gradient, understood to be the difference between current model and the impedance inversion of the migrated acquired data. No forward modeling or source estimation are required to compute the gradient. The cost per iteration is the same of the PSPI migration. A post-stack method is shown to be even cheaper and promising. Using a well sonic log to calibrate the gradient reduces method cost and improves its resolution, and is our most impressive solution.

Acknowledgements

I would like to express my gratitude to my supervisors, Dr. Larry Lines and Dr. Rob Ferguson, for all the leading, support, knowledge share, patience (a lot of it), coffee breaks, and great experience. I would like to thank the other members of my committee, Dr. Kris Innanen, and Dr. Don Lawton for the assistance they provided at all levels of the research project.

A very special thanks goes to Laura Baird. She helped and supported me on every moment of my Ph.D. program, the good and the bad ones. And, of course, all the good talks we had.

I thank CREWES for all financial and technical support, as well its sponsors, and NSERC (Natural Science and Engineering Research Council of Canada) through the grant CRDPJ 461179-13. I would like to thank the Department of Geoscience for the international differential tuition fees support.

I must also acknowledge my friend and colleagues in CREWES, specially Dr. Babatunde Arenrin, Tianci Cui, Dr. Shahin Jabbari, Raul Cova and Khaled Al Dulaijan, for all the good moments and very helpful discussions.

I can not forget to thank Dr. Gary Margrave. He was always open for some really good discussions and suggestions to my project, leading me to great solutions and results.

I would like to thank my parents for giving all the education, support, and the incentive to study and follow a career. Of course, also for all the love they gave me while I grew up.

At last, I would like to thank my family: Pipoca, Lyon, and, the most important person to me and the love of my live, my wife Soane Mota dos Santos. Without her I wouldn't have reached the achievements I accomplished in my career. She is my support, my partner, my best friend, and my counselor. Thanks for being on my side.

Table of Contents

List of Figures and Illustrations

List of Symbols, Abbreviations and Nomenclature

Symbol	Definition
U of C	University of Calgary
Ph.D	Doctor of Philosophy
FWI	Full waveform inversion
PSPI	Phase-shift-plus-interpolation
RTM	Reverse-time migration
BLIMP	Band-limited impedance inversion
AVO	Amplitude versus offset
Φ	Objective function
C_m	Misfit function
H	Hessian
α	Step length
G	Green's function
$\mathbf{J}$	Frèchet derivative matrix (Jacobian)
ω	Frequency
∇	Gradient
∂	Partial derivative
$\Re$	Real part
BPTR	Backpropagated time-reverse
$\mathbf{B}$	Forward modeling operator
NMO	Normal Moveout
CMP	Commom-mid point

Chapter 1

Introduction

Seismic inversion techniques are the ones that use intrinsic information contained in the data to determine rock properties by matching a model response that ”describes” the data. Some examples are the variation of amplitude versus offset, or AVO (Shuey, 1985; Fatti et al., 1994), the traveltime differences between traces, named traveltime tomography (Langan et al., 1984; Bishop and Spongberg, 1984; Cutler et al., 1984), or even by matching synthetic data to the observed data, as it is done in full waveform inversion (Tarantola, 1984; Pratt et al., 1998; Virieux and Operto, 2009; Margrave et al., 2010), among others. These inversions can compute rock parameters such as P and S waves velocities, density, viscosity and others. In this work we are focused in the inversion of the P wave velocity.

FWI is a least squares based inversion, whose objective is to find the model parameters that minimizes the difference between observed (acquired) and synthetic data (Margrave et al., 2011), called the *residuals*. This is accomplished in an iterative fit method by linearizing a non-linear problem. It is a machine learning method very similar to a *Ridge Regression* (Chipman, 1999), which minimizes non-linear problems by adding a regularization term to avoid over fitting (smoothing the model). In seismic processing, we can regularize the inversion by convolving the model with a 2D Gaussian window (Margrave et al., 2010).

The full waveform inversion was proposed in the early 80’s, as reviewed by Pratt et al. (1998), but the technique was considered too expensive in computational terms. Lailly (1983) and Tarantola (1984) simplified the methodology by using the steepest-descent method (or gradient method) in the time domain to minimize the objective function without calculating, explicitly, the partial derivatives. They show that the gradient is equivalent to a reverse-time migration (RTM) of the residuals. Pratt et al. (1998) develop a matrix formulation for the full waveform inversion in the frequency domain and presented more efficient ways to compute the gradient and the inverse of the Hessian matrix (the sensitivity matrix), using the Gauss-Newton, and the Newton approximations. The FWI is shown to be more efficient if applied in a multi-scale method, where lower frequencies are inverted first, and higher frequencies are inverted as more iterations are done (Pratt et al., 1998; Virieux and Operto, 2009; Margrave et al., 2010). An overview of the FWI theory and studies is compiled by Virieux and Operto (2009). Lindseth (1979) showed that an impedance inversion from seismic data alone is not effective due to the lack of low seismic frequencies during the acquisition but could be compensated by the match with a sonic log profile. Margrave et al. (2010) and Romahn and Innanen (2016) used a gradient method and matched it with sonic log profiles to compensate the absence of the low frequency and to calibrate the model update by computing the step length and a phase rotation (avoiding cycle skipping). Margrave et al. (2010) also proposed the use of a PSPI (phase-shift-plus-interpolation) migration (Ferguson and Margrave, 2005) instead of the RTM, so the iterations are done in time domain but only selected frequency bands are migrated, using a deconvolution imaging condition (Margrave et al., 2011; Pan et al., 2013) as a better reflectivity estimation. Warner and Guasch (2014) use the deviation of the Wiener filters of the real and estimated data as the objective function, which provides very accurate models.

The present project is focused on the implementation of the steepest descent (gradient) method. I am presenting my tests and results following the order how developments were implemented during the Ph.D program.

The starting point of the research is presented in chapter 2, showing the results of the gradient method using the PSPI migration (Ferguson and Margrave, 1996, 2005) with a deconvolution imaging condition to backpropagate the residuals (Margrave et al., 2010; Guarido et al., 2014) on a simple velocity model. The method is applied in time domain where a 2D finite difference algorithm generates the synthetic data but only a selected frequency band is migrated (thanks to the way PSPI migration works), and a line search is used for the step length estimation. Lately, the algorithm is tested on the Marmousi model (Versteeg, 1991), which becomes the main testing model for the next chapters.

Following chapter 3, it is proposed that the gradient can be computed by taking the average of each frequency migrated separately (Guarido et al., 2015). Improvement also came by the implementation of a conjugate gradient algorithm (Zhou et al., 1995; Vigh and Starr, 2008). The step length is estimated by a single forward modeling using Pica et al. (1990)'s approximation.

In chapter 4, I introduce the *band limited impedance inversion* (BLIMP) to transform a reflection coefficient gradient (migration output) to a velocity update that is optimized by the step length (Guarido et al., 2016).

The most promising results of the research are presented on chapters 5, 6 and 7, with a new interpretation of the gradient and the possibility for a forward modeling-free FWI. In chapter 5, it is proposed a new approximation for the gradient that doesn't require forward modeling and source estimation while computing the residuals: we understood it as the impedance (velocity) difference between the acquired data with BLIMP applied and the current model (Guarido et al., 2016), reducing time and resource required to apply the FWI routine in the Marmousi simulation.

In chapter 6 the understanding of the forward modeling-free gradient allowed the method to be expanded to post-stack solution, allowing the routine to be applied by anyone with a simple laptop.

Calibrating the gradient by using a sonic log as a pilot (Margrave et al., 2010; Romahn and Innanen, 2016) combines with the pre and post-stack forward modeling-free gradient methods, to create a method 100% forward modeling-free, and it is presented in chapter 7. In the same chapter, we also show how the improvement of the velocity model after the FWI can provide a better migration velocity.

Finally, this thesis ends with the conclusions and an analysis of the time/resolution trade-off of each methodology tested.

Chapter 2

First analysis of the full waveform inversion method

This section presents the basic mathematical approximation of a full waveform inversion that to estimates the gradient from the objective function by a *reverse time migration* (RTM), the matrix representation for the Newton, Gauss-Newton and gradient (steepest descent) methods and the deconvolution imaging conditions for the gradient.

Analysis and validation of the method is done on a simple acoustic model using the steepest descent approximation. Residuals are migrated with a *phase-shift plus interpolation* (PSPI) migration, with a deconvolution imaging condition, instead of the RTM. I do an analyze in how the migrated residuals must be muted before stacking to avoid the addition of artifacts in the inverted model. The step length is estimated by trial and error and interpolation.

The inverted models shown high resolution on a simple geology model, but its accuracy varies with the quality of the initial model. I also tested the algorithm in the Marmousi model, showing convergence of the objective function with the divergence of the inverted model, suggesting the minimization to a local minimum.

2.1 Theory

2.1.1 The objective function

The goal of the FWI is to minimize the difference between the acquired data and the synthetic data obtained from an initial guess model by iteratively updating the model. The misfit function, or residual wavefield, is:

$$\delta P(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega | s_0^{(n)}) \equiv P(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega) - G(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega | s_0^{(n)}) \tag{2.1}$$

where $P(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega)$ is the acquired data, $G(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega | s_0^{(n)})$ is the synthetic data, $s_0^{(n)}(\boldsymbol{r}) \equiv 1/c_{0,n}^2(\boldsymbol{r})$ is the squared-slowness in the *nth* iteration, $c_0^2(\boldsymbol{r})$ is the velocity, $\boldsymbol{r}_g$ and $\boldsymbol{r}_s$ are the receiver and source position vectors respectively, and ω is the angular frequency. On each iteration, I update s_0 until the objective function Φ is minimized:

$$\Phi\left(s_0^{(n)}\right) \equiv \frac{1}{2} \int \left(\sum_{s,g} |\delta P|^2 \right) d\omega \tag{2.2}$$

Margrave et al. (2011) derive the parameter update $\delta s_0^{(n)}(\boldsymbol{r''})$ by calculating the first derivative of $\Phi(s_0^{(n)} + \delta s_0^{(n)})$ and setting it to zero. The expression is:

$$\delta s_0^{(n)}(\boldsymbol{r''}) = -\int {H^{(n)}}^{-1}(\boldsymbol{r''}, \boldsymbol{r'}) g^{(n)}(\boldsymbol{r'}) d\boldsymbol{r'} \tag{2.3}$$

where

$$g^{(n)}(\boldsymbol{r'}) = \frac{\partial \Phi\left(s_0^{(n)}\right)}{\partial s_0^{(n)}(\boldsymbol{r'})} \tag{2.4}$$

and

$$\begin{aligned} g^{(n)}(\boldsymbol{r'}) &= \frac{\partial \Phi\left(s_0^{(n)}\right)}{\partial s_0^{(n)}(\boldsymbol{r'})} \\ &= \frac{1}{2} \sum_{s,g} \int \Re \left\{ \frac{\partial G(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega | s_0^{(n)})}{\partial s_0^{(n)}(\boldsymbol{r})} \delta P^*(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega | s_0^{(n)}) \right\} d\omega \end{aligned} \tag{2.5}$$

is the gradient. The Hessian is

$$H^{(n)}(\boldsymbol{r''}, \boldsymbol{r'}) = \frac{\partial^2 \Phi\left(s_0^{(n)}\right)}{\partial s_0^{(n)}(\boldsymbol{r''}) \partial s_0^{(n)}(\boldsymbol{r'})} \tag{2.6}$$

where $\boldsymbol{r}$, $\boldsymbol{r'}$ and $\boldsymbol{r''}$ are arbitrary positions into the slowness model. The Hessian defines the curvature of the misfit function at $\boldsymbol{m}_0$.

Margrave et al. (2011) compute the gradient in terms of the Green's function applying a small perturbation $\delta s_0^{(n)}(\boldsymbol{r})$ leading to small changes in the field, which I represent by $\delta G(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega | s_0^{(n)})$. Now, I have the relationship:

$$\lim_{\delta s \to 0} \frac{\delta G}{\delta s} = \frac{\partial G(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega | s_0^{(n)})}{\partial s_0^{(n)}(\boldsymbol{r})} = -\omega^2 G(\boldsymbol{r}_g, \boldsymbol{r}, \omega | s_0^{(n)}) G(\boldsymbol{r}, \boldsymbol{r}_s, \omega | s_0^{(n)}) \tag{2.7}$$

Substituting equation 2.7 into equation 2.5, the gradient can be written:

$$g^{(n)}(\boldsymbol{r}) = \sum_{s,g} \int \omega^2 [G(\boldsymbol{r}, \boldsymbol{r}_s, \omega | s_0^{(n)})] \times [G(\boldsymbol{r}_g, \boldsymbol{r}, \omega | s_0^{(n)}) P^*(\boldsymbol{r}_g, \boldsymbol{r}_s, \omega | s_0^{(n)})] d\omega \tag{2.8}$$

Equation 2.8 can be interpreted as a depth migration of the data residuals (more specifically, a reverse-time migration). Margrave et al. (2010) use a similar representation for the gradient but for the time domain and recognize that the gradient is a pre-stack reverse-time migration (RTM) with a cross-correlation imaging condition.

In the next section, I present a matrix approximation for the FWI and the Newton, Gauss-Newton, and gradient approximations of the Hessian.

2.1.2 The Newton, Gauss-Newton and gradient methods

The objective of the FWI is to find a model that generates synthetic data equal to the observed data (where the misfit function vanishes).

Rewriting equation 2.1 as a misfit vector, I have $\Delta \mathbf{d} = \mathbf{d}_{obs} - \mathbf{d}_{syn}(\mathbf{m})$ of dimension N, where $\mathbf{d}_{obs}$ is the acquired data, $\mathbf{d}_{syn}$ is the data prediction (forward modeling) and $\mathbf{m}$ is the model. The minimization of the misfit function is done using least-squares method. The L2-norm of the misfit is:

$$C(\mathbf{m}) = \frac{1}{2} \Delta \mathbf{d}^{\dagger} \Delta \mathbf{d} \tag{2.9}$$

where † denotes the transpose conjugate.

Virieux and Operto (2009) minimize equation 2.9 with a second-order Taylor-Lagrange expansion around a initial model $\mathbf{m}_0$ plus a perturbation model $\Delta\mathbf{m}$ (the updated model is $\mathbf{m} = \mathbf{m}_0 + \Delta\mathbf{m}$), with L2-norm as $C(\mathbf{m}_0 + \Delta\mathbf{m})$. They compute the derivative with respect to the model parameter m and set it equal to zero to find the minimum. The solution is written as:

$$\Delta\mathbf{m} = -\left[\frac{\partial^2 C(\mathbf{m}_0)}{\partial \mathbf{m}^2}\right]^{-1} \frac{\partial C(\mathbf{m}_0)}{\partial \mathbf{m}} \tag{2.10}$$

On the second part of the equation 2.10, the first derivative of the objective function is the gradient and can be written as:

$$\nabla C_{\mathbf{m}} = \frac{\partial C(\mathbf{m}_0)}{\partial \mathbf{m}} = -\Re\left[\mathbf{J}^{\dagger}\Delta\mathbf{d}\right] \tag{2.11}$$

where $\mathbf{J}$ is the sensitivity or the Frèchet derivative matrix (Jacobian) and $\Re$ denotes the real part of the gradient.

The first term of equation 2.10 is found by a differentiation of equation 2.11 and evaluating it at $\mathbf{m} = \mathbf{m}_0$:

$$\frac{\partial^2 C(\mathbf{m}_0)}{\partial \mathbf{m}^2} = \Re\left[\mathbf{J}_0^{\dagger}\mathbf{J}_0\right] + \Re\left[\frac{\partial \mathbf{J}_0^T}{\partial \mathbf{m}^T}(\Delta\mathbf{d}_0^* \ldots \Delta\mathbf{d}_0^*)\right] \tag{2.12}$$

where the symbol $*$ denotes complex number and $(\Delta\mathbf{d}_0^* \ldots \Delta\mathbf{d}_0^*)$ is a vector with the same size of parameters $\mathbf{m}$. More details can be found in Pratt et al. (1998). Substituting equations 2.11 and 2.12 into equation 2.10:

$$\Delta\mathbf{m} = -\left\{\underbrace{\Re\left[\mathbf{J}_0^{\dagger}\mathbf{J}_0 + \frac{\partial \mathbf{J}_0^T}{\partial \mathbf{m}^T}(\Delta\mathbf{d}_0^* \ldots \Delta\mathbf{d}_0^*)\right]}_{\text{Hessian}}\right\}^{-1} \underbrace{\Re\left[\mathbf{J}^{\dagger}\Delta\mathbf{d}\right]}_{\text{Gradient}} \tag{2.13}$$

This result is known as the *Newton method* and has a quadratic local convergence. This solution is used for nonlinear problems (as in most seismic problems), but demands a big effort to be calculated (Pratt et al., 1998).

As an approximation of equation 2.13, the problem is assumed to be linear and the second term of the Hessian vanishes, leading to the solution:

$$\Delta \mathbf{m} = -\left\{ \Re \left[\mathbf{J}_0^{\dagger} \mathbf{J}_0 \right] \right\}^{-1} \Re \left[\mathbf{J}^{\dagger} \Delta \mathbf{d} \right] \tag{2.14}$$

This approximation is known as the *Gauss-Newton* method.

The gradient method is another simple approximation of the solution of the equation 2.13 where the Hessian is taken to be equivalent to the indented matrix I, and the gradient is multiplied by a scalar α, which is determined, in general, by a line search (Pratt et al., 1998) to make the convergence faster. The gradient method can be written as:

$$\Delta \mathbf{m} = -\alpha \Re \left[\mathbf{J}^{\dagger} \Delta \mathbf{d} \right] \tag{2.15}$$

In the next section, the Gauss-Newton solution is used to demonstrate the deconvolution imaging condition in the migration (gradient) as another type of Hessian approximation.

2.1.3 Deconvolution imaging condition

At this point, let me first introduce the forward problem (namely, modeling the full seismic wavefield). The wave equation in time domain is:

$$\mathbf{M}(\mathbf{x}) \frac{d^2 \mathbf{u}(\mathbf{x}, t)}{dt^2} = \mathbf{A}(\mathbf{x}) \mathbf{u}(\mathbf{x}, t) + \mathbf{s}(\mathbf{x}, t) \tag{2.16}$$

where $\boldsymbol{M}$ and $\boldsymbol{A}$ are the mass and stiffness matrices, respectively (Marfurt, 1984), $\boldsymbol{s}$ is the source, $\boldsymbol{u}$ is the wavefield, t is the time and $\boldsymbol{x}$ represents the spatial coordinates. In frequency domain (taking the Fourier transform of equation 2.16), the wave equation reduces is reduced to a system of linear equations, with the source in the right-hand side and the wavefield as the solution (Virieux and Operto, 2009). Equation 2.16 can be compacted to:

$$\mathbf{B}(\mathbf{x}, \omega) \mathbf{u}(\mathbf{x}, \omega) = \mathbf{s}(\mathbf{x}, \omega) \tag{2.17}$$

where $\boldsymbol{B}$ is the impedance matrix, or a forward-problem operator (Marfurt, 1984).

Now, let me go back to the Gauss-Newton method and introduce an approximation of the inverse Hessian in terms of a deconvolution imaging condition. Usually a correlation imaging condition is used during migration due to its stability, but it lacks true amplitude. Margrave et al. (2010) use a deconvolution imaging condition in the migration of the residuals and Margrave et al. (2011) develop a numerical assumption. Let's rewrite the equation 2.14 as:

$$\begin{aligned}\Delta\mathbf{m} &= -\Re\left[\left(\mathbf{J}^{\dagger}\mathbf{W}_d\mathbf{J}\right)+\epsilon\mathbf{W}_m\right]^{-1}\Re\left[\mathbf{J}^{\dagger}\mathbf{W}_d\Delta\mathbf{d}\right] \\ &= -\underbrace{\Re\left[\left(\mathbf{J}^{\dagger}\mathbf{W}_d\mathbf{J}\right)+\epsilon\mathbf{W}_m\right]^{-1}}_{\text{Inverse of Hessian term}}\underbrace{\Re\left[\mathbf{J}^{T}\mathbf{W}_d\Delta\mathbf{d}^{*}\right]}_{\text{Gradient term}}\end{aligned} \tag{2.18}$$

where $\mathbf{W}_d = \mathbf{S}_d^t\mathbf{S}_d$ is a data-weighting matrix, $\mathbf{W}_m = \mathbf{S}_m^t\mathbf{S}_m$ is a regularization matrix and ϵ is a stabilization factor (to avoid zeros during computation). The operator $\mathbf{S}_d$ can be used as a diagonal weighting operator to control the weight of the elements of the misfit vector. The operator $\mathbf{S}_m$ is, in general, used as a roughness operator, to penalize the roughness of the model $\mathbf{m}$. The symbol * is the complex conjugate.

The Jacobian can be explicitly written as:

$$\mathbf{J} = \frac{\partial\mathbf{u}}{\partial\mathbf{p}} = \mathbf{B}^{-1}\frac{\partial\mathbf{B}}{\partial\mathbf{p}}\mathbf{u} \tag{2.19}$$

where $\mathbf{B}$ is the forward modeling operator, also known as the impedance matrix (Marfurt, 1984), $\mathbf{B}^{-1}$ is the Green's operator and $\mathbf{u}$ is the wavefield.

Now, replacing the Jacobian of equation 2.19 in gradient term of equation 2.18, expanding and letting $\mathbf{W}_d = \mathbf{I}$, the gradient is:

$$\nabla C_{\mathbf{m}} = \Re\left[\mathbf{u}^T\frac{\partial\mathbf{B}}{\partial\mathbf{p}}^T \underbrace{\mathbf{B}^{-1^T}\Delta\mathbf{d}^*}_{\substack{\text{back-propagated time}\\\text{reversed residual}}}\right] \tag{2.20}$$

The gradient of the equation 2.20 is a reverse time migration with a cross-correlation imaging condition (Tarantola, 1984).

Doing the same in the Hessian term, I have:

$$\mathbf{H} = \Re\left[\mathbf{u}^\dagger \overbrace{\left(\frac{\partial \mathbf{B}}{\partial \mathbf{p}}\right)^\dagger \underbrace{\mathbf{B}^{-1^\dagger}\mathbf{B}^{-1}}_{\text{geometrical spreading}} \frac{\partial \mathbf{B}}{\partial \mathbf{p}}}^{\text{scatterer weighting}} \mathbf{u}\right] \tag{2.21}$$

The term $(\partial \mathbf{B}/\partial \mathbf{p})^T$ is equals to ω^2 and $(\mathbf{B}^{-1})^\dagger(\mathbf{B}^{-1})$ is, for a homogeneous medium, equals to r^{-2} (Margrave et al., 2011). Substituting equations 2.20 and 2.21 into equation 2.18, I get:

$$\Delta\mathbf{m} = \left(\frac{r}{\omega}\right)^2 \underbrace{\frac{\Re\left(\mathbf{u}^T\mathbf{BPTR}\right)}{\Re\left(\mathbf{u}^\dagger\mathbf{u}\right)}}_{\substack{\text{deconvolution} \\ \text{imaging condition}}} \tag{2.22}$$

where the **BPTR** is the backpropagated time-reversed data residual $\mathbf{B}^{-1^T}\Delta\mathbf{d}^*$ (in other words, a reverse-time migration), which is divided by the autocorrelation of the downward propagated field. This is equivalent to a deconvolution imaging condition (Claerbout, 1971).

Equation 2.22 shows that the model update can basically be computed by a reverse-time migration of the residuals with a deconvolution imaging condition. Margrave et al. (2010) used a WEM (wave-equation migration), combined with well information to do a velocity inversion. Our goal is to follow the same assumption as Margrave et al. (2010), interpreting the FWI routine as combination of seismic processing tools and using a PSPI (phase-shift-plus-interpolation) migration to compute the gradient.

2.1.4 Trial and error line search to estimate the step length

Gradient is a result of the migrated residuals (which are stacked for all shot positions) and it is a reflection coefficients model. Coming back to equation 2.15, the gradient requires to be multiplied by a scalar α before updating the current model. This scalar determines the *step length* (Lines and Treitel, 1984). On the steepest descent method, the step length can be understood in two different ways: 1) as the gradient is a matrix of small numbers (reflection coefficients), the step length can be understood as a scale factor that optimizes the update and, 2) it can be interpreted as an impedance inversion operator (if an impedance inversion is not applied on the gradient). I am going to refer the step length as a scale factor.

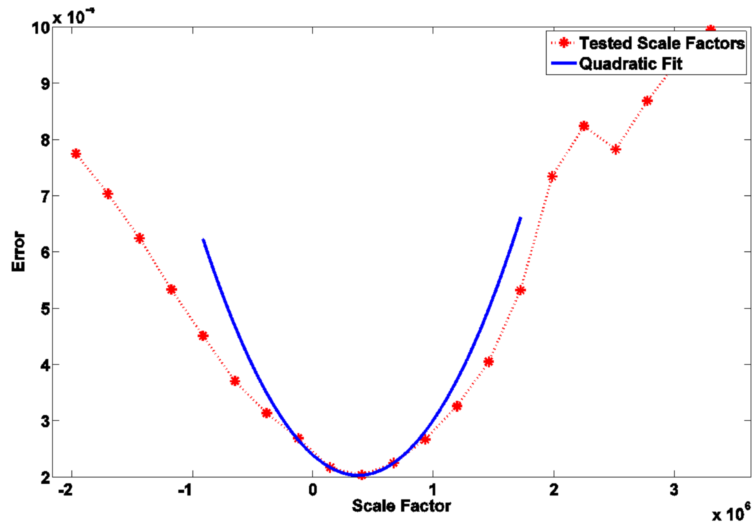


Figure 2.1: The trial and error line search. 21 tries (red dots) are performed around a minimum and maximum and a quadratic interpolation (blue line), using the 3 tries with the smallest error, contains the optimized scale factor for the model update.

On this initial test of the FWI method, the scale factor (step length) needs to be selected in a way to minimize the objective function (equation 2.1). This can be accomplished by a line search. I use 21 different values of step lengths (by find the maximum and minimum difference of the gradient and current model, and using 21 values equally distant inside the interval), update the model with each one of them, generate a synthetic shot for each one, and compute the error (objective function) for each one of the models (the red dots on figure 2.1), so I can draw the objective function to analyze its behavior and be able to optimize it. Later, the 3 points with the smallest errors are used to interpolate a quadratic curve (the blue line on figure 2.1). The optimized scale factor is the minimum value of the quadratic fit.

2.1.5 Migration cost analysis

One of the main proposals of this work is to replace the pre-stack depth migration from RTM to PSPI. The main reason is to reduce cost, but preserving its resolution the most. Gray (1999) compares the RTM, PSPI and Kirchhoff migrations by their computational cost in terms of Fourier transforms and grid points for each depth step.

Let N be the total number of grid points of a seismic data ($N = n \times m$ of a matrix with size $n \times m$). The Kirchhoff migration is a ray-tracing based migration and its costs is estimated (Gray, 1999) as approximately in the order of $O(N)$, mainly dominated by the cost of the interpolation (summation over the number of the traces in the aperture). Kirchhoff migration can handle step dips, but lacks in retrieve an accurate amplitude in the process and has limitations for lateral variations of the velocity model.

The total cost of a Fourier transform is $Nlog(N)$. The PSPI migration transforms the data to the frequency domain by a Fourier transform and apply a downward continuation of a depth step Δz for N_v reference velocities. The cost for that is, for all the reference velocities, of $O(N_v N log(N))$. The PSPI migration provides a more accurate amplitude and handles lateral velocity variation, but can not image steep dips with high accuracy.

The reverse-time-migration (RTM) does not have any of the limitations listed above, but it comes with a high computational cost ($O(N^2)$). In table 2.1 I summarize the advantages and disadvantages of the three times of migration discussed.

Table 2.1: Comparing the advantages and disadvantages of the Kirchhoff, PSPI and RTM migrations.) are simulations of well positions.

	Kirchhoff	**PSPI**	**RTM**
Lateral velocity variation	No	Yes	Yes
Can image steep ($> 70^o$) dips	Yes	No	Yes
Accurate amplitude	No	Yes	Yes
Computational cost	$O(N)$	$O(N_v N log(N))$	$O(N^2)$

In my work, I selected a migration algorithm that provides a high resolution image with lower costs. By selecting the PSPI to be my standard migration operator, I have a balanced algorithm in the cost-benefit trade-off. However, I believe that FWI can be executed with any pre-stack depth migration. For more simple areas, maybe the Kirchhoff migration is enough for the inversion. On areas with presence of salt bodies, the RTM (back to the standard FWI) would be the perfect choice.

2.2 Simulation

2.2.1 Mute selection

The FWI using a PSPI migration with a deconvolution imaging condition is tested on synthetic data only. A velocity model (figure 2.2) is used for the forward modeling, simulating a 2D seismic marine acquisition, and has a resolution of 10x10 meters. The red line is the

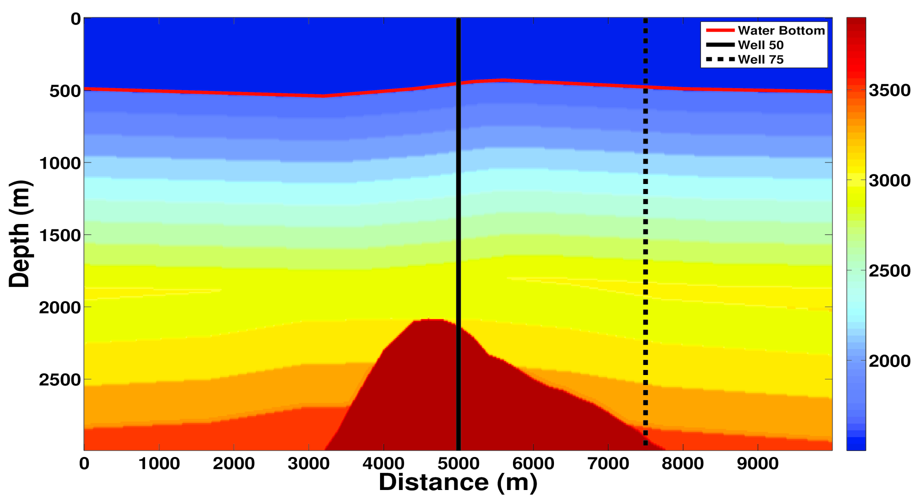


Figure 2.2: Velocity model used to create the shots for the FWI test. The color bar represents the wave propagation velocity in m/s. The red line is the water bottom picking and the vertical black lines (solid and dashed) are simulations of well positions.

water bottom picking and the vertical black lines (solid and dashed) are simulations of well positions. The water bottom picking is used as a mute edge (everything above the picking is muted, even for initial model as for each updated iteration) and the well positions are used in the next session to create initial velocity models.

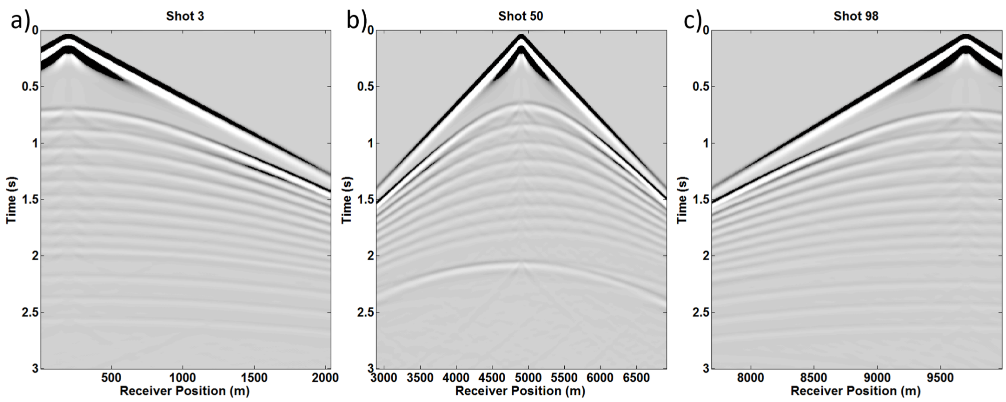


Figure 2.3: Shots created using the velocity model of the figure 2.2. a) Shot 3, b) shot 50 and c) shot 98.

Synthetic shots are generated to be used as the ”recorded” data. To simulate a seismic acquisition, 100 shots were created using the velocity model of the figure 2.2 with an acoustic finite difference forward modeling algorithm with a second order Laplacian approximation (Youzwishen and Gary, 1999). Shots and receivers are at the surface ($z = 0$). Some examples are found on figure 2.3.

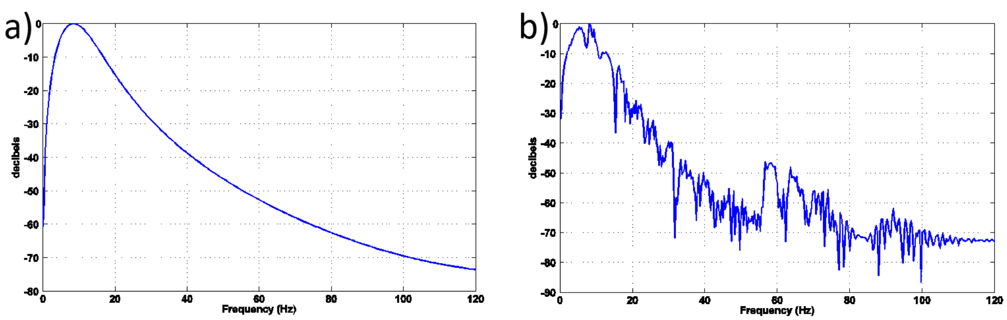


Figure 2.4: Amplitude spectrum for the a) wavelet and b) shot 50.

The shots have a split-spread receiver pattern and they may have different numbers of receivers that go from 201 in the borders to 401 in the center, with 3s of record time. The shot spacing is 100m and the receiver spacing is 10m (maximum offsets of -2000m to 2000m). However, for the FWI code, only 96 shots were used (the same number of nodes in the supercomputer available, for parallel processing). The dominant frequency of the source wavelet used is 10Hz, but the dominant frequency of the recorded data is about 8Hz (probably due some relationship between wavelength and layers thickness), as shown on figure 2.4.

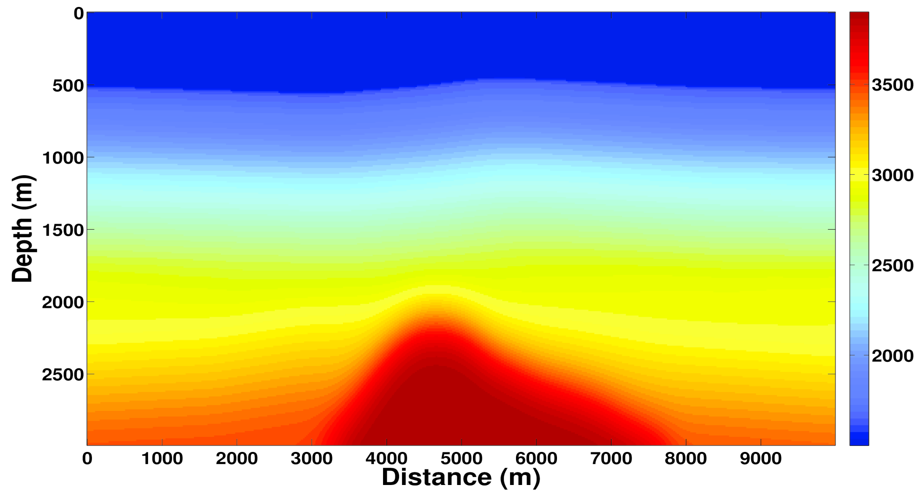


Figure 2.5: The initial model for the test is a smoothed version of the correct model (figure 2.2).

The initial velocity model (figure 2.5) for this test is a smoothed version of the real velocity model (figure 2.2). The Gaussian window for smoothing is calculated as $\Lambda_{nom} = f_{dom}/V_{mean}$, where V_{mean} is the mean velocity of the real velocity model, equals to 2539 m/s. This equation is the same as used by Margrave et al. (2010). A mute is applied above the water bottom (red line) to eliminate any artifact above it. The same water bottom mute is applied at each iteration in the stacked migrated residuals. It is important to remember that this kind of velocity model is very hard (if not impossible) to be obtained in a real case. But, for now, I just want to analyze if my algorithm is stable and converges.

New shots are generated with the velocity model of the figure 2.5, as shown in figure 2.6: a) is the real shot, b) is the synthetic data and c) are the residuals. It is important to remember that the same wavelet of the input shots is used to create the synthetic shots at each iteration. This is the reason the direct wave is perfectly removed in the residuals. I also note a small WB in the synthetic shot, due to the WB mute applied in the initial model.

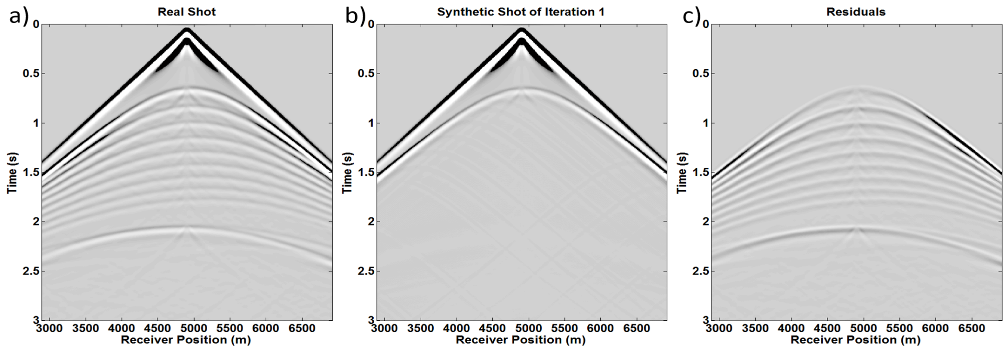


Figure 2.6: Shot number 50 in the first iteration (synthetic shots created using initial velocity model of figure 2.5a), where a) is the real shot, b) is the synthetic data and c) are the residuals.

The residuals are migrated using the PSPI migration (Gazdag and Sguazzero, 1984; Ferguson and Margrave, 1996, 2005), using selected frequency bands, same procedure as Margrave et al. (2010). However, different frequency bands are used in this work. It is usual to start the inversion with low-frequency content (Pratt et al., 1998; Plessix et al., 2010), to avoid a local minimum, and then invert higher frequencies. In the absence of low

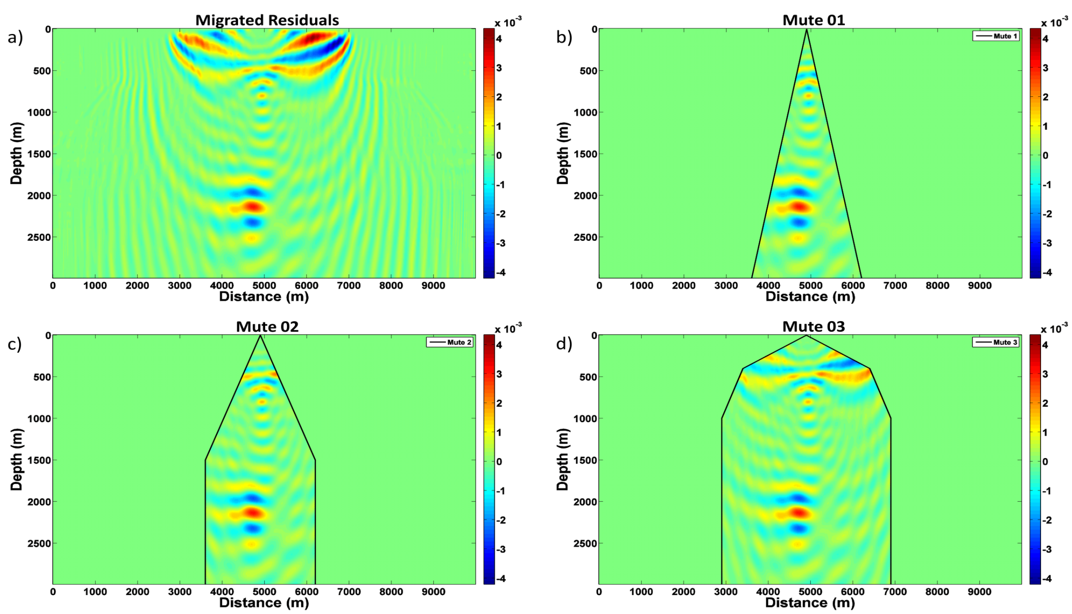


Figure 2.7: Differents mutes applied in the migrated residuals: a) no mute, b) narrow mute, c) normal mute and d) wide mute.

frequencies in the data, a deconvolution can be applied to recover and invert the pseudo-low frequencies (Fei et al., 2012), but the success of the inversion is questionable. Warner et al. (2013) invert only the frequencies with a high signal to noise ratio, starting with the lowest frequency possible. In this work, I am using frequencies where the amplitude spectrum is above -10dB. Looking back at figure 2.4 b, I decided to use frequencies from 6 to 15Hz. The PSPI migration algorithm allows us to choose the frequency bands to migrate. Here I am using the frequencies above where the amplitude spectrum is -10dB as the maximum frequency of the band, and the minimum will always be 1Hz. This way, the first iteration has the residuals migrated in the frequency band of 1 to 6Hz (as the frequencies above -10dB goes from 6Hz to 15Hz). Pratt et al. (1998) invert for a chosen frequency until stability of the objective function is reached and then vary the frequency content. 96 residuals (acquired minus synthetic shots) are migrated and then stacked to create the model update. A line search, with 21 guesses, is used to calculate the step length, and it is controlled by the shot in the central position in the model. Plessix et al. (2010) repeat same frequency band 20 times before change to next frequency band. Here, I repeat the same frequency band 10 times before increasing the maximum frequency by 1Hz. In other words, I repeat the inversion in the frequency band of 1-6Hz 10 times, then 1-7Hz 10 times and keep this process until reach the maximum frequency of 15Hz, or until I decide to stop the process.

Margrave et al. (2010) suggest the use of a mute in the migrated residuals to avoid artifacts. Figure 2.7 shows 4 different mute tests applied in the migrated residuals during the process: a) no mute, b) narrow mute, c) normal mute and d) wide mute, for the data inverted using the initial velocity model of the figure 2.5.

The absence of a mute drives the inversion to a final model with great resolution (figure 2.8), but allows the appearance of artifacts, such as the almost vertical object in the central area of figures 2.8b, c and d, leading the line search to stop updating the model by selecting a step length close to zero. These artifacts may be the sum of some far offset effects in the migrated residuals and a mute looks to be necessary. It is possible to observe some shadows

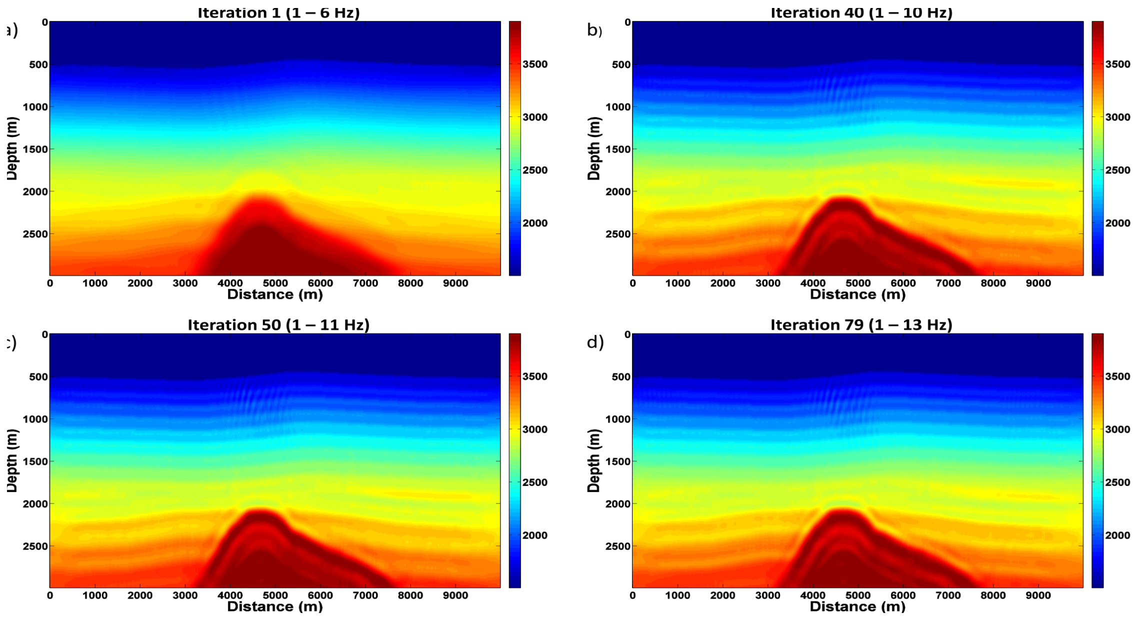


Figure 2.8: Inverted velocities with no mute applied in the migrated residuals at a) iteration 1, b) iteration 40, c) iteration 50 and d) iteration 79. Artifacts are observed in the data.

around and inside the high velocity body. They are larger for the first iterations, but start to decrease as the migrated frequencies increase.

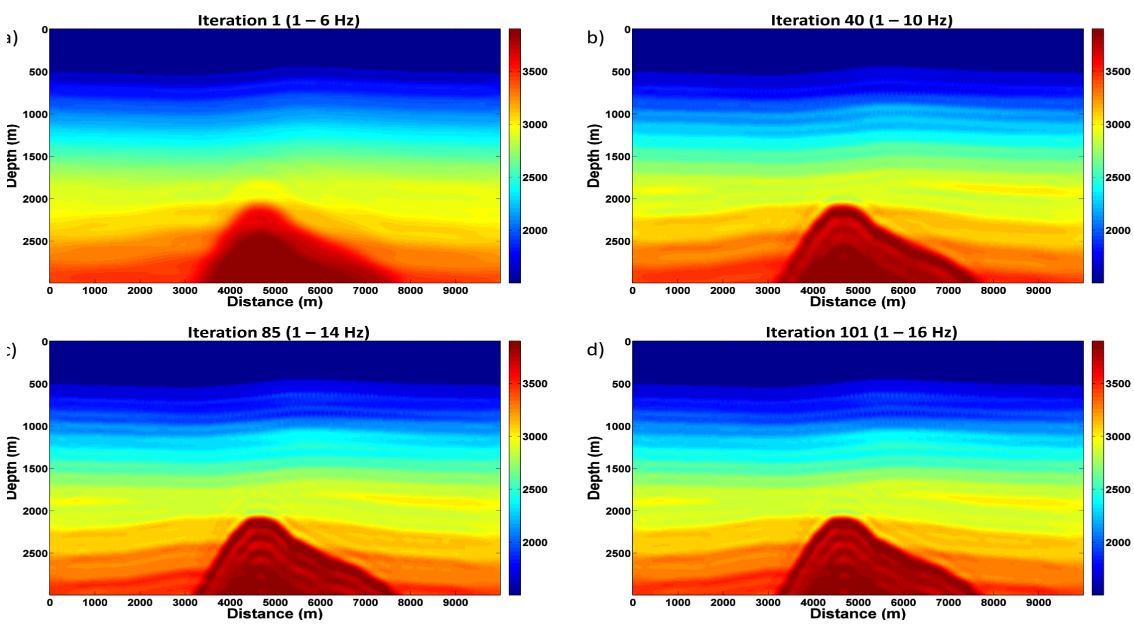


Figure 2.9: Inverted velocities with the narrow mute (figure 2.7b) applied in the migrated residuals at a) iteration 1, b) iteration 40, c) iteration 85 and d) iteration 101.

A first test of a mute is the narrow and very aggressive mute of the figure 2.7b, allowing only a small amount of data to pass the selection. The results are presented in figure 2.9. Artifacts due to far offset effects are not observed anymore. However, the mute is too aggressive at lower depths, and the borders of the mute windows are present in the velocity model (diagonal events in the shallow area of the whole model). I also see the presence of the shadows close to the high velocity body, same as seen in figure 2.8. Even so, the result is encouraging.

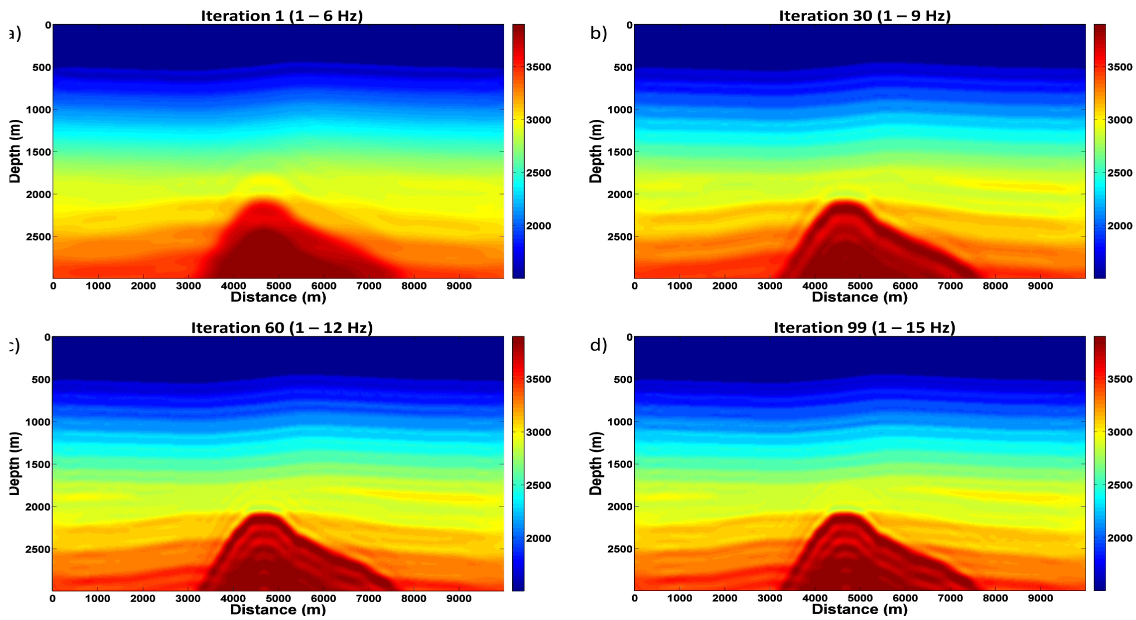


Figure 2.10: Inverted velocities with the normal mute (figure 2.7c) applied in the migrated residuals at a) iteration 1, b) iteration 30, c) iteration 60 and d) iteration 99.

Apparently, the mute needs to be chosen carefully. If too aggressive, a mute can remove important signal from the migrated residuals, and a too generous one can allow the appearance of artifacts. Let me analyze the other muting options.

Figure 2.10 shows the results of a less aggressive mute of the figure 2.7c. No artifacts are observed and, differently of the narrow mute results, the lower depth parts of the inverted velocities show a better resolution. The shadows around high velocity bodies are still present. It is possible to say that I had a great improvement by choosing a not so aggressive mute.

This bring us one important question: can I be even more generous with the mute and allow more information to be integrated to the model without the addition of artifacts?

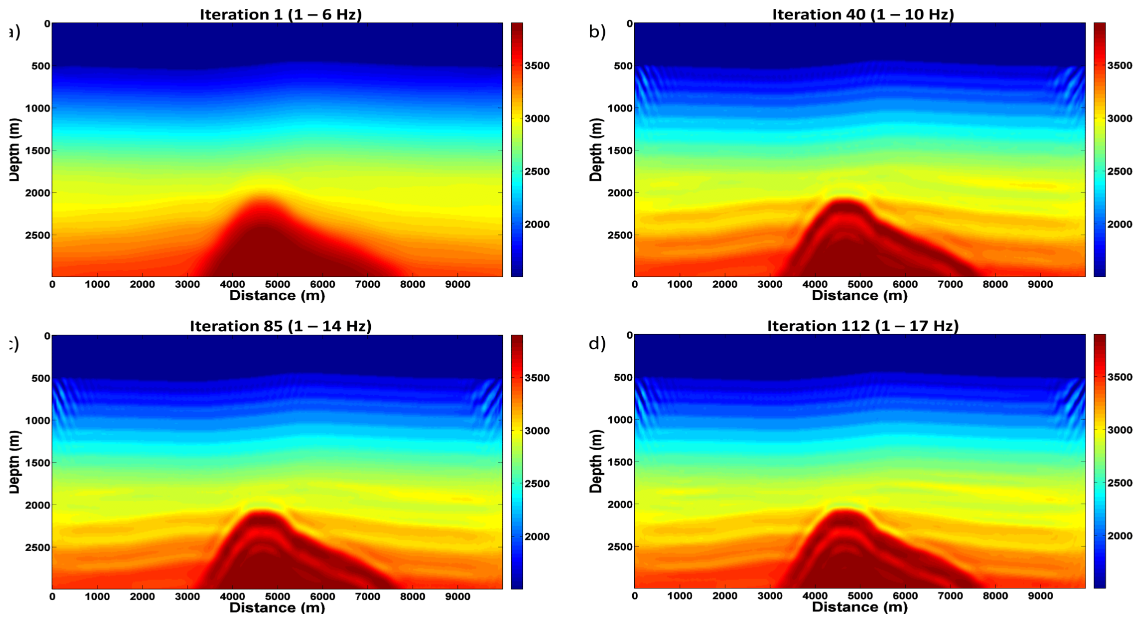


Figure 2.11: Inverted velocities with the wide mute (figure 2.7d) applied in the migrated residuals at a) iteration 1, b) iteration 40, c) iteration 85 and d) iteration 112.

Now, more data is allowed to pass with a conservative mute (figure 2.7d). I am trying to have a more accurate model update without adding artifacts. However, the inverted models of figure 2.11 show that this goal is not reached. Strong artifacts appeared in the edges and center portions of the inverted velocity model, similar of the artifacts in the figure 2.8, suggesting that the mute cannot be too generous, or artifacts can be "included" in the inverted model. Curiously, a wide mute allowed the appearance of more artifacts than using no mute at all. Apparently, the far offset events cancel each other in the borders of the model if it not removed, but these same events are still evident in the central area.

Figure 2.12a compares the shots errors (the average residuals of all the synthetic shots relative to the real shots, per iteration. In other words, the objective function per iteration) for all the mute tests done, showing a best behavior for the inversion with normal mute (blue line). As this is a synthetic test, it is possible to compare each inverted velocity model

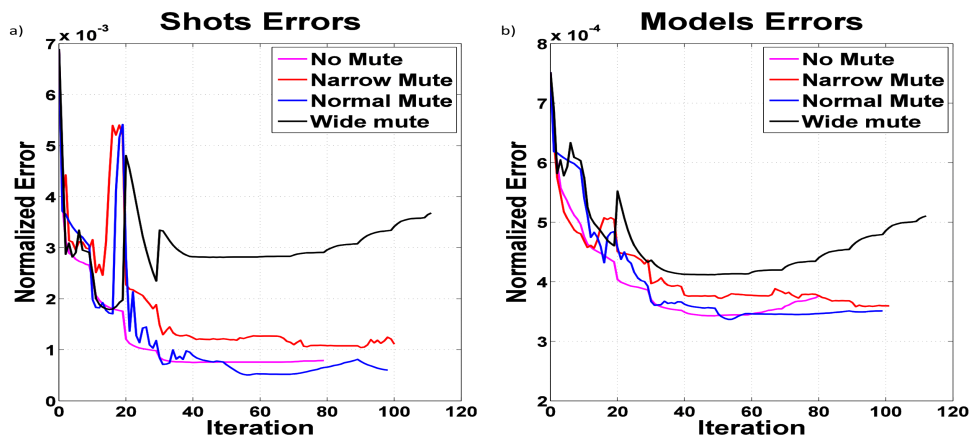


Figure 2.12: a) objective function and b) model deviation. In both plots, the normal mute (blue line) shows the best behavior.

per iteration to the real velocity model (model deviation), which is plotted in figure 2.12b. Again, the normal mute shows a better behavior, with the lowest errors. The worst behavior happens with the wide mute, due to the strong artifacts in the borders. The test with no mute applied has a better behavior than the narrow mute test, even with the artifacts that appeared in the middle of the model. The low depth stripes in the narrow mute test must have a high weight in the errors calculations.

The narrow and normal mute tests do not appear to have the errors stabilized (figures 2.12 a and b), suggesting that I must keep with the iterations at higher frequency. In the next session, it is done for the normal mute, until the algorithm reaches 40Hz as the maximum inverted frequency.

Comparing all the resulted velocity models with the initial (dashed line) and real (black line) velocity models in figure 2.13, the normal mute test (red line) placed the reflectors (layers interfaces) more precisely than the others ones. For this 1D analyzes, no mute (green line) and the wide (magenta line) look to provide a reasonable inversion, in contrary of the conclusion by checking figures 2.8 and 2.11, respectively, where artifacts are present in the model. The narrow mute (blue line) do not match the reflectors properly, as the wide mute shows higher effect of the shadows around the high velocity body (around 2000m).

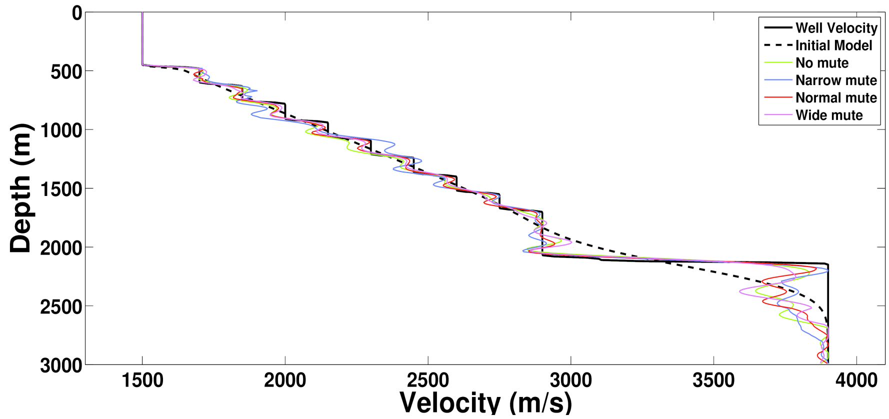


Figure 2.13: Comparing the velocities models in the center position of the real model (black line), initial model (dashed line) and the last iteration of all the tests. The normal mute test (red line) placed the reflectors (layers interfaces) in a more precise position than the others tests.

The results so far show that a not too aggressive and not too conservative mute is the best choice, and showed that I must keep inverting higher frequencies. However, the initial velocity model is a "good" start, as it is a smoothed version of the real velocity model. In the next section, the smoothed and other initial velocity models are tested.

2.2.2 The initial velocity model

Warner et al. (2013) use a PSDM velocity as starting model. The reason is to start the process as close as possible to the global minimum, avoiding the inversion to be "trapped" in a local minimum. I am testing this effect using different initial velocity models. All the tests here are done using the normal mute (figure 2.7c). For now on, I am inverting higher frequencies, going up to 40Hz. However, for frequencies higher than 15Hz the frequency range is repeat 5 times instead of 10, and, for frequencies higher than 25Hz, the maximum frequency of the range is increased by 5Hz instead of 1Hz. This choice is due to the low convergence rate.

Different initial velocity models were tested. One is simply the same as used for the mute test, but using only the normal mute and continuing the iterations for higher frequencies. Others two are initial velocity model based in well simulations, drawn in figure 2.2. One is

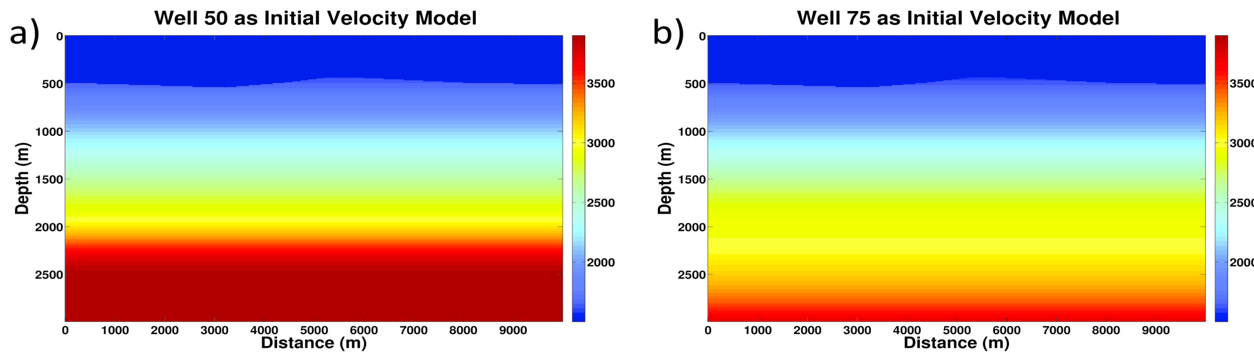


Figure 2.14: Initial velocity models using as base a) well 50 and b) well 75 in the right (as draw in figure 2.2).

generated from the "Well 50", and the other is generated from the "Well 75". In both cases, the initial velocity model consist on flat layers, which were smoothed and plotted in figure 2.14.

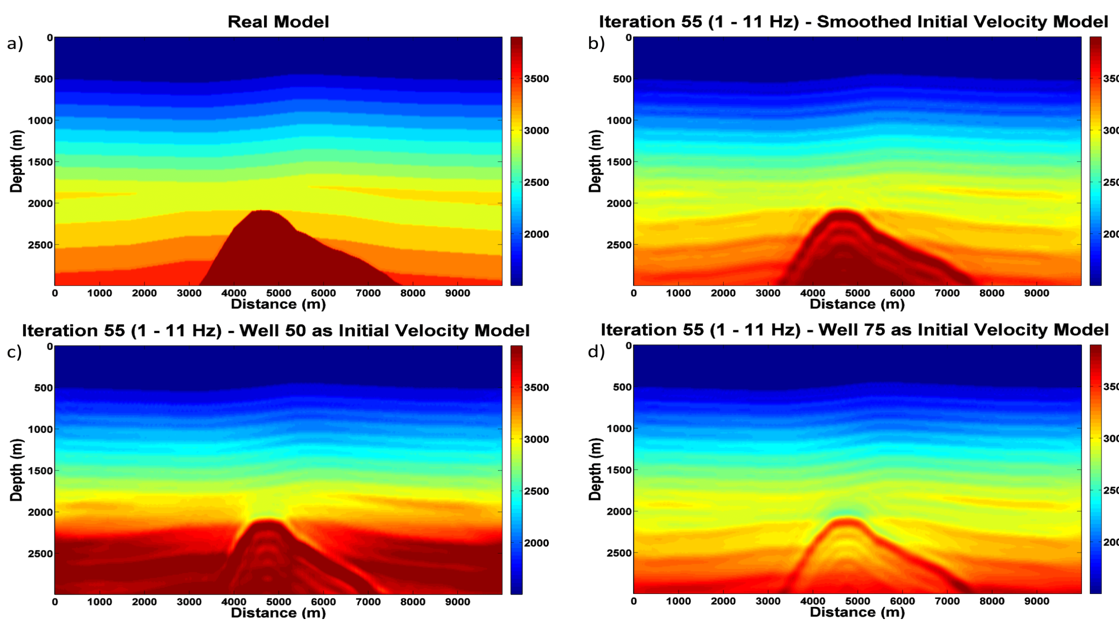


Figure 2.15: Inverted velocities with different initial velocity models, where a) is the real model, b) is the iteration 55 of the inversion using the smoothed initial velocity model (figure 2.5), c) is the iteration 55 of the inversion using the well 50 as the initial velocity model (figure 2.14a) and d) is the iteration 55 of the inversion using the well 75 as the initial velocity model (figure 2.14b).

Inversion results are shown on figure 2.15 (the iteration 55 for all the cases), and in figure

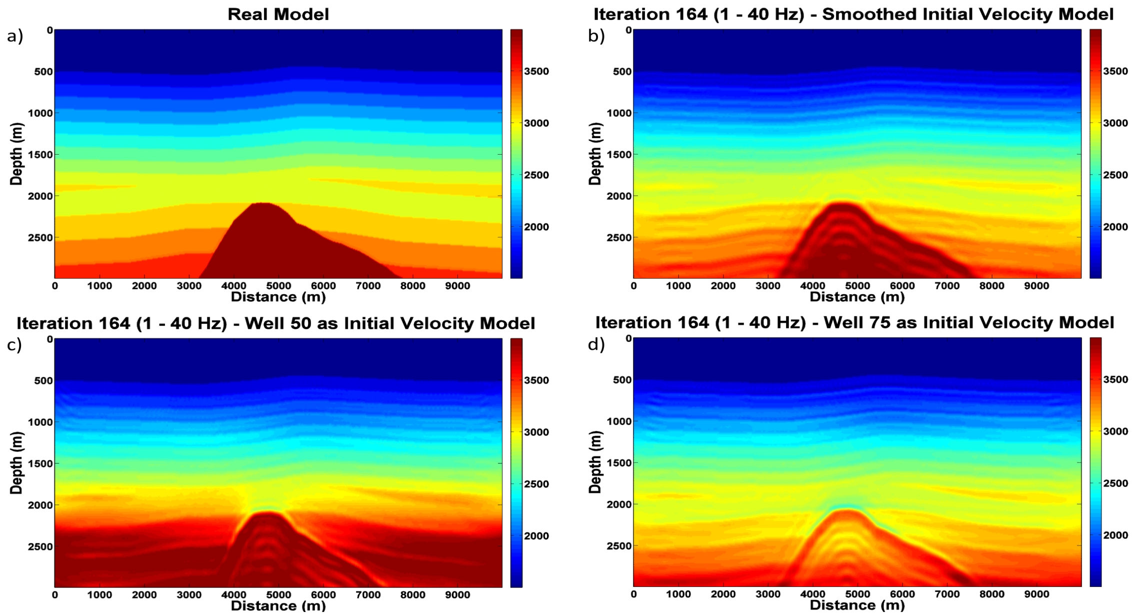


Figure 2.16: Same as figure 2.15, but for the iteration 164. The inversion started to diverge after ~60 iterations for almost all the cases. For the inversion using Well 75 (c), the inversion is still converging.

2.16 (final model). Inversion is more successful when using a smoothed initial velocity model (figures 2.15b and 2.16b). I also have a reasonable result when using the initial velocity model based on Well 75 (figures 2.15d and 2.16d), where the shallow and deep layers have a good resolution, as the high velocity body edges, but not correctly "filled in". Even the sharp layers around 2000m have a better resolution. The results when using the initial velocity of figure 2.14a, showed on figures 2.15c) and 2.16c, biased at higher depths, where the interface of the layers and the high velocity body edges were found, but the correct velocities not. For the shallow part, the resolution looks better than when the smoothed initial velocity model is used.

Note that the inversions of figure 2.16 start to show some high sharp layers that do not exist (for example, around 2500m at figures 2.16b and 2.16d), due to high frequency noise. I also observed that, in both tests using the simulated wells, artifacts started to appear in the borders of the models. This might be due to line search be controlled by the shot in the middle of the model (step length is optimized only for the selected shot). This suggests that the line search must be improved, using more shots on different positions.

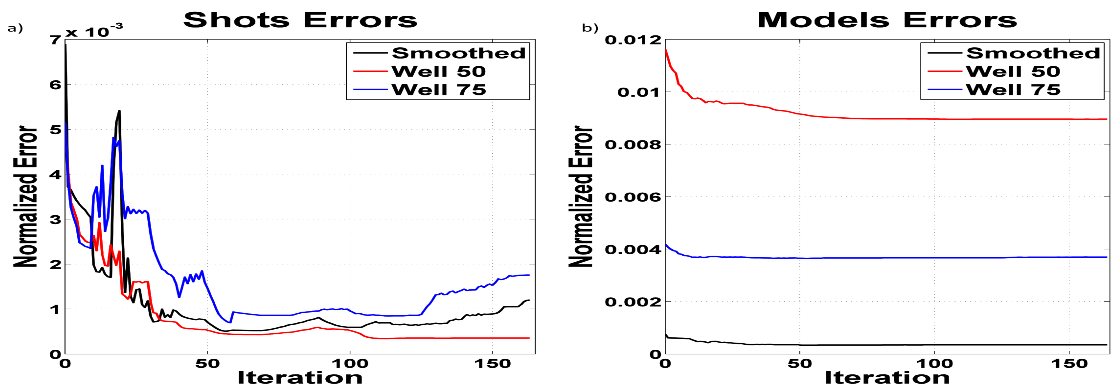


Figure 2.17: a) objective function and b) model deviation, using different initial velocity models. For the shots errors, the inversion using the well 50 as initial velocity model looks more stable. However, comparing the models errors, the best result is the one using a smoothed velocity model as the initial model.

When I compare the shots and models errors per iteration (figure 2.17), I have a very interesting observation. If we look to the models errors (figure 2.17b), we have the same conclusion as looking the figures 2.15 and 2.16: the best inversion comes when using a smoothed initial velocity model (black line at figure 2.17b), and the worst result comes when using the Well 50 as initial velocity model (red line). However, if we look at the shots' errors in figure 2.17a (the error that I can calculate on a real data inversion), I have a different result: the best shot errors comes when using the Well 50 as initial velocity model (red line). I believed it happened because the shot I selected to be the pilot one (to compute the step length and the objective function) has the same horizontal coordinate as the Well 50 (in the center of the model), making the initial model optimized in this shot position. I chose to compute the objective function (shot errors) only at the pilot position (shot in the center of the model). Maybe the previous observation would differ if I chose to compute the objectve function for all the shots and average. Interpreters and seismic processors can be biased to use the less optimized starting model if the choice is careless.

Looking at the 1D velocity comparison on figure 2.18, I observe that the inversion using the Well 75 as initial velocity model (red and blue dashed lines in figure 2.18b) is more

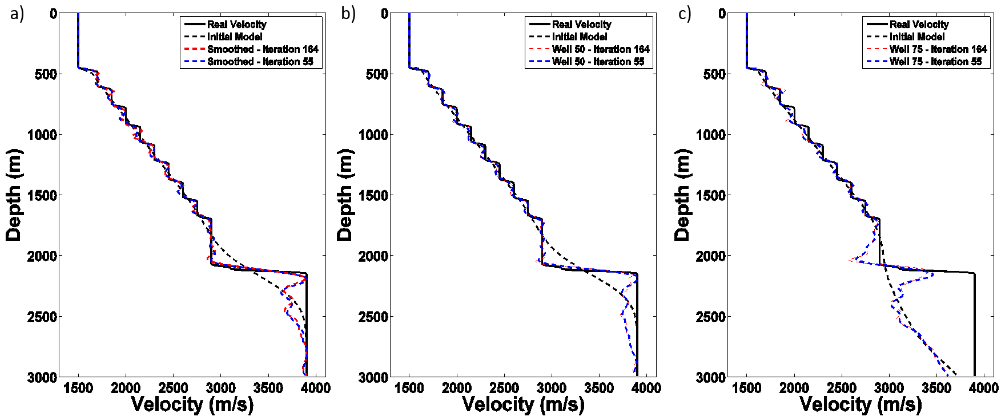


Figure 2.18: Comparing the velocities models of iteration 164 (red dashed line) and iteration 55 (blue dashed line) in the center position of the real model (black line) and the initial model (dashed line) with the a) last iteration of the smoothed initial model test, b) the well 50 as initial velocity model test and c) the well 75 as initial velocity model test.

accurate in the shallow part than the inversion when using the smoothed initial velocity model (red line in figure 2.18a).

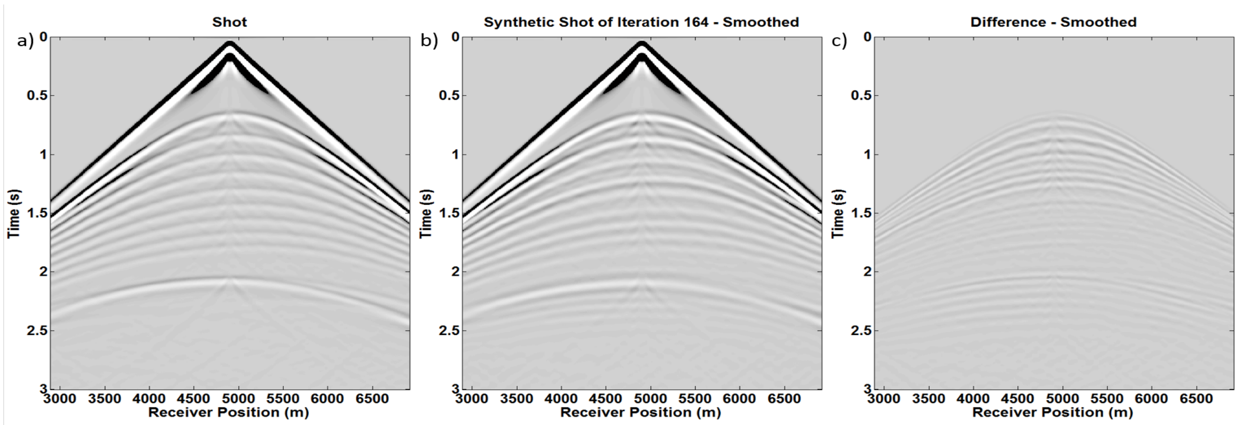


Figure 2.19: Comparing a) the real shot, b) synthetic shot of iteration 164 and c) the difference, with the smoothed initial velocity test.

Comparing the differences between real and synthetic shots of the iteration 164 of the inversion using smoothed initial velocity model (figure 2.19c) and inversion using the Well 50 as the initial velocity model (figure 2.20c), it is notable that the final model of the Well 50 inversion leads to synthetic data more similar to the real data than with the final model of

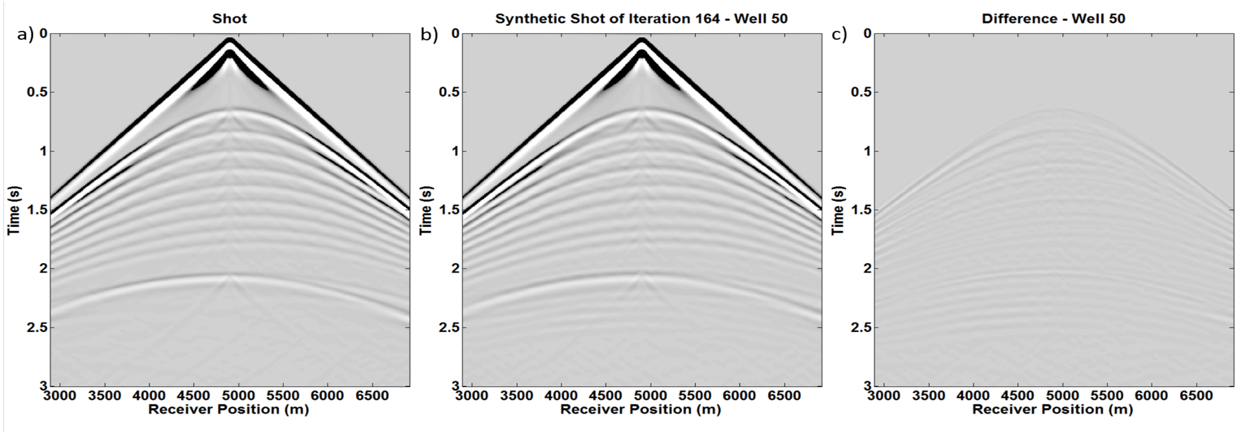


Figure 2.20: Comparing a) the real shot, b) synthetic shot of iteration 164 and c) the difference, with the well 50 as the initial velocity test.

the smoothed inversion. In others words, I found a bad solution that can generate synthetic shots closer to the real ones. Treitel (1989) has a similar experience and suggested that a good fit of the shots is necessary but it is not a sufficient condition for an accurate inversion. It can be avoided if the initial guess is not too far off the true model.

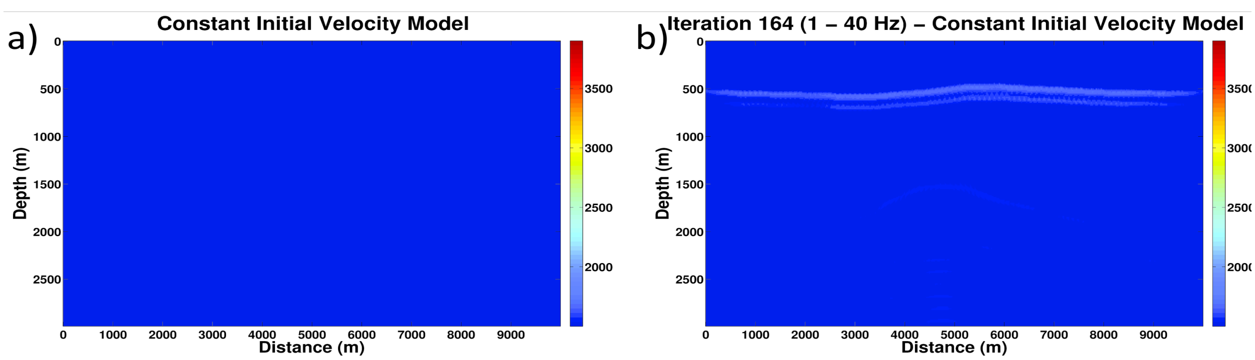


Figure 2.21: Results for the a) constant velocity initial model with b) 164 iterations. The inversion stopped in a local minimum.

Figure 2.21 shows the result when I apply the FWI in the dataset using an initial velocity model with a constant velocity of 1500m/s (water velocity). The inversion stopped in a local minimum after a few iterations.

I can say that the better the initial velocity model is, better will be the inversion. However, I had reasonable results for the wells tests. Maybe, if I have an initial model generated by an interpolation of two or more wells, the final model can have a comparable resolution to the one inverted from the smoothed version of the correct model.

2.2.3 First analysis using the Marmousi model

After good results using a simple velocity model, now I increased the challenge by applying the algorithm in the Marmousi model (figure 2.22). It has 3000m depth, 10420m width and 10m of resolution. The minimum wave propagation velocity in the model is 1500m/s and the maximum is 5500m/s

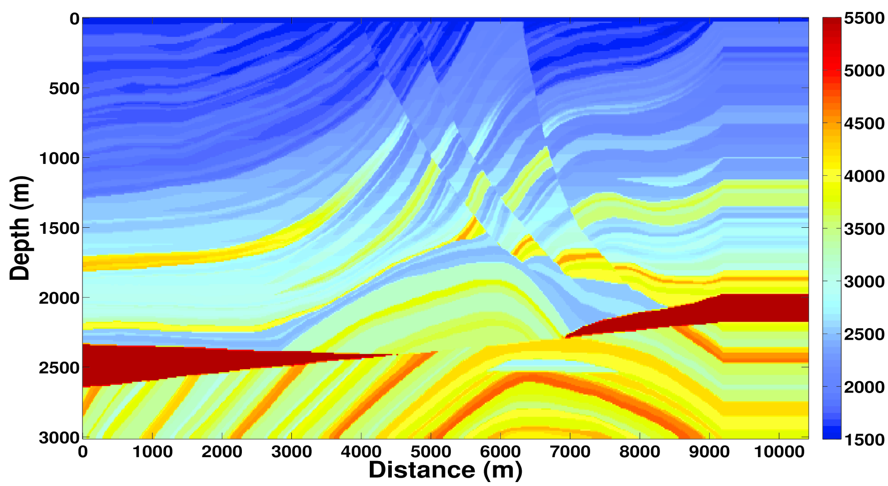


Figure 2.22: The Marmousi 2D model. The colors bar indicates the wave propagation speed in m/s.

102 shots were generated to simulate acquired data. Shot spacing is 100m, with a maximum of 401 receivers (varying in the borders), and 2000m of maximum offset. Register time is 3s, and sample rating of 4ms. Some examples of the shots shown on figure 2.23. The dominant frequency of the wavelet is 10Hz (figure 2.24a) which generated data with similar dominant frequency of 10Hz (figure 2.24b).

The starting model (figure 2.25) is a smoothed version of the model on figure 2.22 using a Gaussian window which width is calculated by $\lambda_{nom} = f_{dom}/V_{mean}$, where V_{mean} is the mean

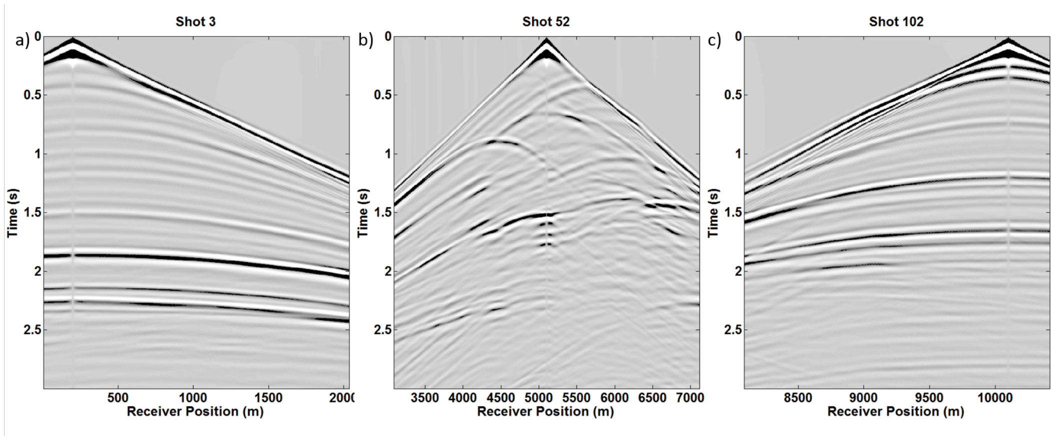


Figure 2.23: Three of the 102 shots used in the test. a) shot 3, b) shot 52 and c) shot 102.

velocity of the correct model.

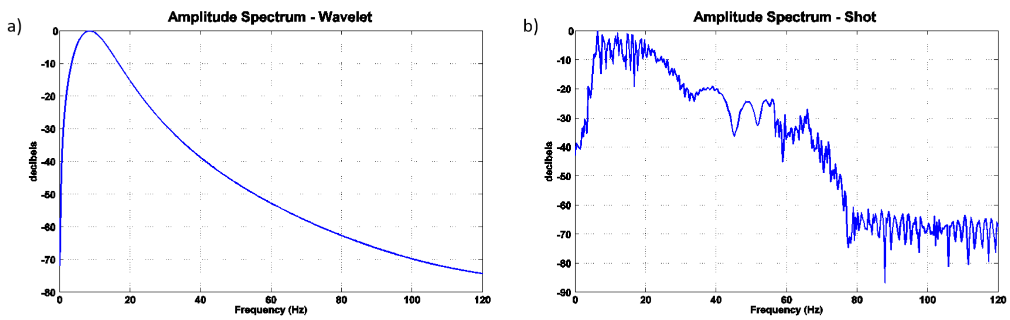


Figure 2.24: Amplitude spectrum of the A) wavelet and B) shot.

For this test, inversion starts with a frequency band of 1-4Hz, and the maximum frequency of the range is increase by one after 5 iterations (the second range is 1-5Hz), up to 13Hz, and then increased by 5Hz each two iterations. A normal mute similar to the mute of the figure 2.7c is applied at each migrated shot. The difference of the mute is that it starts at 300m in the surface. Stack is smoothed at each iteration by convolving it with a 2D Gaussian window (Margrave et al., 2010), whose width follow the relationship $h_w = v_{mean}/(6f_{max})$. The difference is that the v_{mean} is the mean velocity of the current model.

Inversion ended after 66 iterations, and results are shown at figure 2.26, where a) is the real model, b) is the initial model, c) is the model at iteration 18 and d) is the model at

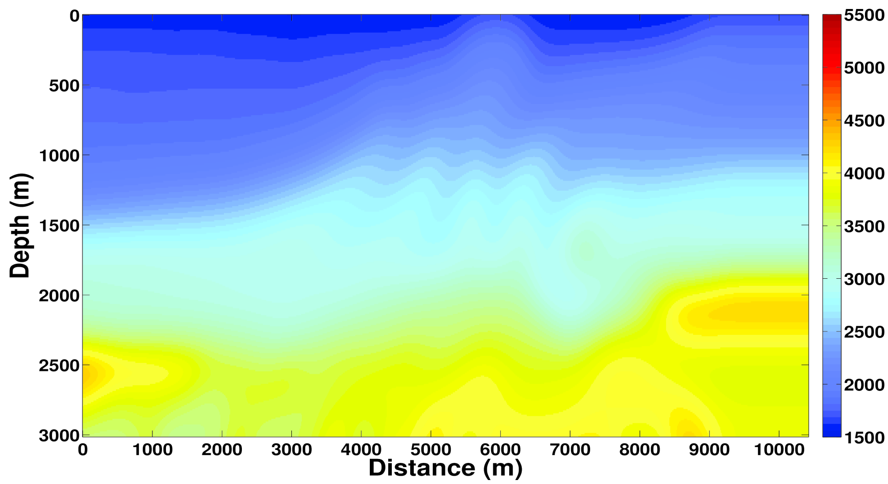


Figure 2.25: Initial velocity model is a convolution of the Marmousi model with a Gaussian window of 290m.

iteration 66. The inversion shows higher resolution after 66 iterations (figure 2.26d), as I migrate higher frequencies and with reasonable high resolution on shallow depth (2000m) but looks to lose resolution for deeper events if compared with the iteration 18 (figure 2.26c).

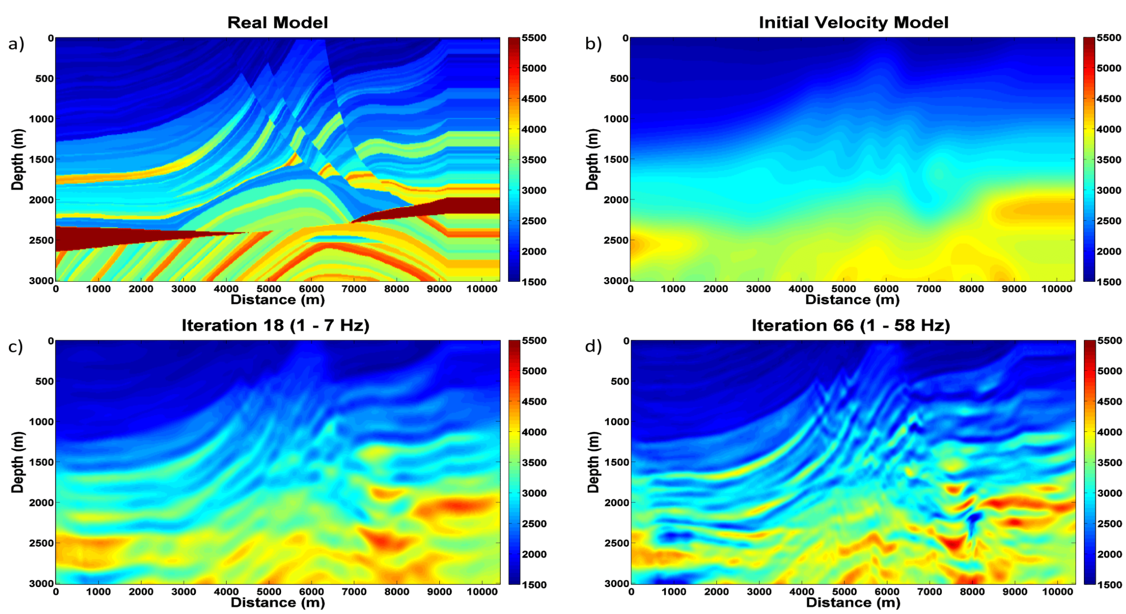


Figure 2.26: a) Marmousi model b) initial model, c) inversion at iteration 18 (lowest model error) d) iteration 66 (lowest shots errors).

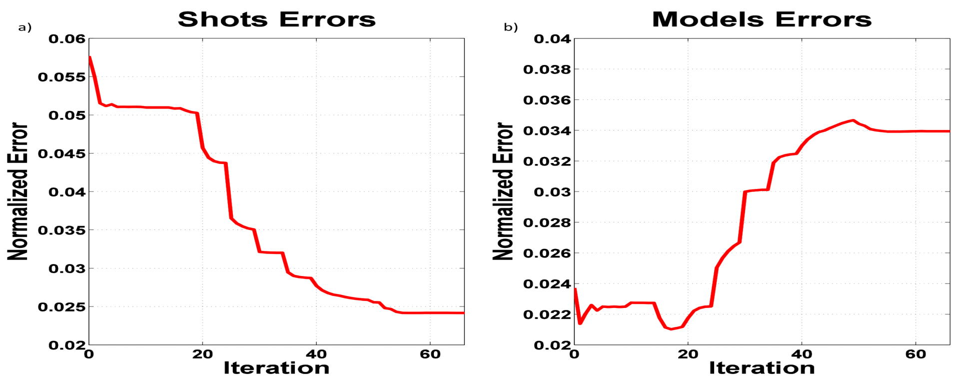


Figure 2.27: a) objective function and b) model deviation. The models look to be fitting better the data at each iteration but the precision of the models decrease starting from iteration 20.

Shots errors and model errors are shown at figure 2.27. Analyzing the shots errors (figure 2.27a), I note that models are updating in a way to minimize the difference of synthetic and real shots with the best fit at iteration 66. But the models errors plot (figure 2.27b), shows that the best model (lowest error) is the model of the iteration 18. I am getting a better fit with a model far from the global minimum. The fit is still not as good as I expect. Apparently, higher frequencies have a lower signal to noise ratio (noise was not added in the synthetic data. We may be observing the presence of numerical noise).

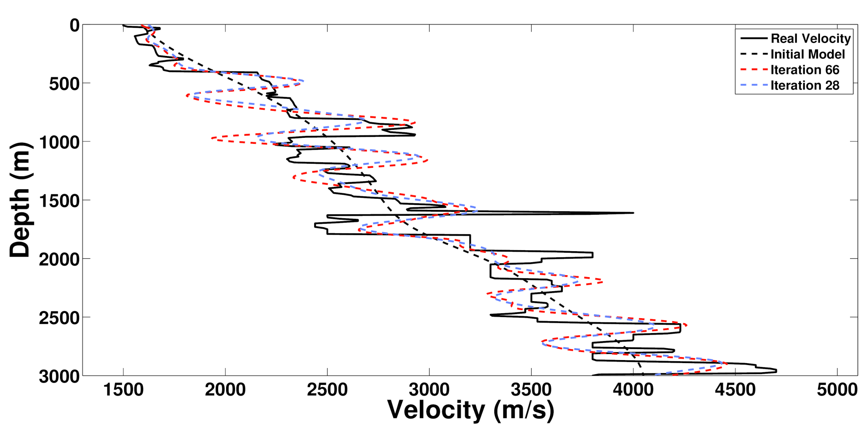


Figure 2.28: Comparing the velocities models in the center position of the real model (black line), initial model (dashed line) and the last iteration of all the tests. The normal mute test (red line) placed the reflectors (layers interfaces) in a more precise position than the others tests.

The 1D plot of figure 2.28 compares the inverted velocities (iteration 66 in red and iteration 18 in blue) with the real (black line) and initial (black dashed line) models in the middle of the survey. The FWI routine worked to find several reflectors positions but couldn't invert thinner layers. This must be caused by the absence of higher frequencies in the shots.

In the next chapter I will present a new gradient approximation, based on a deconvolution idea, where each frequency is migrated separately and then weighted averaged, using the step length as weight, and will be combined with a conjugate gradient algorithm.

2.3 Conclusions

In this chapter I presented the gradient descent method of the full waveform inversion applied for an acoustic inversion. I showed that the PSPI migration has less computational cost than the RTM, and that the PSPI migration can be used to compute the gradient, with some trade-off in cost and resolution. The step length is estimated by a line search and quadratic interpolation with high accuracy, but requires several synthetic shots. The migrated residuals need to be muted before stacking to prevent the appearance of artifacts in the updated velocity model, but can not also be too aggressive. I also tested the gradient descent FWI on a simple model with different initial models, whose were a lateral extrapolation of two sonic logs in the model. I observed that the accuracy of the inverted model depends on how close the initial model is to the true model. Finally, I tested the algorithm in the Marmousi model, showing a convergence of the objective function (difference between acquired and synthetic shots), but the inverted model was diverging, suggesting the minimization to a local minimum.

Chapter 3

A monochromatic averaged gradient approximation and impedance inversion

On the chapter 2 our first analysis of the FWI methodology is applied on a simple model. In this section, I am focusing on synthetic simulations in the Marmousi model. Here, I propose a new approximation for the gradient, based on monochromatic migrated shots, stacked, and then averaged. It is also shown a faster and cheaper method to compute the step length. As preliminary test, an impedance inversion is applied on the gradient, seeking improvements on the working flow, but it requires a better strategy. Impedance inversion of the gradient will be explored deeply on chapter 4.

3.1 Theory

3.1.1 Monochromatic averaged gradient

Minimizing the objective function of equation 2.4, one of the approximate solutions is the gradient (steepest descent) method:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n \mathbf{g}_n \tag{3.1}$$

where α is the step length (or scale factor), g is the gradient computed by back-propagating the data residual and n is the *n-th* iteration. Here, the gradient is understood to be the PSPI migrated residuals with a deconvolution imaging condition (Margrave et al., 2010, 2011; Pan et al., 2013; Guarido et al., 2014, 2015).

In this work I am proposing to estimate the gradient as the weighted average of each frequency content of the data migrated separately, using the step length for each frequency as weight. This means that for an iteration that I selected the frequency band from 1Hz to 5Hz, I migrate each frequency (1Hz, 2Hz, 3Hz, 4Hz and 5Hz) of each shot independently, generate a pseudo-gradient for each frequency, estimate one step length for each pseudo-gradient, and compute the weighted average of the pseudo-gradients using the step length as weight. This way, equation 3.1 can be written as:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \frac{1}{N}\sum_{i=1}^{N} \alpha_n(\omega_i)\mathbf{g}_n(\omega_i) \tag{3.2}$$

where ω_i is the *i-th* frequency in the total of N frequencies used to compute the gradient. The early iterations use very low frequencies only ($\approx 2 - 4Hz$) and next iterations have the lower frequency fixed and the maximum frequency increased, ending up with a larger frequency band. This means that, for each iteration, N migrations are computed. The step length α will be computed for each frequency prior the stack. This way I expect to flat the amplitude spectrum of the gradient and avoid the source effect.

3.1.2 Step length computation

As a cheaper way than a trial and error, Pica et al. (1990) computed the step length by minimizing the objective function:

$$C(\mathbf{m_{n+1}}) = C(\mathbf{m}_n + \alpha_n\mathbf{g}_n) = \left[\mathbf{d}_0 - \mathbf{d}\left(\mathbf{m}_n + \alpha_n\mathbf{g}_n\right)\right]^T \left[\mathbf{d}_0 - \mathbf{d}\left(\mathbf{m}_n + \alpha_n\mathbf{g}_n\right)\right] \tag{3.3}$$

I also have:

$$\mathbf{F}\delta\mathbf{m} = \lim_{\epsilon\to 0} \frac{\mathbf{d}(\mathbf{m} + \epsilon\delta\mathbf{m}) - \mathbf{d}(\mathbf{m})}{\epsilon} \tag{3.4}$$

where $\mathbf{F}$ is an operator that takes the derivative of $\mathbf{d}$ at the point $\mathbf{m}$, and $\delta\mathbf{m}$ is the perturbation in the model.

Equation 3.3 can be written as:

$$C(\mathbf{m_{n+1}}) = \left[\mathbf{d}_0 - \mathbf{d}\left(\mathbf{m}_n\right) + \alpha_n \mathbf{F}_n \mathbf{g}_n\right]^T \left[\mathbf{d}_0 - \mathbf{d}\left(\mathbf{m}_n\right) + \alpha_n \mathbf{F}_n \mathbf{g}_n\right] \tag{3.5}$$

Minimizing equation 3.5 relative to α_n (taking the derivative and making it equals to zero), leads to the optimal α_n:

$$\alpha_n = \frac{\left[\mathbf{F}_n \mathbf{g}_n\right]^T \left[\mathbf{d}_0 - \mathbf{d}\left(\mathbf{m}_n\right)\right]}{\left[\mathbf{F}_n \mathbf{g}_n\right]^T \left[\mathbf{F}_n \mathbf{g}_n\right]} \tag{3.6}$$

This reduces the problem for the step length to a single forward modeling in a perturbation in the velocity model $\mathbf{m} = \mathbf{m}_n + \epsilon \mathbf{g}_n$. It is cheaper than a step length computed by trial and error (Guarido et al., 2015).

For our tests, estimation of the step of equation 3.6 is done by using a single reference shot (the one in the center of the model).

3.1.3 The conjugate gradient

Equation 3.1 can be rewritten substituting the gradient $\mathbf{g}_n$ by the conjugate gradient $\mathbf{h}_n$ (Zhou et al., 1995; Vigh and Starr, 2008; Ma et al., 2010):

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n \mathbf{h}_n \tag{3.7}$$

where

$$\mathbf{h}_0 = \mathbf{g}_0, \beta_n = \frac{\mathbf{g}_n^T \left(\mathbf{g}_n - \mathbf{g}_{n-1}\right)}{\mathbf{g}_{n-1}^T \mathbf{g}_{n-1}}, \mathbf{h}_n = \mathbf{g}_n + \beta_n \mathbf{h}_{n-1} \tag{3.8}$$

The conjugate gradient is an algorithm that has the goal of finding the nearest local minimum. It is based on using conjugate directions (a combination of gradients of previous iterations and the current gradient), instead of the downhill approximation of the steepest-descent method. This implementation comes in hope to reduce the number of iterations, and increase the resolution of the inversion.

3.1.4 Impedance inversion

The tests done until now use the migration output (reflection coefficients) as the model update, as I used very low frequencies (starting in $2Hz$). However, I believe that the gradient must be processed and an impedance inversion should be applied (Margrave et al., 2010, 2011).

For normal incident P-waves, the reflection coefficients due to impedance contrast in the interface of two mediums is (Treitel et al., 1995):

$$R_i = \frac{\rho_{i+1}V_{i+1} - \rho_i V_i}{\rho_{i+1}V_{i+1} + \rho_i V_i} \tag{3.9}$$

where R_i is the reflection coefficient on the ith interface, ρ is the density and V is the P wave velocity propagation. The multiplication of density and velocity is the acoustic impedance I_i. For small contrasts of acoustic impedances in geological interfaces (usually < 0.3), equation 3.9 can be written:

$$R_i = \frac{\Delta I_i}{2I_i} \tag{3.10}$$

Assuming the reflection sequence as continuous in time, and taking the limit $\Delta_t \to 0$:

$$R(t) = \frac{1}{2}d\left(\ln I(t)\right) \tag{3.11}$$

Integrating over time and assuming constant density:

$$V(t) = V_0 e^{2\int_{t_1}^{t} R(\tau)d\tau} \tag{3.12}$$

The gradient can be inverted from reflection coefficients to impedance by taking the exponential of the integrated trace in time.

3.2 Simulation

3.2.1 Computation of the monochromatic averaged gradient approximation

The synthetic test is done in the Marmousi model (figure 3.1). Simulated acquired shots were generated using an acoustic finite difference algorithm, in a total of 102 shots. Shot interval is 100m, and a maximum of 401 receivers (varying in the edges of the model), with 2000m of maximum offset. The register time is 3s and sample rate is 4ms (figure 3.2). The dominant frequency of the wavelet is 5Hz. In the previous chapter is was 10Hz. I decided to change the dominant frequency to be the same as in Margrave et al. (2010).

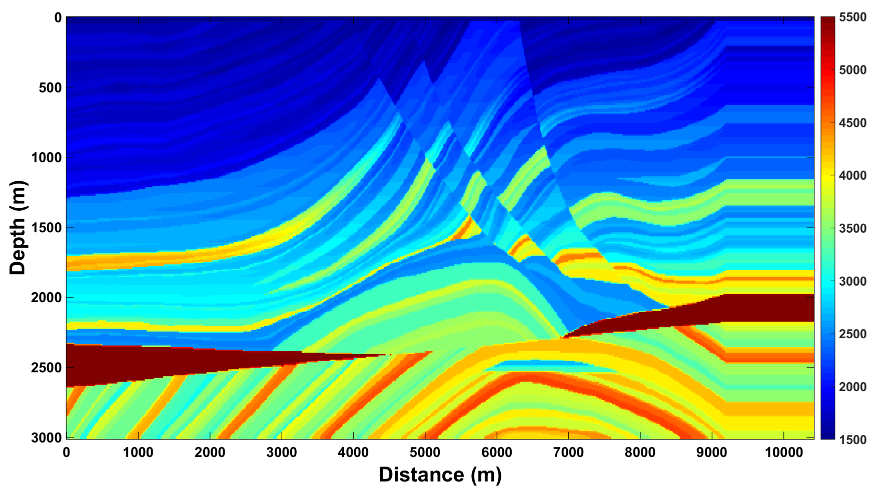


Figure 3.1: The Marmousi 2D model. The color bar indicates the wave propagation velocity in m/s.

During the FWI routine, synthetic shots are created with the same finite difference forward modeling algorithm, wavelet and acquisition parameters as the acquired shots. I am not giving any special treatment for the wavelet. I am just assuming that the wavelet is perfectly recovered. The only difference is the velocity model used to generate the synthetic shots. For the first iteration, the synthetic shots are generated over the starting model, usually a depth migration velocity (Virieux and Operto, 2009), and then the updated velocity models from the second iteration. The residuals are migrated with the PSPI migration

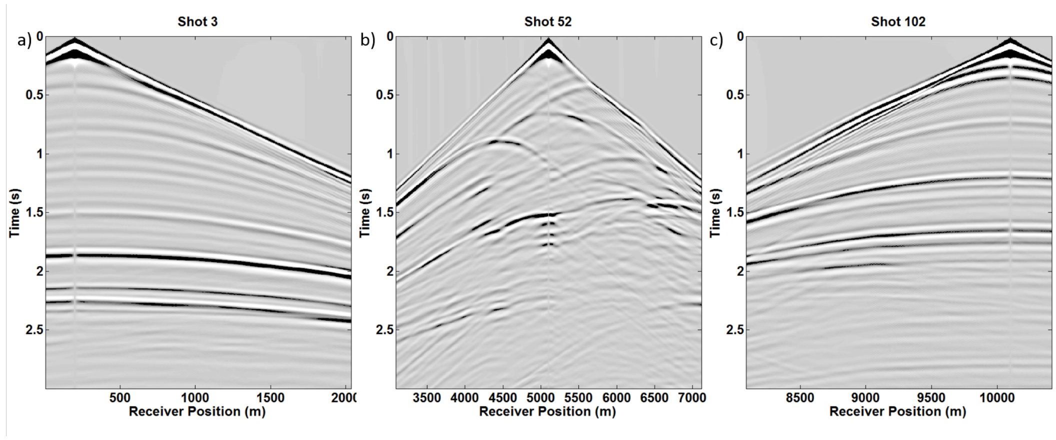
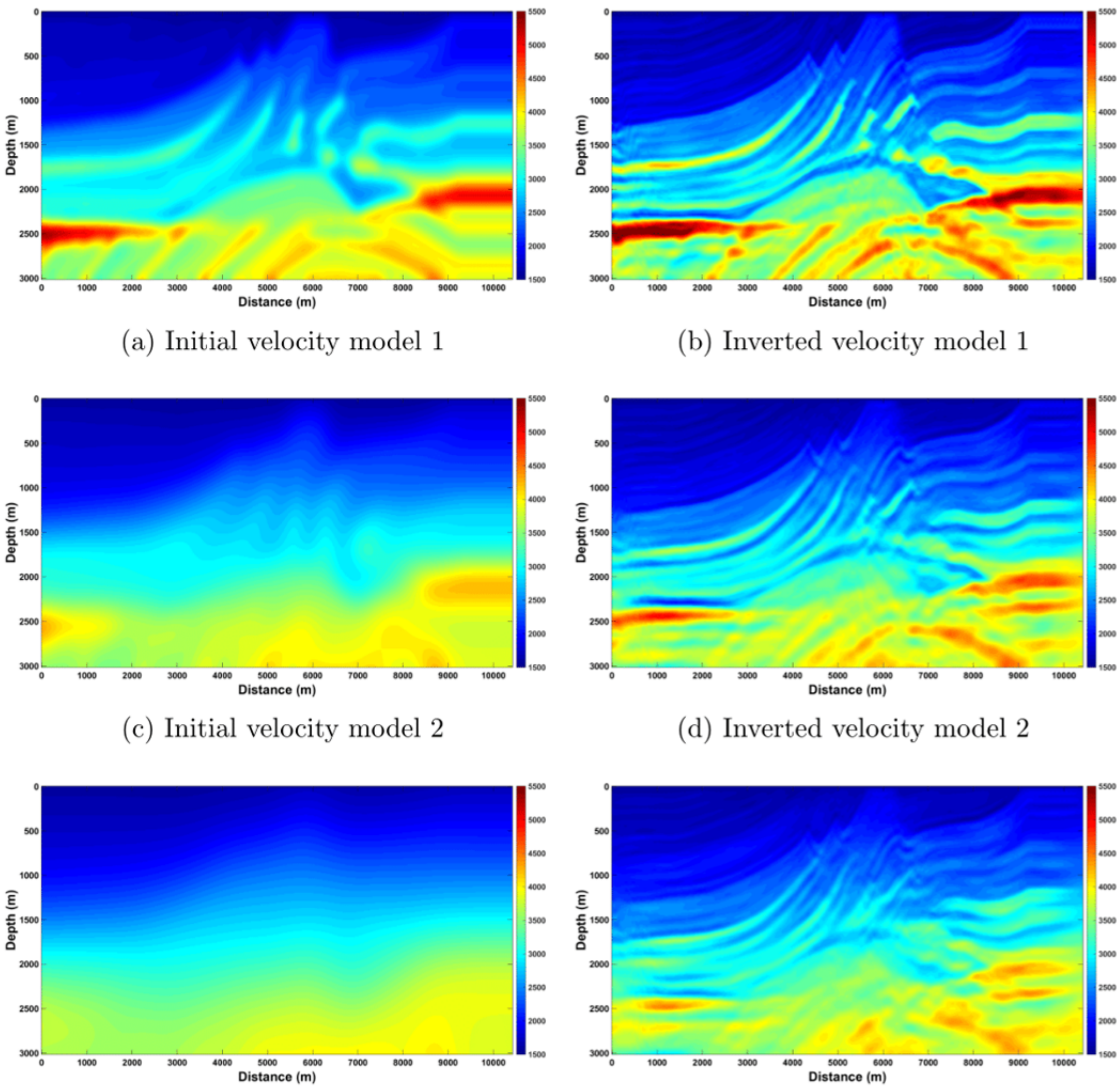


Figure 3.2: Three of the 102 shots used in the test. a) shot 3, b) shot 52 and c) shot 102.

with a deconvolution imaging condition. For the first iterations, only very low frequency are used ($2-4Hz$), with the top frequency of the band increasing by $2Hz$ when errors are stable (when variation of the 3 last iterations is less than 0.1%). Each frequency is migrated independently, i.e., for the range of $2-4Hz$ three migrations are necessary (one for $2Hz$, one for $3Hz$ and one for $4Hz$). All the migrated residuals are muted (Guarido et al., 2014), stacked, and smoothed (Margrave et al., 2010; Guarido et al., 2014). For now, an impedance inversion is not applied. The step length is calculated using equation 3.6 for each monochromatic gradient. The final gradient is the average (equation 3.2) of all scaled monochromatic gradients and it is used to update the model for the next iteration.

To analyze the initial guess, I am using the steepest-descent method. Application of the conjugate gradient will be studied later. Three different initial guesses are tested (figure 3.3). All the models are a smoothed version of the correct model 3.1. Each one of the initial models represent a different level of difficult for the routine. The model of figure 3.3a is a starting model closer to the global minimum, the model of figure 3.3c is more challenging (and is the main initial model for all the others tests) and the model of figure 3.3e contains only trends of the Marmousi model.



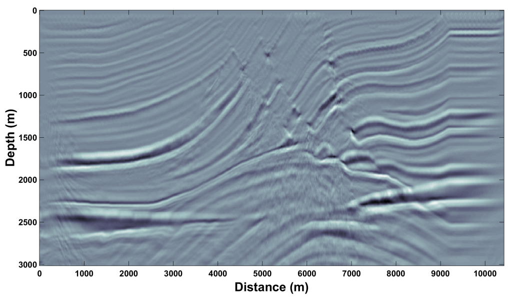
(a) Initial velocity model 1 (b) Inverted velocity model 1

(c) Initial velocity model 2 (d) Inverted velocity model 2

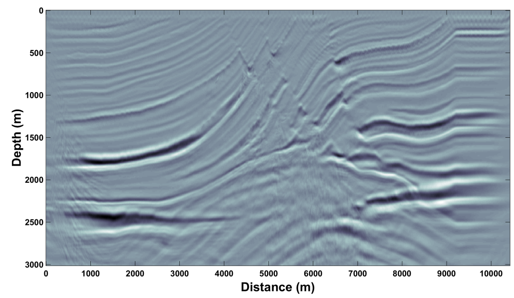
(e) Initial velocity model 3 (f) Inverted velocity model 3

Figure 3.3: The tested initial velocity guesses. All of them is the original Marmousi model convolved is a different Gaussian window.Also the inverted models. Closer to the global minimum the initial guess is, better is the inverted model.

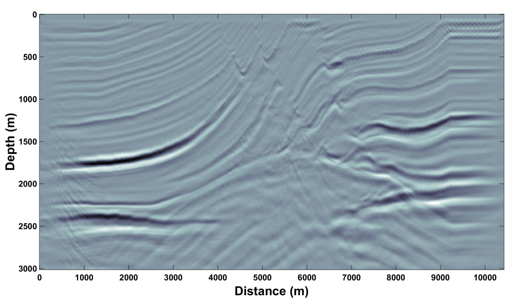
Figure 3.3 shows, as well, the inversion results using the correspondent initial model. It is notable that closer to the global minimum the initial model is (i.e., closer the initial model is to the true model), the inverted model has higher resolution. This can be explained by the quality of the migrated residuals. The interfaces are placed closer to the correct position and inversion has a higher change to converge to the global minimum. Figure 3.3b is a higher resolution model that contains mostly characteristics of the Marmousi model (figure 2.22), and is a remarkable result. When the initial guess starts to get far from the global minimum (as figures 3.3c and 3.3e), the inverted model loses resolution. However, the velocity models of figures 3.3d and 3.3f are still impressive (the model of figure 3.3d has, mostly, equal or more features than the good initial guess of figure 3.3a). I am confident that all the final models can be used as migration velocity.



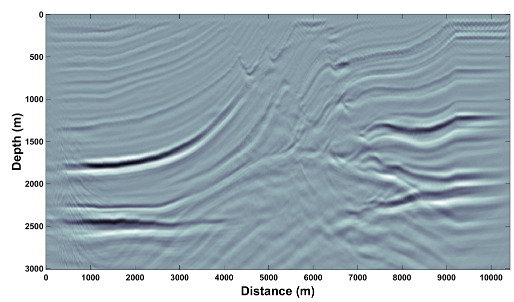
(a) Migrated shots using Marmousi model



(b) Migrated shots using velocity of figure 3.3b



(c) Migrated shots using velocity of figure 3.3d

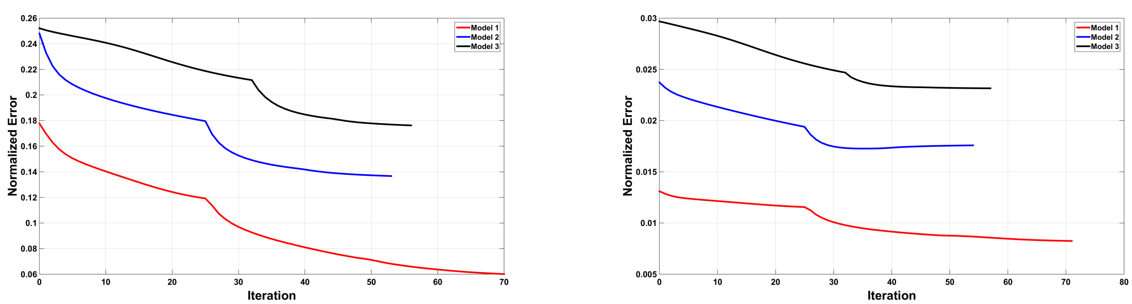


(d) Migrated shots using velocity of figure 3.3f

Figure 3.4: Migrated shots using the inverted models of figure 3.3. Better the model, higher is the migration quality.

The next step is to use the resulted models of figure 3.3 as migration velocities. A pre-stack depth migration (in here the same PSPI algorithm) is used to migrate each shot, and then they are stacked. Figure 3.4a is the migrated section using the Marmousi model (slightly smoothed). The figures 3.4b, 3.4c and 3.4d are migrated sections using velocities of figures 3.3b, 3.3d and 3.3f, respectively. Again, as better the model gets, the migrated section using an inverted velocity converges to the real solution.

Convergence is checked on figure 3.5a, where errors (normalized $norm-2$) of the reference shot is computed per iteration. For the three tests, errors decrease at each iteration until stability is reached at maximum frequency of $61Hz$. Note how the error decreases as inversion uses an initial guess that is closer to the global minimum. When analyzing figure 3.5b, I see similar behavior as the shot errors. The difference is on model 2 (blue line), with the minimum at iteration 34 (model deviation started to increase. The shot error may converge, but the model deviation can diverge), suggesting that the inversion encountered a local minimum.



(a) Shot errors

(b) Model errors

Figure 3.5: Errors ($norm-2$) of the inversions for each initial model. a) are the errors related to real synthetic reference shots and b) is the error of the inverted model of each iteration when compared to the Marmousi model.

Figure 3.6 is the 1D plot of the center row of the real, initial and inverted model for all the three tested models. In the left is the inversion using the best initial model and the final answer gets closer to the real one. As the initial guess gets far of the global minimum (center and right plots), I can still invert very reasonable models, but the resolution decreases as the distance of the initial model to the global minimum increases. Even though, all the runs were able to add important information to the initial guess.

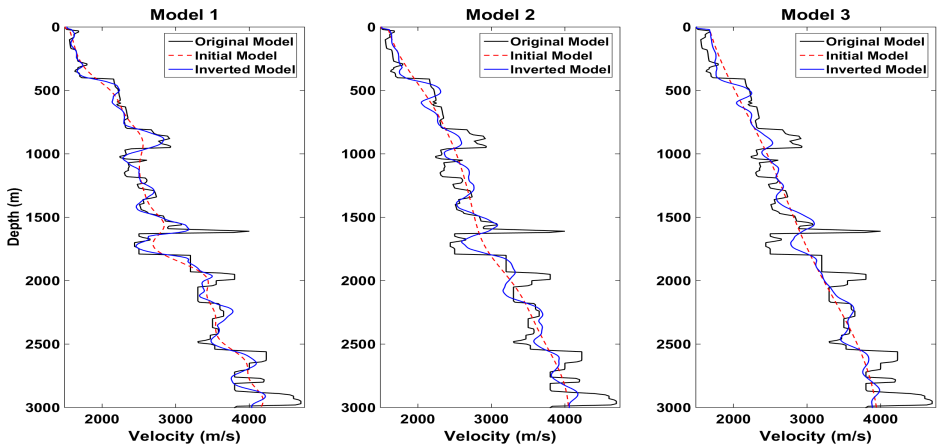


Figure 3.6: The 1D plot of the center row of the real, initial and inverted model for all the three tested models. The inverted model is more accurate when the initial model contains some information of the true model (figure 3.5).

3.2.2 Application of the conjugate gradient

Now the conjugate gradient is estimated using equation 3.8 and model is updated according to equation 3.7. Impedance inversion is not applied yet.

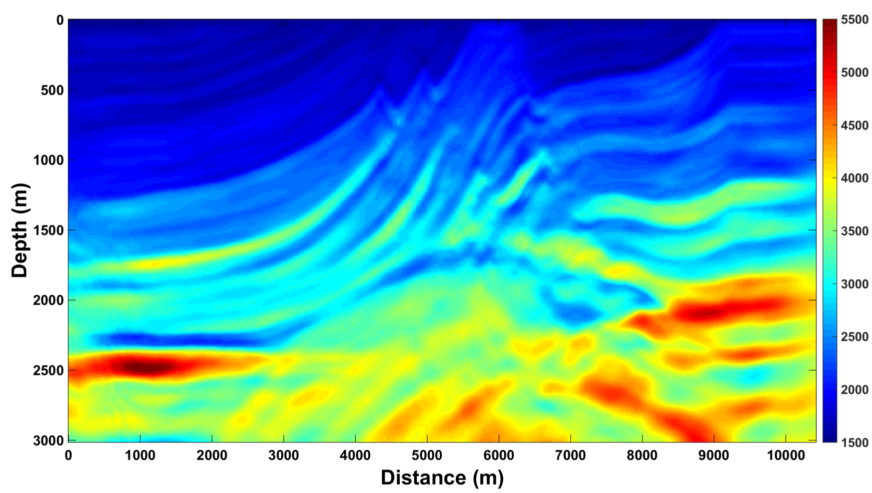
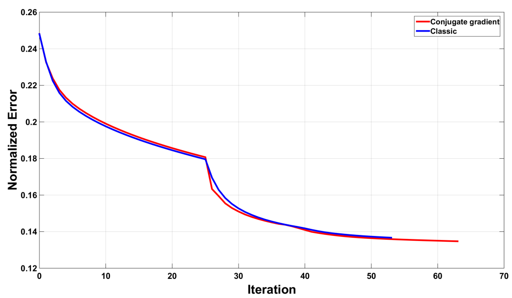
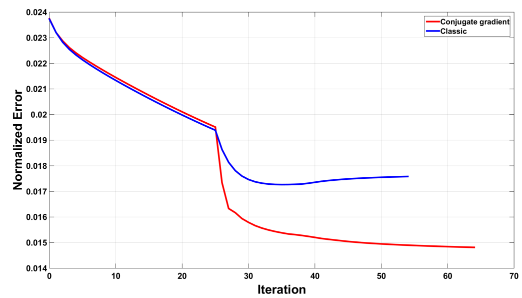


Figure 3.7: Inverted model using the conjugate gradient. The initial guess is the one on figure 3.3c. The final model has higher resolution than the model of figure 3.3d.

I used the same routine as I used on previous section, but the conjugate gradient is computed after the scaled monochromatic gradients are averaged. Figure 3.7 is the inverted model using as starting of figure 3.3c. Conjugate gradient improved the inversion quality if compared with the steepest-descent one (figure 3.3d).



(a) Shot errors



(b) Model errors

Figure 3.8: Errors of the inversions for each initial model. a) are the errors related to real synthetic reference shots and both classic and conjugate gradient show very similar behavior, and b) is the error of the inverted model of each iteration when compared to the correct model.

Reference shot errors (normalized $L2-norm$) of the conjugate gradient inversion (red line) is compared with the errors of the classic gradient inversion (blue line) on figure 3.8a. Both lines have a very close "path" until iteration 52, when the classic gradient stabilized while the conjugate gradient kept updating the model and reducing the error for 10 more iterations. This indicates that the conjugate gradient was best seized while inverting higher frequencies, providing a final model with improved quality. Actually, the real improvement of the conjugate gradient can be easily checked on figure 3.8b. The error of the models per iteration shows that the conjugate gradient leads the inversion closer to the global minimum. This is a very important observation, as the conjugate gradient strongly increase inversion resolution with insignificant computational cost.

Figure 3.9 is the 1D plot of the center row of the real, initial and inverted model comparing classic and conjugate gradients. Both inversions look close to each other. However, the conjugate gradient inversion (red dashed line) looks slightly closer to the Marmousi model (black line). As it was discussed before, the conjugate gradient method improves the inverted

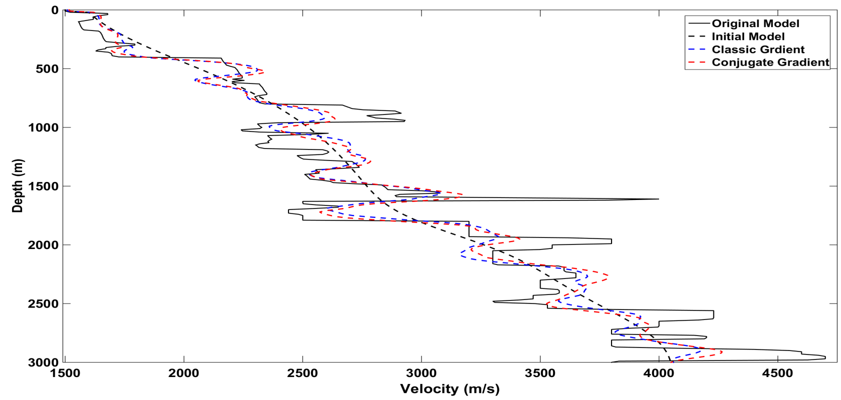
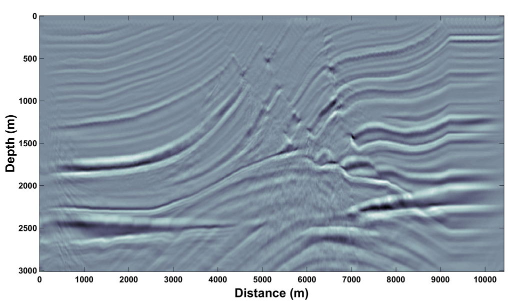


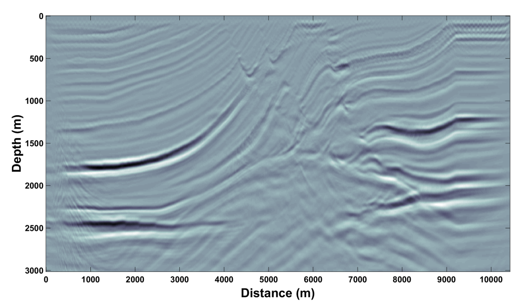
Figure 3.9: The 1D plot of the center row of the real, initial and inverted model comparing classic and conjugate gradients. Both inversions have close behavior, but the conjugate gradient inversion shows a slightly better (closer to the real model) result.

model if compared to the steepest-descent.

Analyzing the migrated section of the figure 3.10b, that used as velocity model the conjugate gradient inversion, there is significant difference when compared with the migrated section that used the steepest-descent inversion model. Shallow structures are clearer, more correctly positioned, and deep layers are less shifted when compared with the migration using Marmousi velocity of figure 3.10a.



(a) Migrated shots using Marmousi model



(b) Migrated shots using model of figure 3.7

Figure 3.10: Comparing the migrated section when using a) the Marmousi model and b) the inverted model of figure 3.7. The second one is very similar to the one of figure 3.4c.

The conjugate gradient is cheap to be applied and results in a great improvement to the inversion quality. It increases the number of iterations to finish the inversion because stop criteria is not reached, meaning that it keeps improving the model for a longer period.

3.2.3 Impedance inversion

Impedance inversion is applied according to equation 3.12 for each monochromatic gradient, and then averaged. However the resulting gradient shown to be phase shifted. Margrave et al. (2010) computed the required phase rotation by least squares minimization using well log as reference. I compute the least squares minimization comparing pilot shot with a synthetic one. The gradient is then converted from depth to time, phase shifted, and then converted back to depth. After, the step lengths for each monochromatic gradient are estimated, then averaged, and the conjugate gradient is computed to update the model.

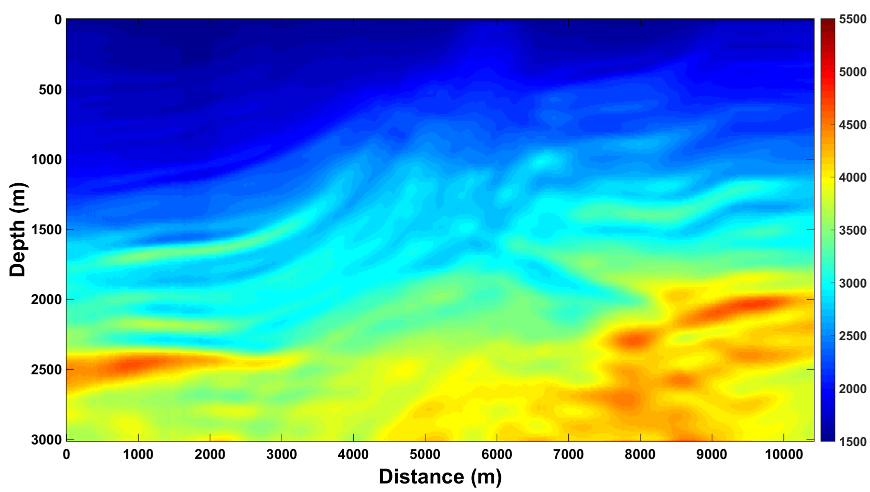


Figure 3.11: Inverted model using the conjugate gradient and impedance inversion. The initial guess is the one on figure 3.3c. I still have a big room for improvement.

The inverted model of figure 3.11 is promising with great potential for improvement, but I could not have an inversion with the same quality of the ones that used scaled reflection coefficients contrasts. Some of the main structures were inverted and the most impressive observation is the inversion of the thick high velocity bodies. Apparently, the use of impedance inversion is a good solution in areas with large continuous structures.

I believe that the main reason for the inversion had converged to a local minimum is due to the phase rotation. I compute the necessary phase by comparing the pilot shot with a synthetic shot (finding the phase shift required to minimize their difference). Maybe a more sophisticated strategy to optimize the phase is required. In chapter 4, I introduce a band-limited impedance inversion (BLIMP) successfully.

3.3 Conclusions

I showed an approximation of the gradient that is the interpolation of pseudo-gradients formed from the monochromatic migrated residuals. It showed to have higher accuracy and resolution than the gradient descent method of chapter 2, but with increased cost to compute the gradient due to the number of migrations required at each iteration (it is the same as the number of unitary frequencies inside the chosen frequency band). The resolution of the inverted model is also improved by the implementation of the conjugate gradient, with neglectable addition in cost. Estimating the step length by an analytical solution (Pica et al., 1990), reduced the number of forward modeling from 21 (our old parameter) to only 1, but its accuracy is equivalent. The impedance inversion of the gradient was tested with promising ending model, and working will continuing on that in the next chapter.

Chapter 4

A band-limited impedance inversion based gradient

This chapter focus on an impedance inversion of the gradient prior the estimation of the step length (Guarido et al., 2016). I apply a band-limited impedance inversion (BLIMP) in the gradient, using the initial model to fill the low-frequency content of the velocity model. This method is stable, faster then the monochromatic averaged gradient, but this saving in cost comes with some loss in the resolution in the shallow area of the inverted model, but with improvements at the deeper area.

4.1 Theory

4.1.1 Gradient method

As presented previously on section 2.1, the goal of the full waveform inversion is to minimize the objective function:

$$C(\mathbf{m}) = ||\mathbf{d}_0 - \mathbf{d}(\mathbf{m})||^2 = ||\Delta\mathbf{d}(\mathbf{m})||^2 \tag{4.1}$$

where Δd is the data residual, m is the model (P-wave velocity) and $||$ represents the norm-2. The minimization is done by calculating the Taylor's expansion of the objective function of the equation 4.1 around a perturbation δm of the model and taking the derivative equal to zero (Tarantola, 1984; Pratt et al., 1998; Virieux and Operto, 2009). The solution is:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - H_n^{-1}\mathbf{g}_n \tag{4.2}$$

where H is the Hessian (or sensitive matrix), g is the gradient computed by back-propagating the data residual and n is the n-th iteration. It is known as the Newton method. For the steepest-descent method, the Hessian matrix is approximated to the identity matrix:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n \mathbf{g}_n \tag{4.3}$$

where α is the step length (or scale factor) is estimated using equation 3.6. At this part, the gradient is understood to be the PSPI migrated residuals with a deconvolution imaging condition (Margrave et al., 2010, 2011; Pan et al., 2013; Guarido et al., 2014). g_n is the conjugate gradient as showed on equation 3.8.

4.1.2 Band-limited impedance inversion (BLIMP)

On chapter 3 I used the stacked migrated residuals (reflection coefficients) to update the model (Guarido et al., 2014, 2015). However the FWI formulation implies that by integrating the trace of the migrated residuals to end up with the gradient converted from reflection coefficients to velocity. In other words, an impedance inversion should be applied (Margrave et al., 2010, 2011).

Previously, on section 3.1.4 and in Guarido et al. (2015), an simple exponential of the trace integration in time is used as impedance inversion with promising results. Now I am using the algorithm of Ferguson and Margrave (1996) to apply a *band-limited impedance inversion* (BLIMP) in the gradient (instead of a trace integration as showed on section 3.1.4). The BLIMP goal is to add the missing low frequency of a seismic trace during the estimation of the impedance inversion by extracting if from a sonic log. To simulate a situation where no well logs are available, I chose to use the initial model as pilot to include the low frequency content. At this point I am assuming that the low frequency trend of the initial model is equivalent to the Earth's low frequency trend for that specific area.

BLIMP is applied in time domain. Inputs (initial model and migrated residuals) are converted from depth to time using the current inverted velocity model and lately, after the process, converted back to depth using the same model. The first step of BLIMP is to specify the relationship between the migrated residuals and the pilot model. Thus, the migrated residuals need to be "converted" to acoustic impedance (P-wave velocity) using the normal incidence approach (Treitel et al., 1995):

$$R_i = \frac{\rho_{i+1}V_{i+1} - \rho_i V_i}{\rho_{i+1}V_{i+1} + \rho_i V_i} \tag{4.4}$$

where R_i is the reflection coefficient on the ith interface, ρ is the density and V is the P wave velocity propagation. The multiplication of density and velocity is the acoustic impedance I_i. For small contrasts of acoustic impedances in geological interfaces (usually < 0.3), equation 4.4 can be written:

$$R_i = \frac{\Delta I_i}{2I_i} \tag{4.5}$$

Assuming the reflection sequence as continuous in time, and taking the limit $\Delta_t \to 0$:

$$R(t) = \frac{1}{2}d\left(\ln I(t)\right) \tag{4.6}$$

Integrating over time and assuming constant density:

$$V(t) = V_0 e^{2\int_{t_1}^{t} R(\tau)d\tau} \tag{4.7}$$

Ferguson and Margrave (1996) replace the reflection coefficients $R(\tau)$ by a scaled reflectivity $S(\tau) = 2R(\tau)/\gamma$. Then 4.7 becomes:

$$V(t) = V_0 e^{\gamma\int_{t_1}^{t} S(\tau)d\tau} \tag{4.8}$$

γ is a scale factor estimated by minimizing the objective function:

$$\Gamma = \sum \left[V_{pilot}(\omega) * B + (\gamma - 1)V(\omega)\right]^2 \tag{4.9}$$

where $V(\omega)$ is the Fourier spectra of equation 4.8, $V_{pilot}(\omega)$ is the Fourier spectra of the pilot velocity model and B is a low-pass filter.

The output impedance is the inverse Fourier transform of the equation 4.10:

$$V_{out}(\omega) = V_{pilot}(\omega) * B + \gamma V(\omega) \tag{4.10}$$

Ferguson and Margrave (1996) use the methodology for impedance (velocity multiplied by the density). I simplified it for acoustic inversion only (constant density).

Equation 4.10 shows that the scaled Fourier spectra of the input migrated residuals, converted to velocity, is added to the low frequency content of the pilot velocity. This means that the missing low frequency on seismic data can be compensated by an initial model that contains a good estimation of the local linear velocity trend. Usually a migration velocity is used for this task.

4.2 Simulation

4.2.1 Input data

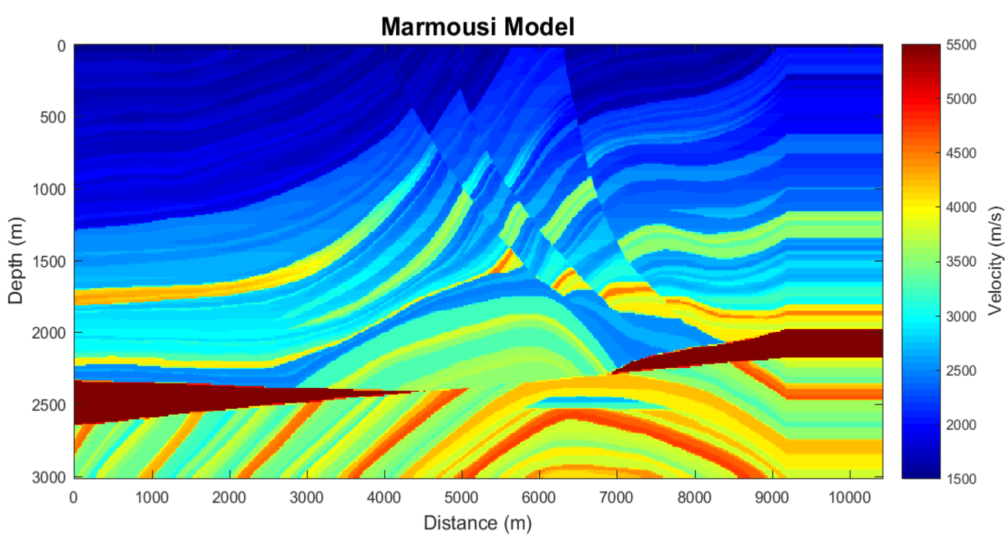


Figure 4.1: The Marmousi 2D model. The color bar indicates the wave propagation velocity in m/s.

The Marmousi model is used for the synthetic test (figure 2.22). Pilot shots (used as the real shots), were created using an acoustic finite difference algorithm. In the total, 102 shots are used to simulate field data. Shot spacing is 100m. Each shot has a maximum of 401 receivers (varying in the edges of the model) totalizing 2000m of maximum offset. The register time is 3s and sample rating of 4ms (figure 2.23). The dominant frequency of the wavelet is 5Hz.

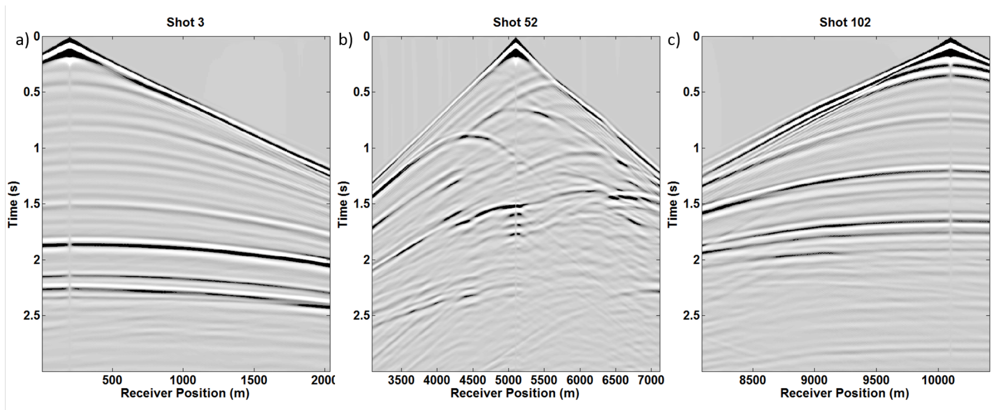


Figure 4.2: Three of the 102 shots used in the test. a) shot 3, b) shot 52 and c) shot 102.

During the FWI routine, synthetic shots are created with the same finite difference forward modeling algorithm, wavelet and acquisition parameters as pilot shots. They differ from each other by the velocity choice, initially a guessing model, usually a depth migration velocity (Virieux and Operto, 2009), and then the update models. The residuals are then migrated with the PSPI migration with a deconvolution imaging condition. For the first iterations, only very low frequencies are used ($4-6Hz$), and the range is increased by $2Hz$ when errors are stable (variation of the 3 last iterations is less than 0.1%). Each residual is migrated separately using the initial or updated model. A mute is applied before stacking the migrated residuals (Guarido et al., 2014), and the stack is preconditioning by convolving it with a 2D Gaussian window (Margrave et al., 2010; Guarido et al., 2014). Ferguson and Margrave (1996)'s algorithm for impedance inversion is used after the conjugate gradient step (Guarido et al., 2016). Lastly, the step length is estimated using equation 3.6, and the model is updated for the next iteration.

4.2.2 Initial model as well control

The band-limited impedance inversion (BLIMP) algorithm is applied on the migrated residuals to estimate the gradient, on an expectation to compensate the lack of low frequencies on the data (the lowest frequency of the seismic data used is 4Hz) following those steps:

- Residuals are computed as the difference between synthetic and acquired seismic shots.
- Each residual is depth migrated using the PSPI algorithm with a deconvolution imaging condition.
- The migrated residuals are muted and stacked to form a pseudo-gradient.
- The pseudo-gradient and current model are stretched from depth to time.
- The band-limited impedance inversion is applied on the gradient using the current model's low frequency (in time).
- Gradient is stretched from time to depth using current model.
- Step length is computed using equation 3.6.
- Model is updated using the scaled gradient.

Even if I compute a gradient that points on the opposite direction to the global minimum and estimate a step length to converge faster to the same point, the gradient descent method is known to converge to a local minimum (Pratt et al., 1998). One way to minimize this effect is to start the inversion with very low frequencies (multi-scale approach). Another way is to start the inversion closer to the global minimum (a good initial model).

I expect a better result for the model closest to the global minimum because it enhances the gradient in two ways: 1) the migrated residuals are in a more correct place with the best model and 2) more details of the low frequency in the initial model (1 to 3Hz) results on a better impedance inversion of the migrated residuals.

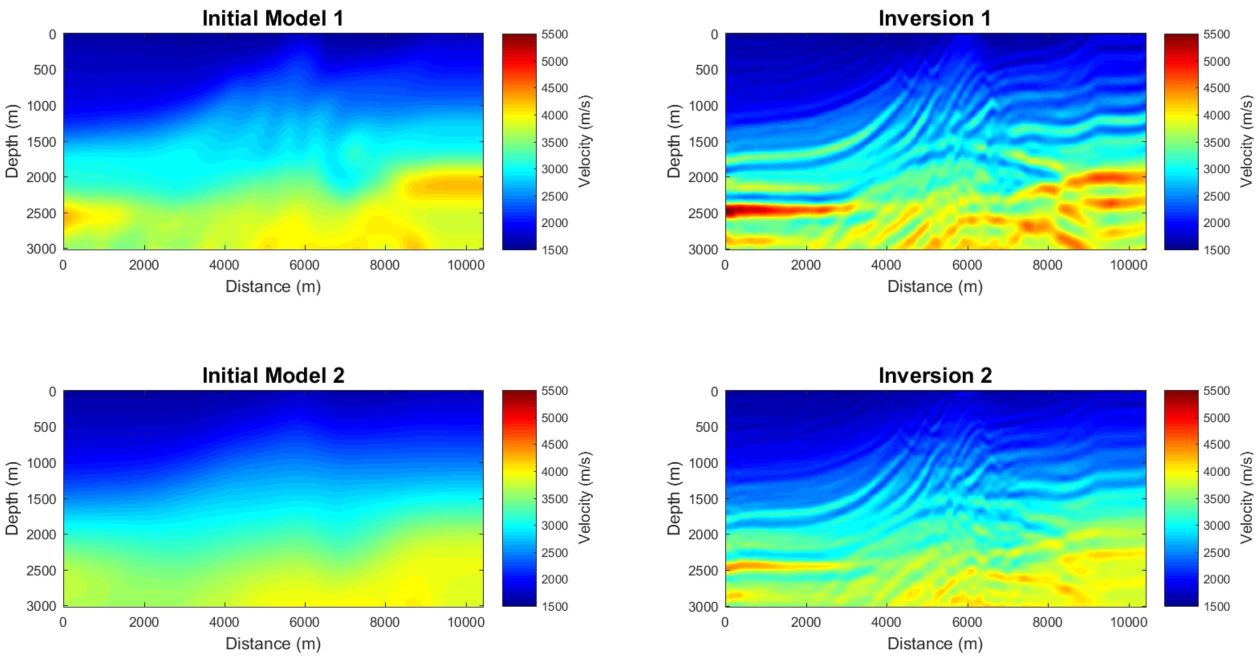


Figure 4.3: Inversion as a function of the initial model. The inversion 1 has more details than the inversion 2 due to the initial model be closer to the local minimum.

Figure 4.3 illustrates the different inversions with varying qualities of the initial model. In both cases, the multi-scale approach is implemented. When the starting point is closer to the global minimum, the inversion leads to a higher resolution resulting model (Guarido et al., 2015). In both cases, it is noticeable of how much information is included to the model. Most of the features from low to mid frequencies were successfully recovered. The faults with high dip angle in the shallow (central portion of the model) are present on both inversions. Even the simulation of a gas anomaly ($x = 6500, z = 2500$) could be interpreted for the inversion with the best initial model.

The algorithm worked successfully for low to mid frequencies (up to 16Hz) and had no updates on higher frequencies (the gradient exists but the step length goes to zero, meaning that the inversion was too close to a minimum point).

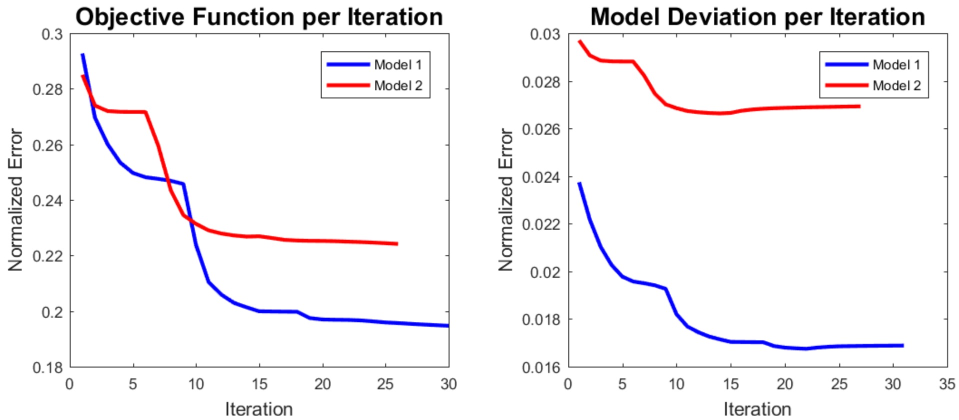


Figure 4.4: Plots of the objective function per iteration (left) and model deviation per iteration (right) of models 1 (blue) and 2 (red).

For the validation of the tests, the errors are plotted on figure 4.4. On the left is the objective function (difference between acquired and synthetic shots) of the two models (1 in blue and 2 in red) and on the right the model deviation (difference between inverted and correct Marmousi models). The most notorious observation is quite obvious: higher resolution of the initial model (closer to the global minimum) leads to a higher resolution inverted model.

It is also possible to observe another behavior on figure 4.4. For both initial model tests there are two steep decrease of the errors: one when I start to update the initial model at very low frequencies ($4 - 6Hz$) and another when the inversion reaches the dominant frequency of the data ($\sim 12Hz$). When comparing the two plots, an odd observation came to our eyes. Even with the objective function decreasing at each iteration, the model deviation has a slight increase on both models. That is the point when the inversion reached frequencies on the data with low signal to noise ratio (numerical noise).

At this point I remember the update is based on equation 3.1. It is, considering the computer power to calculate it, cheaper than equation 3.2, method used by Guarido et al. (2015), as it requires only one migration pass per iteration. Figure 4.5 is a group of images comparing both works. BLIMP FWI model (top left) seems to work better on deeper events, shows more continuity of the geological structures at the model and suffer from less borders effects, while the monochromatic averaged gradient FWI model (bottom left) did a better job on the shallow part of the model, and shows a "cleaner" image. In the end, both inverted models are good representations of the Marmousi model. However, the costs on computing the gradient are very different. I could change the interpretation of the gradient to make its estimation cheaper and still have a similar model.

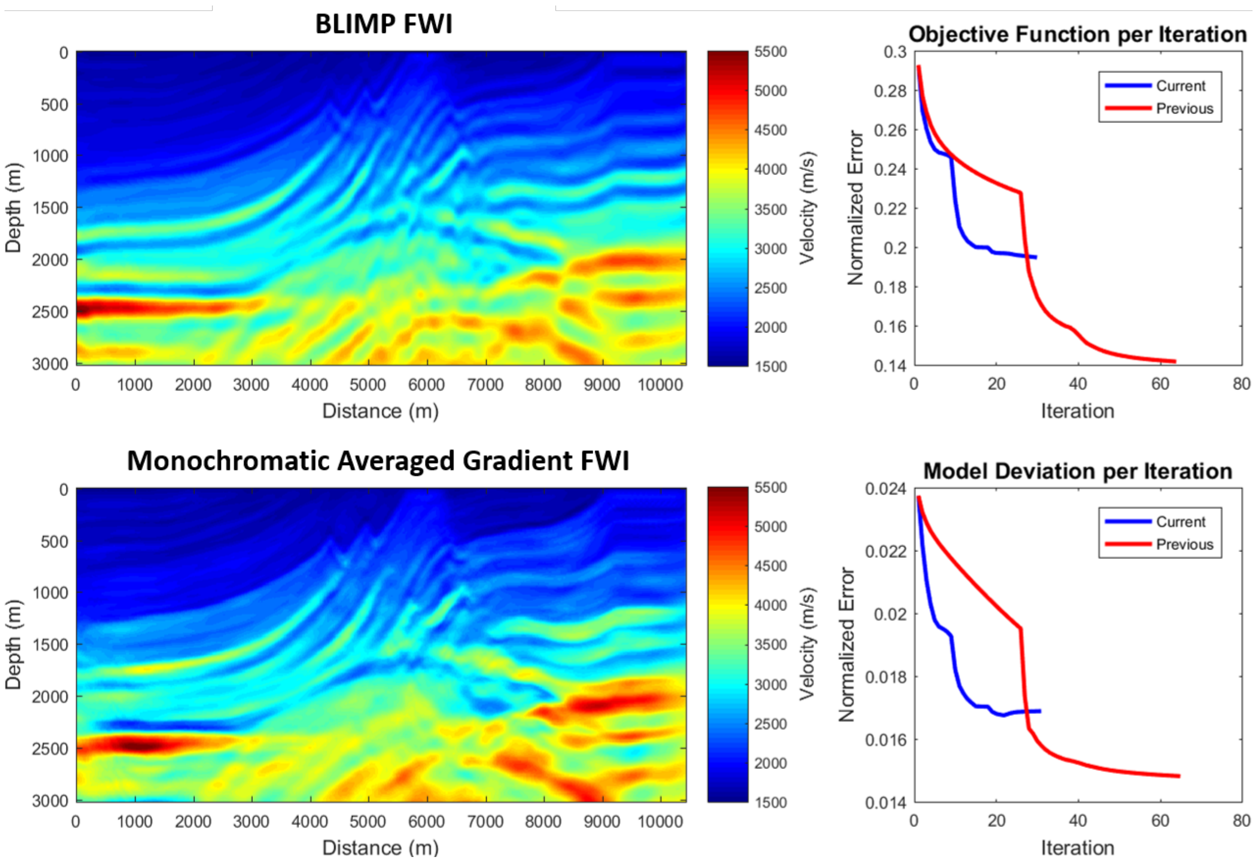


Figure 4.5: On the left are the resulting velocity models on BLIMP FWI (top) and monochromatic averaged gradient FWI (bottom). On the right are the comparison of the objective function (top) and models deviations (bottom) per iteration.

Checking the plots of the objective function behavior (top right) and model deviation (bottom right) per iteration, BLIMP FWI (blue lines) converge faster (less number of iterations) than the monochromatic averaged gradient method (red lines). Actually it converges with less than a half of iterations of the other method. But the errors do not look similar to each other as the models do. The shot differences are much lower on the monochromatic averaged gradient inversion than on the current one. However, the same can not be said about the model deviation. There is still a difference but not so large as the shot differences. The normalized model errors were reduced from 0.024 to 0.017 (BLIMP FWI) and 0.015 (monochromatic averaged gradient FWI). In others words, the resulted models are equivalent, but the costs are not.

The inversion applying the BLIMP algorithm on the migrated residuals showed to be cheaper than the work on Guarido et al. (2015) in two points: the costs on computing the gradient, and the number of iterations required to reach convergence.

4.3 Conclusions

To apply a band-limited impedance inversion (BLIMP) in the gradient provides me with a model update that is actually a residual velocity model, and not a model of reflection coefficients. As I am using the initial model to fill the low frequency gap of the seismic data, it must contain the general trend of the subsurface. When compared to the monochromatic averaged gradient of chapter 3, the BLIMP gradient loses resolution for shallower events, but it has improved resolution at deeper areas. The BLIMP gradient also has reduced cost, as I am not migrating individual frequencies, but the whole frequency band at once (for each iteration).

Chapter 5

The forward modeling-free gradient: a pre-stack approximation

This chapter is focused on the implementation of a new interpretation of the gradient in the FWI theory. We treat the gradient as the difference between the current model to the impedance inversion of the migrated acquired data. This method requires no forward modeling to estimate the gradient, only two forward modelings to estimate the step length using Pica et al. (1990)'s algorithm. It is stable and cheaper than any method presented until now, but the inverted model has comparable resolution.

5.1 Theory

5.1.1 New approximation for the gradient

Computing the gradient requires three very known seismic processing steps: PSDM of the residuals, stacking and impedance inversion (Guarido et al., 2016). We call those steps as operators: M for migration, S for stacking and I for impedance inversion. Equation 3.1 can be re-written as:

$$\begin{aligned}\mathbf{m}_{n+1} &= \mathbf{m}_n - \alpha_n \mathbf{g}_n \\ &= \mathbf{m}_n - \alpha_n I\left\{S\left[M\left(\Delta\mathbf{d}(\mathbf{m_n})\right)\right]\right\} \\ &= \mathbf{m}_n - \alpha_n I\left\{S\left[M\left(\mathbf{d}_0 - \mathbf{d}(\mathbf{m_n})\right)\right]\right\} \end{aligned} \tag{5.1}$$

where $\Delta\mathbf{d}(\mathbf{m_n})$ is the n-th iteration residual $\mathbf{d}_0 - \mathbf{d}(\mathbf{m_n})$, $\mathbf{d}_0$ is the acquired data, $\mathbf{d}(\mathbf{m_n})$ is the synthetic data of the n-th iteration and $\mathbf{m_n}$ is the n-th iteration inverted model. For simplification and easier visualization, let's set $\mathbf{d}(\mathbf{m_n}) = \mathbf{d_n}$. Then equation 5.1 is:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n I \left\{ S \left[M \left(\mathbf{d}_0 - \mathbf{d_n} \right) \right] \right\} \tag{5.2}$$

Considering the linearity property of the migration operator, the acquired and synthetic shots can be migrated separately:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n I \left\{ S \left[M \left(\mathbf{d}_0 \right) - M \left(\mathbf{d_n} \right) \right] \right\} \tag{5.3}$$

The next step is to use the linearity property of the stacking operator. Then equation 5.3 becomes:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n I \left\{ S \left[M \left(\mathbf{d}_0 \right) \right] - S \left[M \left(\mathbf{d_n} \right) \right] \right\} \tag{5.4}$$

For the impedance inversion operator, the first approximation we tried was $I(x_1 - x_2) = I_0 \frac{I(x_1)}{I(2_2)}$. Then equation 5.4 can be written as:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n I_0 \frac{I \left\{ S \left[M \left(\mathbf{d}_0 \right) \right] \right\}}{\underbrace{I \left\{ S \left[M \left(\mathbf{d_n} \right) \right] \right\}}_{\text{Current model}}} \tag{5.5}$$

where the denominator is the impedance inversion of the stacked depth migrated synthetic shots, and it is equivalent to the velocity model on which the forward modeling algorithm is ran into. In others words, it is the model of the current iteration, $\mathbf{m}_n$. Then equation 5.8 can be simplified to:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n I_0 \frac{I \left\{ S \left[M \left(\mathbf{d}_0 \right) \right] \right\}}{\mathbf{m}_n} \tag{5.6}$$

Equation 5.6 shows to be unstable as it can be divided by zero (as we are inverting for lower frequencies greater than 3Hz and not capturing the Earth linear trend, it can result on zeros and even negative impedances). On a tentative to avoid this we added a *stabilization factor* SF:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n I_0 \frac{I\left\{S\left[M\left(\mathbf{d}_0\right)\right]\right\}}{\mathbf{m}_n + SF} \tag{5.7}$$

Choosing a value for the stabilization factor showed to be challenging. A too small value can lead areas of the gradient to have a huge value compared to its neighbors. A large value (around 1 and above) can change the denominator significantly. And the stabilization factor needed to be chosen by trial and error at each iteration. To avoid the division of equation 5.7 looks to be the best strategy at this point. So I took the risk to say that the impedance inversion operator is *approximately* linear (for that, we assume Earth impedance to follow a linear trend and a Taylor expansion of the reflection coefficients is used to estimate the update as a perturbation of Earth's impedance), and we end up with another solution for equation 5.4:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n (I\left\{S\left[M\left(\mathbf{d}_0\right)\right]\right\} - \underbrace{I\left\{S\left[M\left(\mathbf{d_n}\right)\right]\right\}}_{\text{Current model}}) \tag{5.8}$$

Again, on the second hand of the gradient approximation we have the model of the current iteration $\mathbf{m}_n$. Then equation 5.8 is simplified to:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n (I\left\{S\left[M\left(\mathbf{d}_0\right)\right]\right\} - \mathbf{m}_n) \tag{5.9}$$

Equation 5.9 understands the gradient as a residual impedance inversion of the acquired data relative to the current model. With this approximation, the gradient can be estimated without the need of forward modelings and source estimation. The objective function is minimized during the estimation of the step length (that requires 2 synthetic data: one for the current model and one for a gradient perturbation of the same model). I previous chapters, at each iteration, it was required 105 forward modelings (104 to estimate the gradient and 1 more for the step length) on the Marmousi model. With the approximation of equation 5.9, it only requires 2 forward modeling to estimate the step length (1 for the current model and 1 for a perturbation of the model, showed on equation 3.6). Actually, the amount of forward modelings required for any project does not depend on how many shots it has, but on how many control points for the step length estimation we want to use (2 for each control location).

5.1.2 Linearity of the migration operator

Claerbout (1971) suggests to map a reflection point (or layer) by a cross-correlation of the downgoing and upgoing wave fields:

$$Im(x, z) = g(x, z, t) \otimes u(x, z, t) \tag{5.10}$$

where $u(x, z, t)$ is the upgoing residuals wavefield $d_m - d_0$, d_m is the synthetic data using model m, d_0 is the acquired data and $g(x, z, t)$ is the downgoing source wavefield. From now on we will omit the coordinates x and z on the notation, and keep in mind that we are imaging in depth. In the frequency domain (after a Fourier Transform) equation 5.10 becomes:

$$Im = U(\omega)G^*(\omega) \tag{5.11}$$

where * denotes the complex conjugate, $U(\omega)$ and $G(\omega)$ are the Fourier Transform of $u(t)$ and $g(t)$, respectively. The linear property of the Fourier Transform leads to $U(\omega) = D_m(\omega) - D_0(\omega)$, where $D_m(\omega)$ and $D_0(\omega)$ are the Fourier Transform of d_m and d_0, respectively. From equation 5.11:

$$\begin{aligned}
Im &= U(\omega)G^*(\omega) \\
&= [D_m(\omega) - D_0(\omega)]\, G^*(\omega) \\
&= D_m(\omega)G^*(\omega) - D_0(\omega)G^*(\omega) \\
Im(x, z) &= g(x, z, t) \otimes d_m(x, z, t) - g(x, z, t) \otimes d_0(x, z, t)
\end{aligned} \tag{5.12}$$

This means that applying the imaging condition of equation 5.10 on the residuals u is equivalent to subtract the imaging condition of d_0 to the d_m, proving the linearity property of the migration operator.

5.1.3 Linearity of the stacking operator

For a common imaging point of a seismic survey we have N traces tr_{u_i} of the residuals $tr_{u_i} = tr_{m_i} - tr_{0_i}$ (i is the shot position or number, m relates the trace to the synthetic shot and 0 to the acquired data). Stacking them is:

$$S_u = \sum_{i=1}^{N} \frac{tr_{u_i}}{N} \tag{5.13}$$

Now solving the sum of the residual terms of equation 5.13 we demonstrate the linear property of stacking operator:

$$\begin{aligned}
S_u &= \sum_{i=1}^{N} \frac{tr_{u_i}}{N} \\
&= \sum_{i=1}^{N} \frac{tr_{m_i} - tr_{0_i}}{N} \\
&= \frac{1}{N}(tr_{m_1} - tr_{0_1} + tr_{m_2} - tr_{0_2} + \cdots + tr_{m_N} - tr_{0_N}) \\
&= \frac{1}{N}[tr_{m_1} + tr_{m_2} + \cdots + tr_{m_N} - (tr_{0_1} + tr_{0_2} + \cdots + tr_{0_N})] \\
&= \frac{1}{N}(\sum_{i=1}^{N} tr_{m_i} - \sum_{i=1}^{N} tr_{0_i}) \\
&= S_m - S_0
\end{aligned} \tag{5.14}$$

5.1.4 Linearity approximation of the impedance inversion operator

Another way to re-write the reflection coefficients R and impedances I of equation 3.9 is given by Treitel et al. (1995) for the reflection at point k is:

$$I_k = I_0 \prod_{k=1}^{N} \left(\frac{1 + R_k}{1 - R_k} \right) \tag{5.15}$$

The next step is to apply a Taylor expansion of the denominator. But first let's understand what values the reflection coefficients can have. A strong water bottom reflector (water velocity = 1500m/s and first layer velocity = 2100m/s) has a reflection coefficient of around 0.21, which squared is 0.04. On a "well behaved" Earth (not considering the water bottom and high impedance layers as salt or basalt), the reflection coefficients have values of around 0.03 or less, which squared are 0.0007 or less. The reason of this analysis is to neglect terms of power 2 and higher on the Taylor expansion and products. With this information in mind, equation 5.15, by a Taylor expansion of the denominator and neglect term with the power of 2 or higher, is approximately:

$$\begin{aligned} I_k &= I_0 \prod_{k=1}^{N} \left[(1 + R_k)(1 + R_k + R_k^2 + R_k^3 + \cdots) \right] \\ &\approx I_0 \prod_{k=1}^{N} \left[(1 + R_k)(1 + R_k) \right] \end{aligned} \tag{5.16}$$

Equation 5.16 leads to an easier solution for the impedance. Now we can open the product and, as previously, neglect powers of 2 and higher of the reflection coefficients:

$$\begin{aligned} I_k &= I_0 \prod_{k=1}^{N} (1 + 2R_k + R_k^2) \\ &\approx I_0 \prod_{k=1}^{N} (1 + 2R_k) \\ &= I_0 (1 + 2R_1)(1 + 2R_2)(1 + 2R_3) \cdots (1 + 2R_N) \\ &= I_0 (1 + 2R_1 + 2R_2 + 2R_3 \cdots 2R_N + O(\text{power} \geq 2)) \\ &= I_0 (1 + 2 \sum_{k=1}^{N} R_k) \end{aligned} \tag{5.17}$$

This last result ended up to be very interesting. The sum of the reflection coefficients is a perturbation linear impedance trend with slope equals to $2I_0$ and intercept I_0. For the

FWI problem, if we have an initial model that contains the local Earth's linear trend (low frequency content), we can update only the perturbation, or the cumulative sum of the migrated acquired data with the low frequency removed. The resulting impedance perturbation will have zeros and negative values. As a model update, this means reducing equation 5.17 to:

$$I_{k_{update}} \approx I_0 \sum_{k=1}^{N} R_k \tag{5.18}$$

It is clear equation 5.18 has the linearity property if we understand that the reflection coefficient of the residuals is the difference of the reflection coefficients of the synthetic and acquired data, and the approximation of equation 5.8 is valid.

In practice, we do not update the model by using equation 5.18. Instead we compute a more precise impedance inversion by the BLIMP method and then removing the low frequency's linear trend (slope) and the intercept (water velocity in our marine simulation), resulting in the velocity perturbation over the Earth's global trend.

5.2 Simulation

5.2.1 Pre-stack depth migration based gradient

Equations 5.9 and 5.7 reduced abruptly the computer power required for the inversion. From now on, all the tests were done on a personal gaming laptop (ASUS Intel Core i7-4700HQ 2.40GHz, 4 cores, 16Gb of Ram memory) in Octave (free scripting software) on a Linux terminal and no parallel processing.

Figure 5.1 shows the inversion based on equation 5.9, starting from the model on the top left of figure 4.3, and with the computer specification cited previously. The result is impressive, and the process stopped after 29 iterations in about 36 hours of run time with no parallel processing (it was repeated using the 4 cores of the computer for parallel processing and elapsed time decreased to 18 hours). The method worked well in the whole model, but

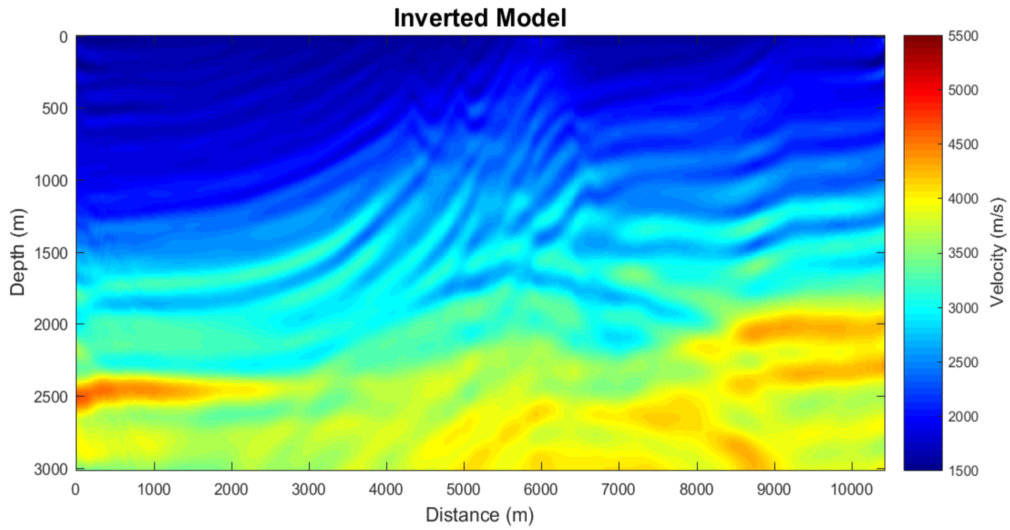


Figure 5.1: Inverted model using the initial model on the top left of figure 4.3 and based on equation 5.9. The algorithm ran on a personal laptop, and no parallel processing in about 36 hours.

was better at shallower areas. Major structures are successfully inverted. Even the fault zones in the center of the model can be interpreted. Using no forward modeling to compute the gradient shows to be a valid approximation. BLIMP provides us an impedance model which includes the perturbation (high frequency) of the impedance to the low frequency model. The difference of the perturbations of current and previous iterations (the proposed gradient) points to the correct minimum, but it is not optimized to minimize the objective function of equation 2.4. The step length (that requires 2 forward modeling per iteration to be estimated) works as the minimization operator. The conjugate gradient was not used (we tested with and without it and the ending models were equivalent).

Objective function and model deviation plots are on figure 5.2. We ran a total of 29 iterations (the code stops when a maximum frequency, selected by the processor, is reached, and it changes the frequency content of the current iteration when the errors converge or start to diverge). The inversion process has 2 major decrease in errors: one inverting up to 6Hz and another up to 10Hz. We believe those are the dominant frequencies related to the wavelength as same "size" of layers in the model. It is interesting to note that even with the objective function keeps decreasing by iteration, the model deviation can diverge

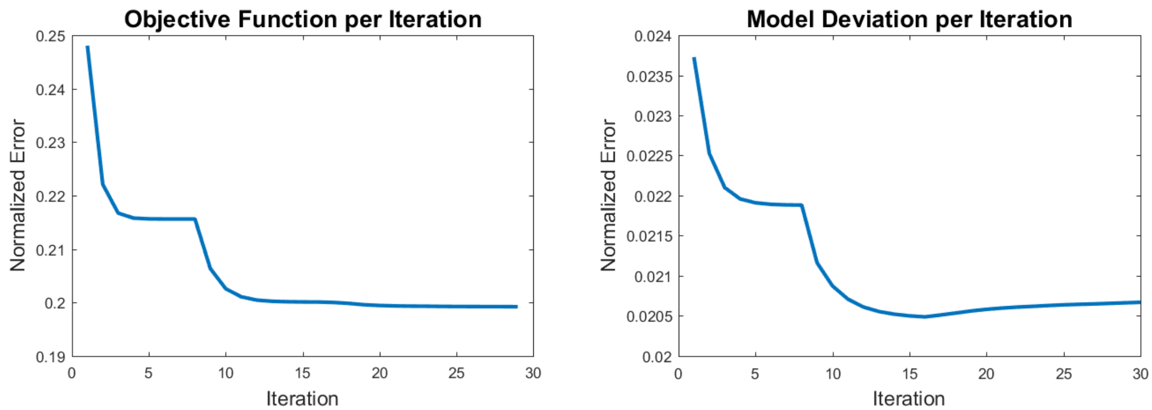


Figure 5.2: Plots of the objective function per iteration (left) and model deviation per iteration (right). There was a total of 23 updates in the model but the convergence is reached with only 12.

at some point (close to iteration 17). The meaning is that the process is trapped on a local minimum of the objective function. With all these information in mind, we can say the method is stable and leads to a optimized model that gets closer to the global minimum than the initial model.

5.3 Conclusions

In chapter 4 I implemented an impedance inversion of the gradient. This lead me to understand the FWI routine as a combination of seismic processing tools, most specifically a pre-stack depth migration, stacking and impedance inversion, which I also called seismic processing operators. By assuming linearity those operators, I ended up with a solution on which I can compute the gradient without any forward modeling (only two forward modeling to estimate the step length). This resulted on a huge reduction is cost, as the main computation power in the FWI routine goes to generate synthetic shots. The inverted model has high accuracy and resolution, and is comparable with previous inversions.

Chapter 6

The forward modeling-free gradient: a post-stack approximation

In this chapter, it is presented the post-stack approximation as an extension of the forward modeling-free gradient method introduced in chapter 5, on which i apply a zero-offset PSPI migration the NMO corrected and stacked CMP gathers. This new approximation is possible when I started to look to the FWI as a seismic processing flow. Its inverted model has lower resolution and is strongly dependent of the initial model, but it has reduced computational cost.

6.1 Theory

6.1.1 Post-stack depth migration-based gradient

Looking back again at equation 5.9, I understand the gradient is estimated by applying a few seismic processing tools. The two first processes consist on apply a pre-stack depth migration in the acquired data and then stack the outputs to create a subsurface image of the Earth. And this is the same goal of a post-stack depth migration, to estimate such image. As both the pre-stack depth migration and the post-stack depth migration are trying to deliver the same kind of output (of course, with different mathematical approximations and different resolution), I am commuting the migration and stacking operators M and I, respectively. This is a philosophical statement based on my interpretation of what do the migration outputs are: if the outputs have the same meaning (a subsurface image), than both migrations (pre and post-stack) are a reasonable approximation for the FWI gradient (in this case, for P-wave velocity only). Pre-stack and post-stack depth migrations have equivalent outputs in an area where the stratigraphy contains only horizontal layers. In a more complex model, I am assuming it is approximately true enough so we can apply the same commutation (equation 6.1).

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n (I \{M [S (\mathbf{d}_0)]\} - \mathbf{m}_n) \tag{6.1}$$

This means I can input in the algorithm a stacked section (figure 6.1) and apply a post-stack depth migration algorithm (I used a zero-offset based PSPI with a cross-correlation imaging condition). This allows the FWI method to be applied on non-usual data, such as 1D surveys (those are rare, and mostly academic surveys). However, at this point, I am dealing with zero-offset data, with no AVO information, and it can be used only to invert impedance.

6.2 Simulation

6.2.1 Post-stack depth migration-based gradient with BLIMP

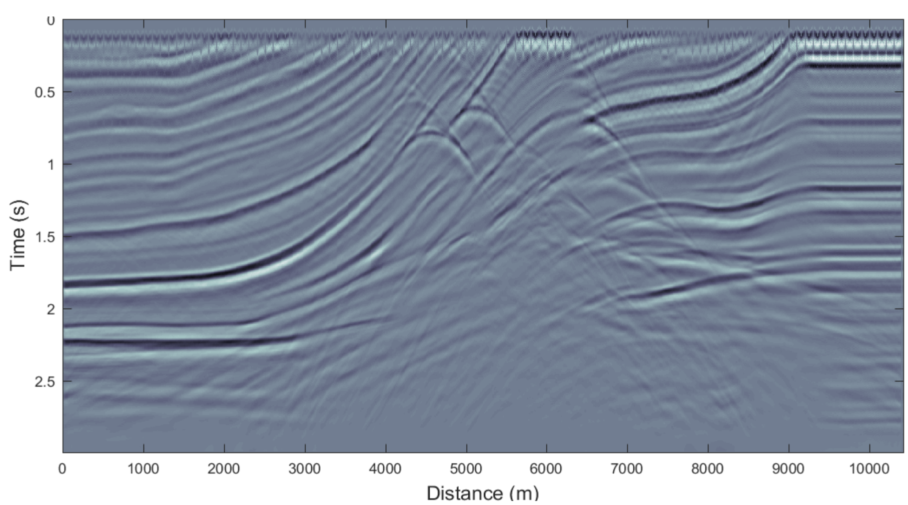


Figure 6.1: Stacked section used as input in the post-stack method.

The input data for the post-stack method is the data stacked where the stacking velocity used is the same initial model of the others tests. The difference is that it was converted to a RMS velocity. Analyzing the stacked section on figure 6.1, I noticed the presence of diffractions on the center area of the model caused by the fault zones. We can also observe a strong horizontal event close to the surface and some internal multiples are very evident in the deeper areas. I took note of all those observations and studied how they affected the inversion.

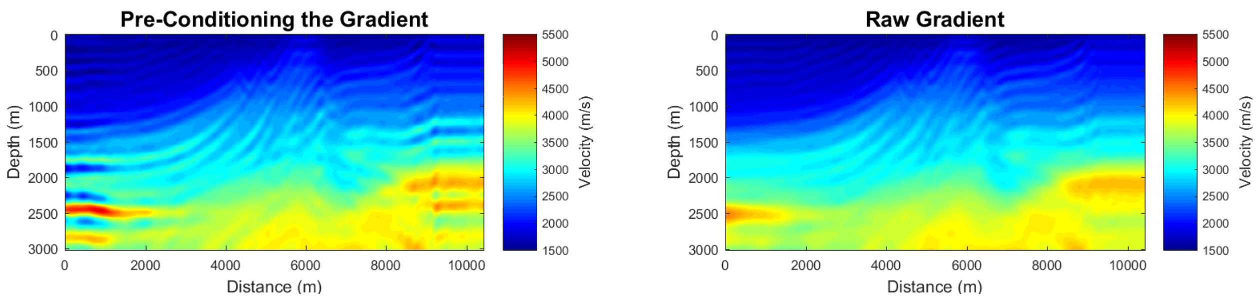


Figure 6.2: Inverted models using: pre-conditioned gradient (left) and the "raw" one (right).

I ran two different preliminarily tests. On the first I pre-conditioned the gradient by convolving it with a 2D Gaussian window. This step was neglected during the second test, where I only scaled the "raw" gradient. In both I used the starting model of the top left of figure 4.3 and results are on figure 6.2. The left model is the resulted inversion when the gradient is smoothed and on the right the gradient was kept raw. In both cases information is included to the starting model, but with reduced resolution when compared to previous inversions. Inversion got truncated by the horizontal event close to the surface that I observed on figure 6.1. Pre-conditioning the gradient did a better job when compared with the raw one. However, it suffers with borders effects. Comparing the objective function (figure 6.3 left) I can say that both are stable with the smoothed gradient (blue line) doing a better job. The borders effects are not notice in the plot as the objective function is computed on a control shot in the center of the survey. Borders effects are more pronounceable on the model deviation plot (figure 6.3 right) where the smoothed gradient initially reduces the model difference and, from iteration 10, it starts to diverge.

This method showed to be even cheaper than the pre-stack one, and the inversion im-

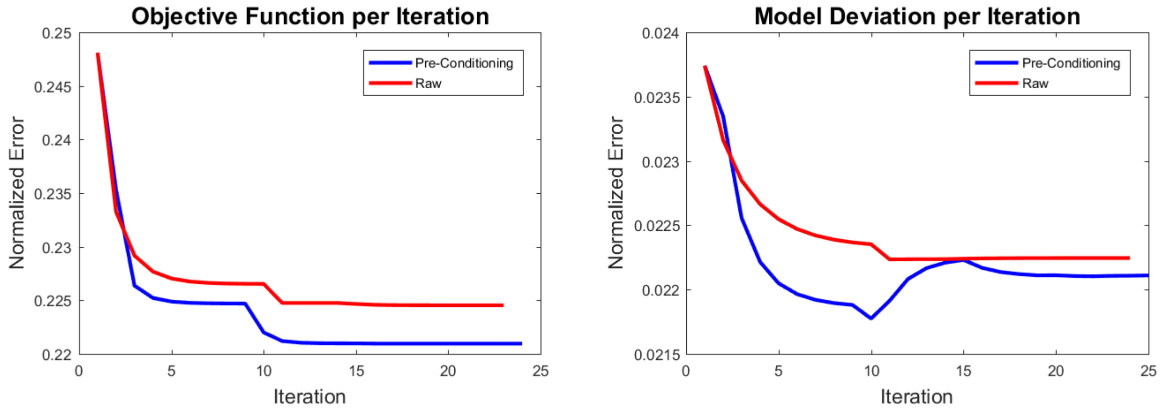


Figure 6.3: Plots of the objective function per iteration (left) and model deviation per iteration (right) with the "raw" gradient (red) and pre-conditioned one (blue).

proved the initial model, retrieving major structures, when compared with the true model.

6.2.2 Post-stack depth migration-based gradient with trace integration

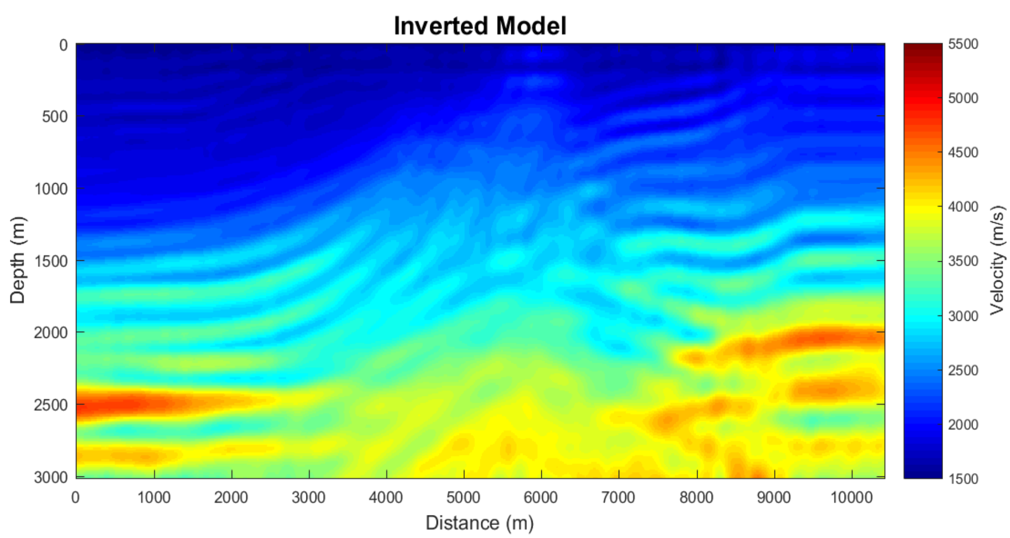


Figure 6.4: Inverted model with the simplified post-stack method. The border effects were suppressed, but with some loss in resolution at the central portion of the model.

In previous chapters and analysis, I have been using the BLIMP to transform the reflection coefficients of the migration output to impedance. However, I was using only the update impedance, and not the whole BLIMP output (the impedance inversion of the migrated trace combined with the initial model). This leads us to think that a simple exponentiation of the trace integration in depth is enough to do such transformation. This also avoid some time-to-depth and depth-to-time conversions, whose slower the method.

Figure 6.4 shows the inversion result. The resolution is improved where the model is simpler (outside the center part). However, when I apply a simple trace integration as the equivalent of the impedance inversion, I ended up with some vertical stripes on the model. To attenuate this effect, I apply a horizontal smooth on the gradient with one side effect: loss in resolution on complex (lateral variation) areas of the model, like the central portion, where the faults are not so clear anymore.

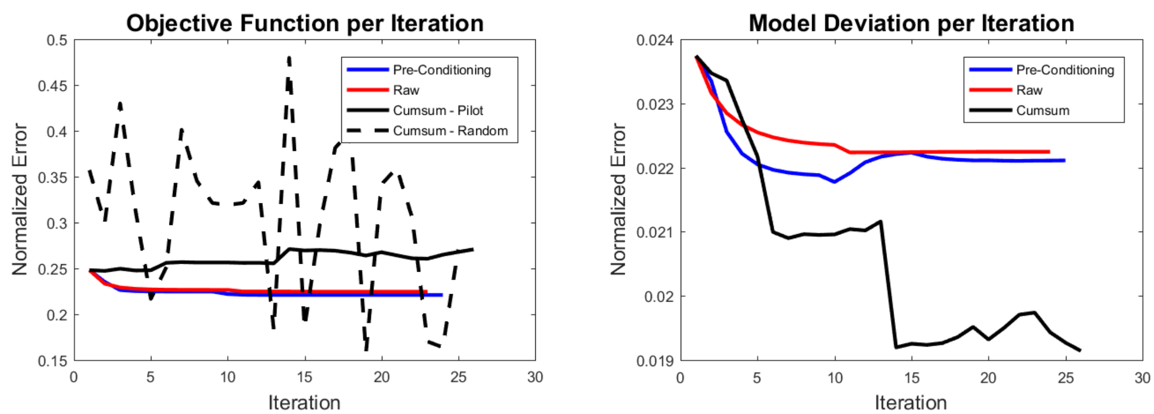


Figure 6.5: Plots of the objective function per iteration (left) and model deviation per iteration (right) with the "raw" gradient (red) and pre-conditioned one (blue) compared with the trace integration ones (black). the dashed black line is the objective function on a random shot position.

Another difference on the method is on the choice of the shot position to estimate the step length. For the previous analysis, I was selecting the shot in the middle of the survey as a fixed point in space. It was biasing the updated model be focused only on the central part, "forgetting" to optimize the sides. For the inversion showed on figure 6.4, at each iteration, a random shot position is selected to estimate the step length, so the update is not biased. On the other side, I lost track of the objective function at each iteration, as shown on figure 6.5. It compares the trace integration method (black lines) with the BLIMP ones (red and blue lines). The dashed black line is the objective function of a random shot position. The objective function ends up to be random, with a trend to decrease. The black line is the objective function of the shot in the middle. As I lose resolution in the central portion of the model, the objective function diverges. Looking to the model deviation, the trace integration provides us a model closer to the real Marmousi.

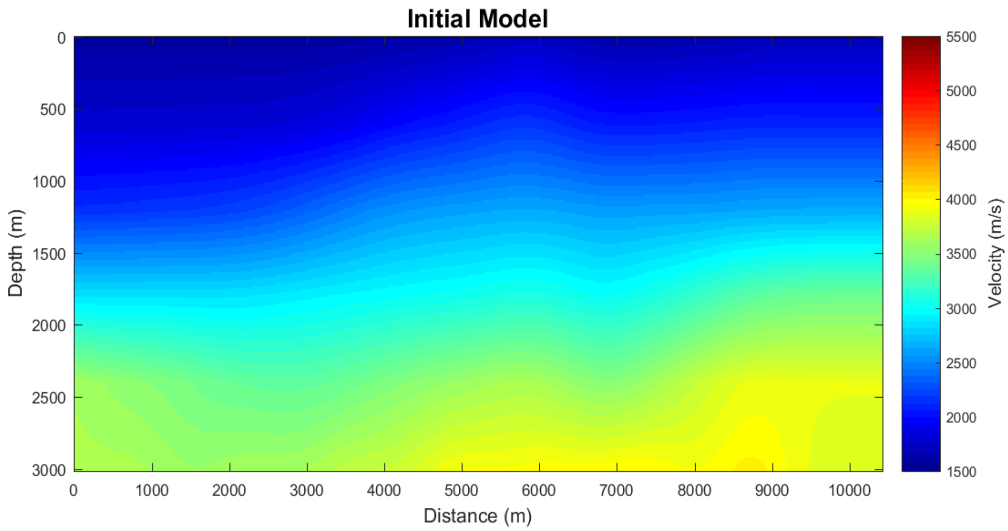


Figure 6.6: A velocity analysis simulation used as initial model.

The post-stack FWI shows to be stable, fast and simple the initial model is a smoothed version of the Marmousi. For the next test, I will use a simulation of a velocity analysis as initial model to understand better the limits of the method (figure 6.6). I will follow the same procedure as before, increasing the maximum frequency of the range by $1Hz$ at each iteration, and selecting a random shot position for the step length.

The inverted model is presented on figure 6.7 and it has reasonable resolution, remembering that I am using a post-stack approximation. Geological features were better "recovered" at the shallow portion of the model, following the same trends as the real Marmousi. The bottom part of the model is dominated by multiples of the high velocity bodies and the central part is still unclear. I started to believe that this inverted model can be used as that starting point for a second pass of the routine, and it resulted on the inversion of figure 6.8. The resolution is improved and some parts in the deep started to gain more clarity, mostly

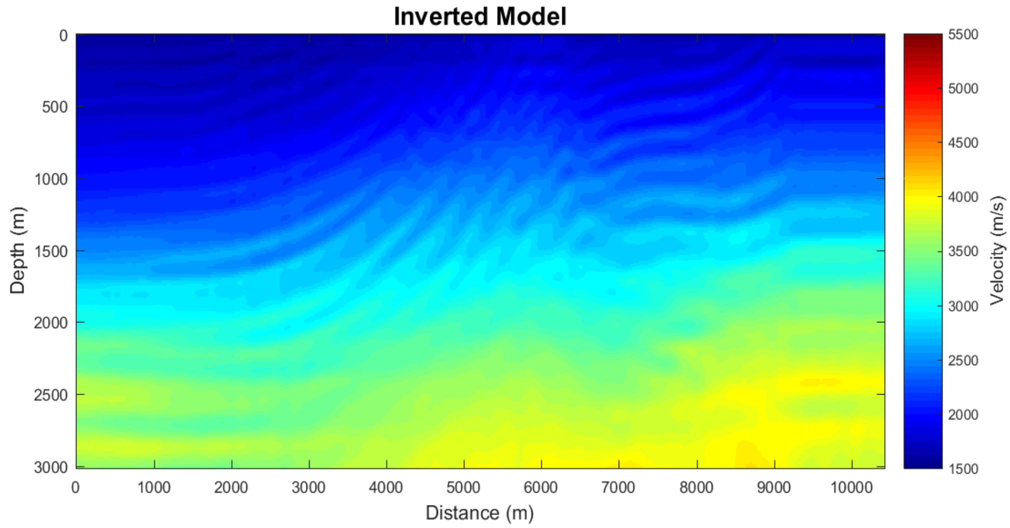


Figure 6.7: Inverted model os the simplified post-stack FWI using figure 6.6 as initial model.

the shapes of the high velocity bodies on the sides of the model. Multiples are stronger after second pass.

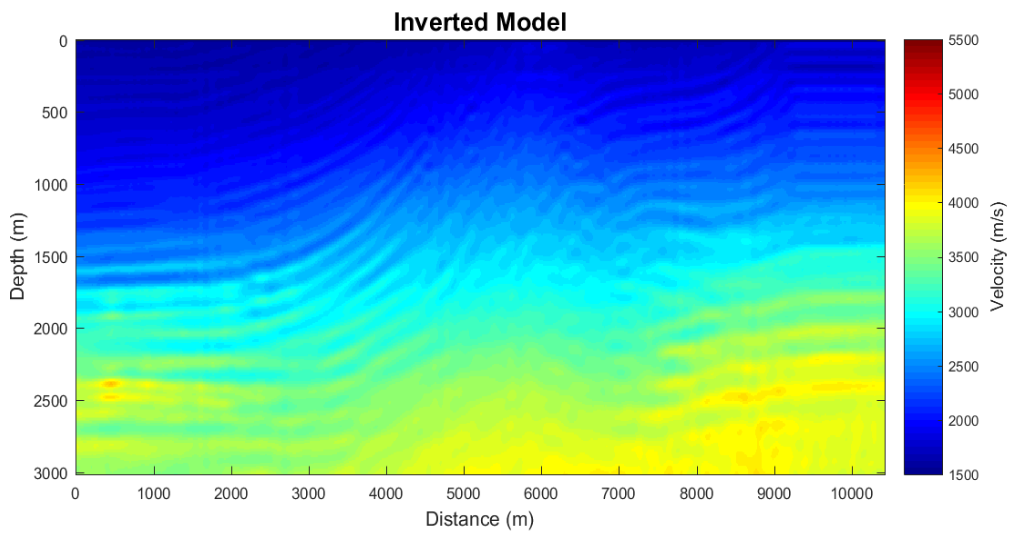


Figure 6.8: Second pass of the post-stack FWI, increasing the resolution of the inverted model.

The model deviation on figure 6.9 shows the behavior of the updates during the first (left of vertical red line) and second (right of vertical red line) passes. It is clear how the inverted model gets closer to the real Marmousi after two passes of the routine, resulting on a very reasonable model, considering its cost. In next chapter I will present a new improvement

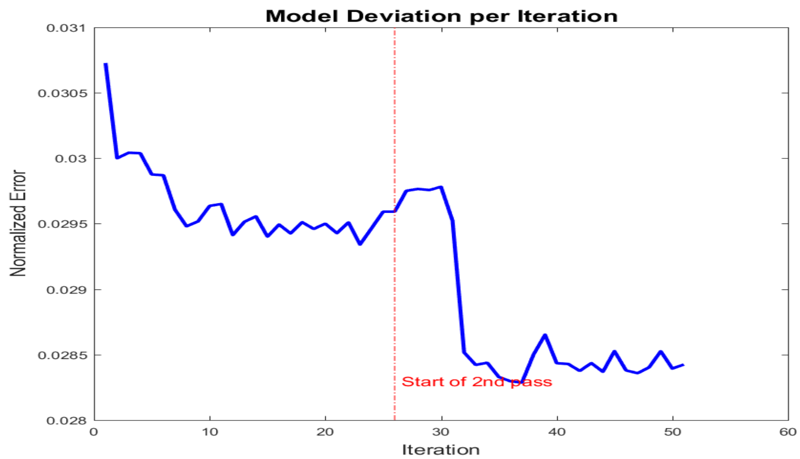


Figure 6.9: Model deviation of the first and second passes of the post-stack FWI using the figure 6.6 as initial model.

that showed to be even cheaper, but leading to more accurate inversions.

6.3 Conclusions

By interpreting that the gradient in the impedance inversion of a subsurface image (migrated gathers), I proposed that the pre-stack depth migration can be replaced by a post-stack depth migration. The inverted model shows the recovering of low-mid frequency content, related to thicker layers. In other words, I could recover the general background structures of the Marmousi model, but with limited resolution, and it can only invert P-wave velocity. This approximation showed to be strongly dependent of the initial model, but it has reduced cost and can be an useful tool to estimate a initial model closer to the global minimum, to be the input of a pre-stack FWI.

Chapter 7

The forward modeling-free gradient: well calibration

In this chapter I am proposing to replace the step length estimation of Pica et al. (1990) for a matching filter (Margrave et al., 2010, 2011; Romahn and Innanen, 2016) estimated by minimizing the difference between a sonic log to the gradient at the same spatial location. This leads to a 100% forward modeling-free FWI approximation. By knowing the true answer at a local area, the gradient is calibrated and the updated model is accurate. The inverted model for the pre-stack approximation has a resolution comparable with the one obtained with the BLIMP gradient in chapter 4, when the same initial model is used. The post-stack approximation is improved when compared with the previous chapter, and can provide an initial model for the pre-stack FWI that is closer to the global minimum.

7.1 Theory

7.1.1 Well calibration

Until now I have used Pica et al. (1990)'s approximation to determine the step length and to calibrate the gradient. However, it goes against our seeking for a forward modeling-free routine, as I still require two forward modeling to estimate the step length. So, I still require a modeling code with reasonable physics embedded and some reasonable source estimation. Margrave et al. (2010) and Romahn and Innanen (2016) calibrate the gradient by matching it with a sonic log at the same spatial location. They estimate a scalar and phase rotation to create a match filter to convolve with the gradient. This would lead us with a FWI algorithm 100% forward modeling-free. So I decided to implement it.

To find the scalar α that minimizes the impedance inverted gradient trace S_{grad} to the impedance (sonic log) well trace S_{well}, I minimize the L2-norm function for α:

$$\Phi = ||S_{well} - \alpha S_{grad}||^2 \tag{7.1}$$

This leads to a simple equation to estimate α:

$$\alpha = \frac{S_{well}^T S_{grad}}{S_{grad}^T S_{grad}} \tag{7.2}$$

For the phase rotation, I use the $Toolbox$ code $constphase.m$ that finds the angle ϕ that makes S_{grad} looks like S_{well}. By finding α and ϕ, I create a matching filter which is convolved with the gradient to have the model update.

Combining the well tie calibration with the forward modeling-free gradient from chapters 5 and 6, I are proposing a new FWI routine that is 100% forward modeling-free. This reduces significantly the cost of the inversion without needing source estimation.

7.2 Simulation

7.2.1 Post-stack approximation with well calibration

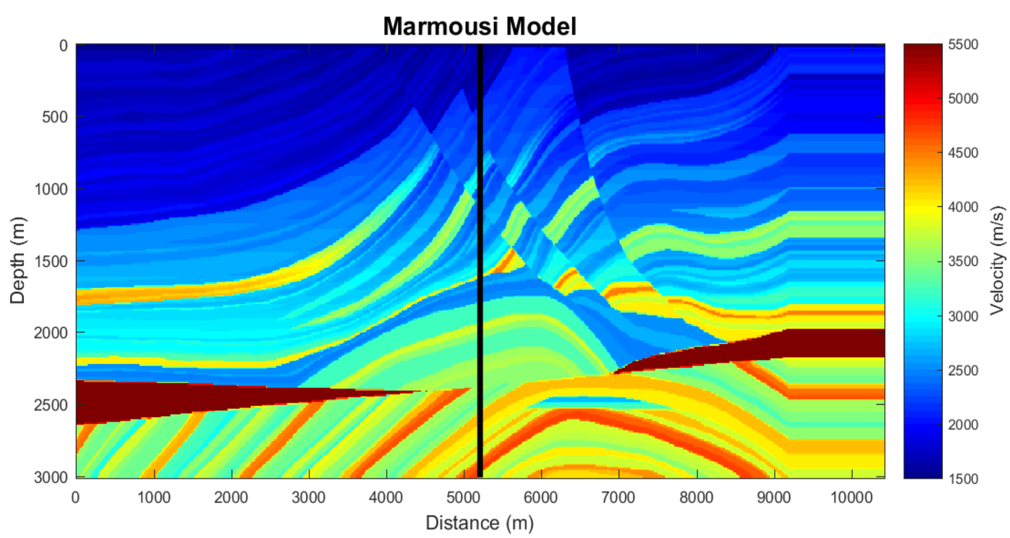


Figure 7.1: True Marmousi model. The black line is the well position.

Test are held in the Marmousi model. The well for calibration is selected to be the column in the center of the model, as shown on figure 7.1.

For the starting model on figure 7.2, I decided to use the same starting model as on figure 6.6. It is a simulation of a velocity analysis output. Our goal is to check if the well calibration can increase the resolution of the inverted model if compared with the one in figure 6.8.

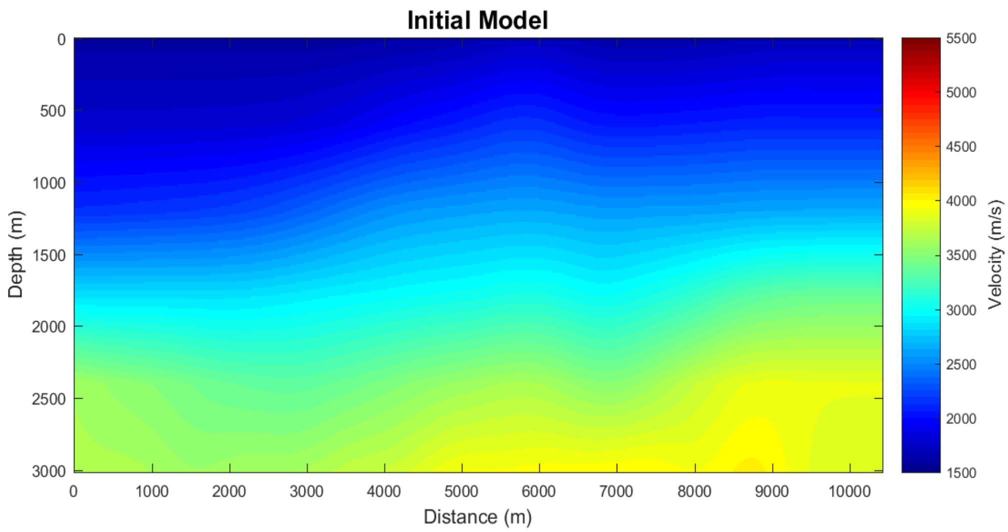


Figure 7.2: The initial model is a simulation of a velocity analysis output.

The main difference of the routine now is to replace the Pica et al. (1990)'s algorithm to estimate the step length by the well calibration. At each iteration I compute the gradient according to equation 6.1 and match its central column with the residual well log (the difference of the well and current model). Computes a scale factor and phase rotation to generate a match filter that, by convolving with the gradient, results on a model updated that is optimized in the well location. The frequency content per iteration is a little different than before. Now, for the first iteration, the frequency range to estimate the gradient is from $4Hz$ to $9Hz$. At each iteration, the frequency range is increased by $1Hz$ until the maximum of $30Hz$.

After running 25 iterations in 8 minutes on a personal laptop, the inverted model of

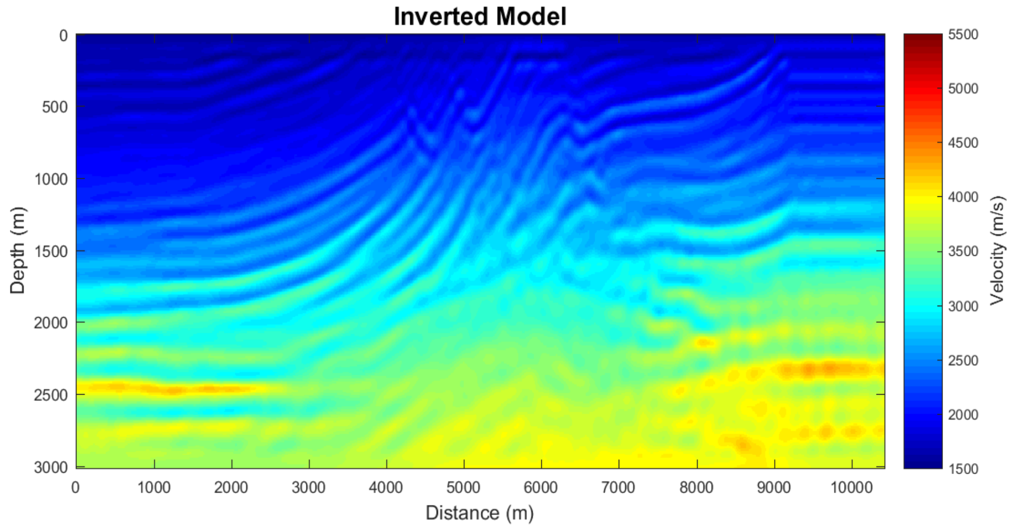


Figure 7.3: Inverted model for the post-stack method with well calibration. It has higher resolution than the inversion of figure 6.8 with lower cost.

figure 7.3 shows impressive resolution (higher resolution than I was expecting) with low cost. All main structures are recovered and even in the deep area of the model, some layers started to show their shape. The three large faults in the central part are also visible and can be used by interpreters. Multiples of the high velocity bodies are present below $2500m$ depth. However, the most impressive result is the reduced cost to run the routine. As cited previously, it took only 8 minutes to be completed. The band-limited impedance inversion based gradient of chapter 4 took about 48 hours to complete and it required 48 nodes of the $CREWES$ supercomputer. Comparing both ending models, the cost reduction came with some loss in resolution, but models are still comparable.

As any synthetic shot is generated during the process, the objective function is not measured at each iteration. But, as I have the true model, I computed the model deviation per iteration in figure 7.4 and it is clear that the inversion is stable. The method converges fast, in about 2 iterations. After that there is a small increment in error that is slowly reduced as iterations advance.

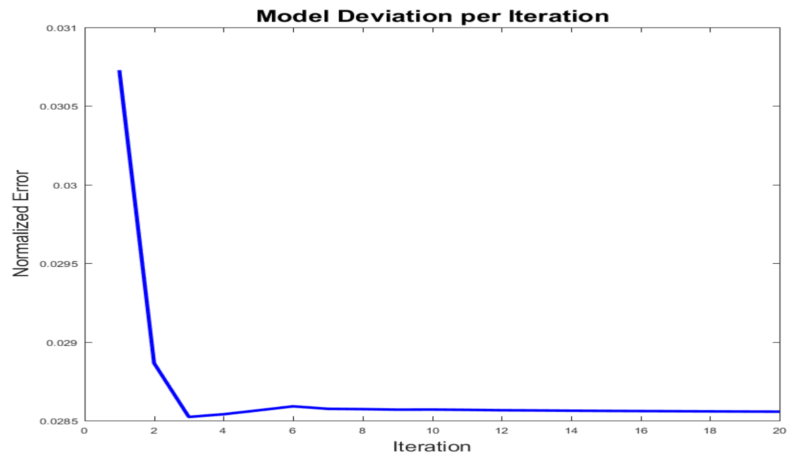


Figure 7.4: Model deviation of the post-stack approximation with well calibration. The convergence is fast, reached on the very firsts iterations.

7.2.2 Pre-stack approximation with well calibration

In the previous section, the post-stack method is combined with a well calibration of the gradient, providing a fast solution, with surprising high resolution. Now, I am testing the same method, but with a pre-stack migration. I am losing in cost, but I am, now, seeking improvements in quality.

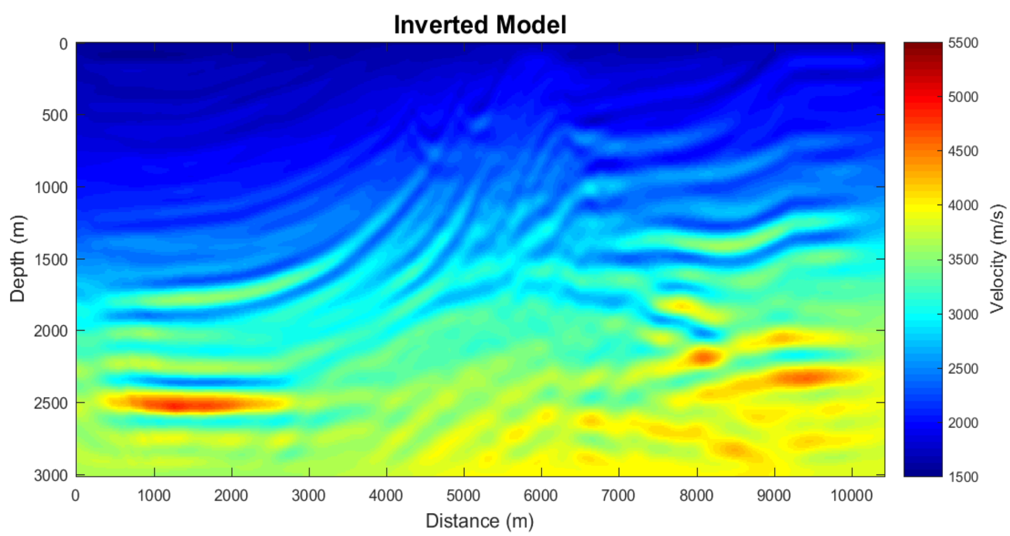


Figure 7.5: Inverted model for the post-stack method with well calibration.

By using the same starting model of figure 7.2, the resulted inversion of figure 7.5 has incredible resolution, improving the resolution of the post-stack method (figure 7.3). The multiples below 2500m are still visible, but the structures in the central part of the model are more clear. The high velocity bodies are better placed with closer true velocity. Only the shallow portion looks to be blurred, when compared to the post-stack method.

The model deviation of figure 7.6 shows a fast convergence. Basically, the best fit is reached at the very first iteration. After that the deviation just flies around the minimum.

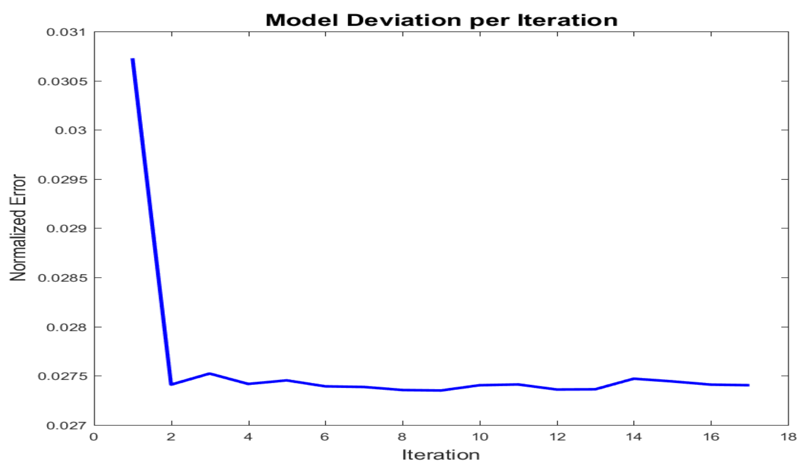


Figure 7.6: Model deviation of the pre-stack approximation with well calibration. The inverted model goes closer to the true model.

This model came with some loss in cost, as it took about 6 hours to be completed using a personal laptop. As surveys get larger, with more shot points and recording time, running time will increase as well. So, all is a matter of cost/quality trade-off.

7.2.3 Velocity models as migration velocities

It is clear as by applying the FWI routine, the starting model is improved and features of the true model are included to the final inversion. This is very helpful for seismic interpretation. Now, let's remember that, usually, the starting model of the FWI routine is the best migration velocity from the seismic processing. So, I am wondering if the FWI can also improve the subsurface image by providing a better migration velocity. For this analysis, 3 velocities are chosen to migrate the acquired data: the velocity analysis initial model (figure 7.7a), the post-stack forward modeling-free with well calibration (figure 7.7c), and the pre-stack forward modeling-free with well calibration (figure 7.7e). For the pre-stack migration, I use the PSPI migration, with a deconvolution imaging condition, on the acquired data. As the data does not contain higher frequencies, I migrate the data on the frequency band from $1Hz$ to $60Hz$. The migrated shots are muted and stacked to form a subsurface image (figures 7.7b, 7.7d and 7.7f).

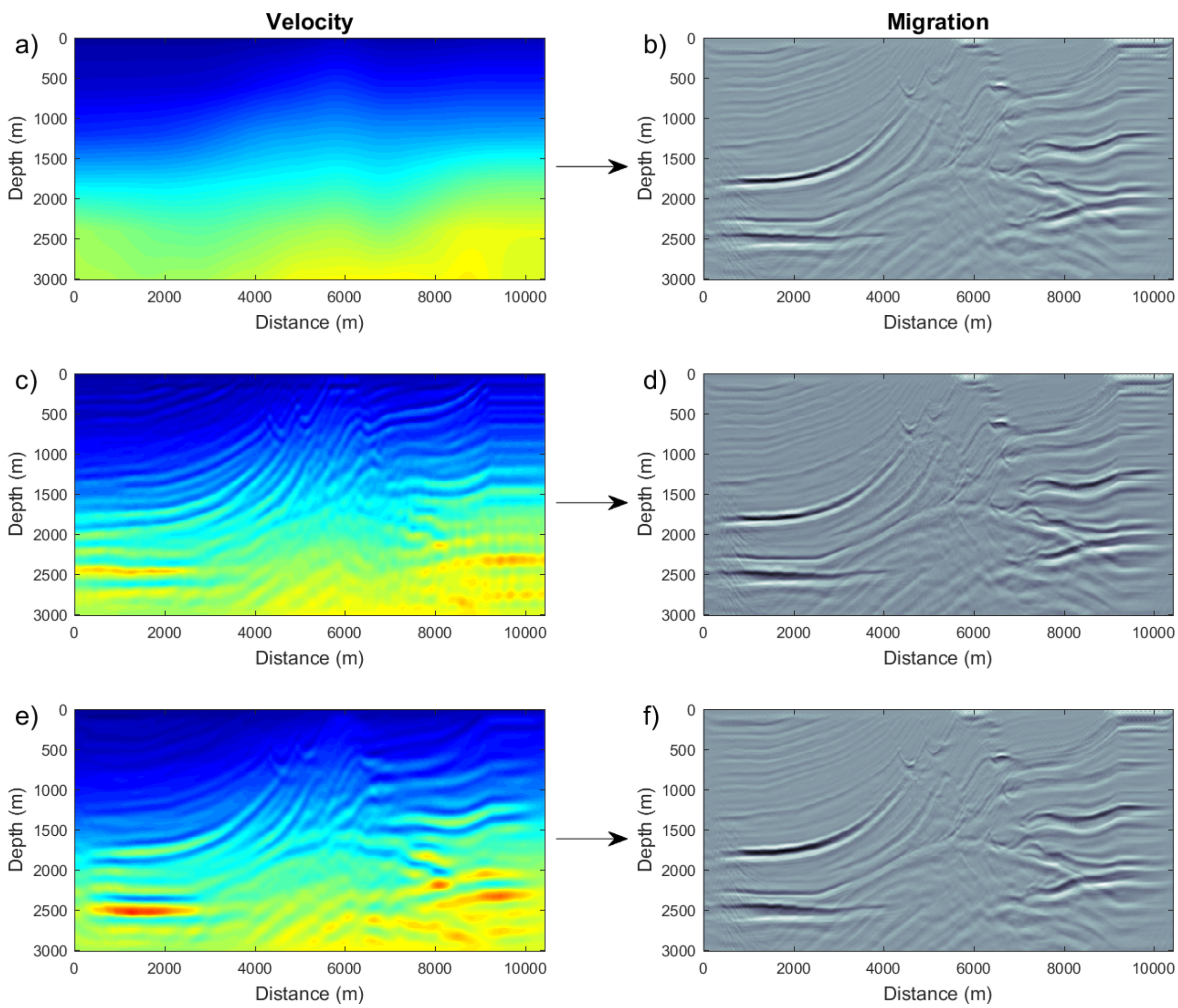


Figure 7.7: Using the models a) initial, c) post-stack and e) pre-stack to apply a pre-stack migration of the acquired data and obtain the respective subsurface images b), d) and f). Migration is improved as the model is more precise.

Looking at all the migrated images (figures 7.7b, 7.7d and 7.7f), I can say that all the migration velocities (figures 7.7a, 7.7c and 7.7e) are very similar. Now, looking closely, as we go deeper in the model, structures have different positions. It is more evident at the high velocity body in the left size of the model on the depth of $2500m$. The post-stack FWI started to place the structure in a more correct place, as the pre-stack FWI did even a better job. So, the higher resolution the model has, higher the resolution the subsurface image have. And it is more evident as deeper as we go.

I am confident that I am proposing a cheap and fast FWI that will improve the initial model to help in the seismic interpretation of the survey and to provide a clearer subsurface image by providing a more accurate migration velocity.

7.3 Conclusions

A 100% forward modeling-free FWI is possible by combining the forward modeling-free gradient approximation of chapters 5 and 6 with a well calibration (Margrave et al., 2010, 2011; Romahn and Innanen, 2016). Due to a calibration of the gradient with a local true answer (a sonic log), the updated model has higher accuracy and resolution than the previous forward modeling-free results. And it is valid for both pre and post-stack approximations. By using the inverted models as migration velocity, the subsurface image seems to be slightly improved as the inverted model is more accurate.

Chapter 8

Conclusions

Results of tests and new interpretations of the FWI steepest-descent algorithm were presented following the chronological development during the Ph.D. program. Now I will show my conclusions and analysis of the cost for each approximation used.

The standard steepest-descent (gradient) method was tested on chapter 2 on a 2D synthetic dataset with great resolution of the inverted models. It showed to be stable on a simple 2D model when a good initial model is used, and promising when the initial model is derived from sonic logs located on different locations of the correct model. The gradient is computed by a PSPI migration of the residuals with a deconvolution imaging condition (that works as a diagonal Hessian approximation). The gradient is smoothed by convolving it with a 2D Gaussian window, attenuating some artifacts, and avoiding over-fitting. A mute must be applied and selected in a way to remove far offset artifacts, but preserving important primaries information. The step length is estimated by a trial and error method that is precise but costly. This initial algorithm was applied in the 2D Marmousi acoustic model reaching convergence and improving the model until certain iteration. It was a good example of an optimization method that got trapped on a local minimum, where the errors of shots decreased at each iteration but the model deviation started to increase after a few iterations decreasing. It suggests that the standard steepest-descent method is stable but requires some modifications as the geologic complexity is higher. Actually, when analyzing the method costs (figure 8.1), the total computational cost can be divided mainly into three portions: the forward modeling (on the residual step), the migration (on the gradient step), and the step length estimation (line search). A forward modeling process, even on an acoustic case, is costly depending on the algorithm used, and it was used a finite difference one

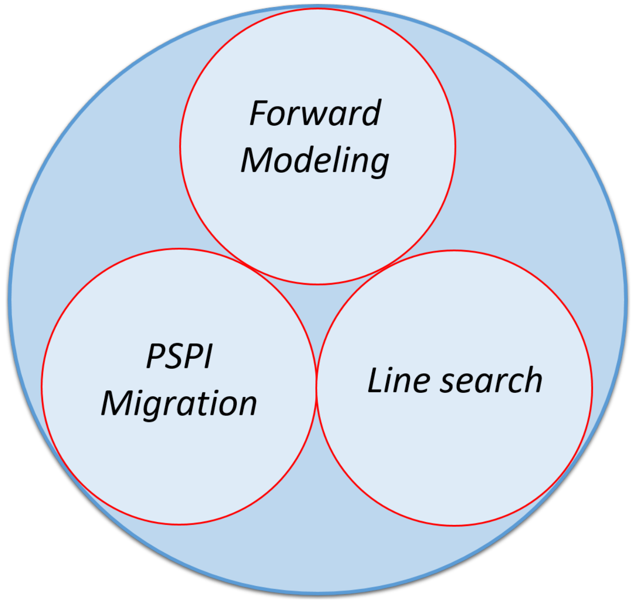


Figure 8.1: Cost related to a regular steepest-descent FWI. The most expensive processes are the forward modeling to compute the residuals, then the PSPI migration to estimate the gradient, and the line search, that requires a few forward modeling calculations to compute the step length (chapter 2).

(expensive but with better response). Remembering that a forward modeling is required for each shot position on every iteration. Pre-stack migration can be highly expensive. PSPI is a one-way phase shift back-propagation method, cheap when compared with others migrations, and with great resolution. The cost for the step length estimation on a trial and error technique is due to the number of forward modelings required for it. Higher precision is reached as more values are tested.

Our first improvement came as a tentative to simulate some level of deconvolution (remove the source), during the gradient estimation, by selecting a frequency band on one iteration and migrating (PSPI) each frequency separately (chapter 3), ending up with several gradients. Thanks to the implementation of a cheaper algorithm to estimate the step length (Pica et al., 1990), the final gradient is a weighted average of all the monochromatic gradients, where the weights are each one of the step lengths. I call this approximation *monochromatic averaged gradient*. I could show that this new approximation results on higher resolution

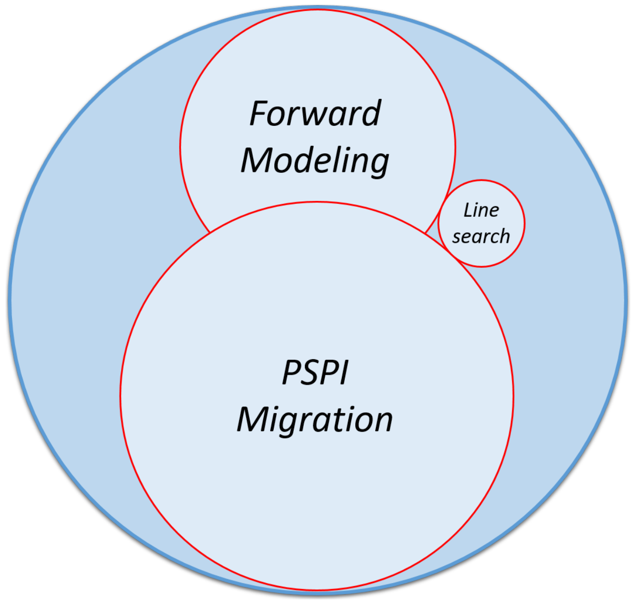


Figure 8.2: For the monochromatic averaged gradient (chapter 3), the cost is reduced for one single forward modeling to compute the step length by using the Pica et al. (1990) approximation, but the migration cost is increased as each frequency is migrated separately, becoming even more costly on higher iterations.

inverted models, which gets better if the initial model is closer to the global minimum, when tested on a 2D Marmousi acoustic seismic simulation. The convergence is even closer to the global minimum if a conjugate gradient algorithm is used, showing to be a great improvement with a very low cost. Figure 8.2 is a representation on how the costs change with the monochromatic averaged gradient. The forward modeling's cost for the residuals calculation kept the same as the one on figure 8.1. Gradient estimation has a higher cost as each frequency is migrated separately. Line search's (step length) cost is decreased for a single forward modeling, but it required to be estimated for each frequency. However, it will only be more expensive than the trial and error when the inversion is working on higher frequencies, and I showed that the FWI is more efficient on lower frequencies. An initial try for an impedance inversion of the gradient (until this point, I were assuming that the migration output - reflection coefficients - was equivalent to an impedance update when multiplied by the step length) was test by a simple trace integration, and I concluded that a better approximation was necessary.

In chapter 4 the impedance inversion of the gradient, prior to the step length estimation, became the focus of the analysis. As I took to our attention before, I was using migration's output (reflection coefficients) as an unscaled impedance update. On each iteration only a selected frequency band is used, this leaded us to decide for a *band-limited impedance inversion* (BLIMP) of the gradient, where the impedance inversion is optimized by using the low frequency content present on the initial model. This means that the initial model must contain, at least, the linear trend of the correct model. I concluded that it was a successful improvement, as it is stable, converges fast, and is less costly than the monochromatic averaged gradient approximation. Figure 8.3 shows the new cost scheme for the BLIMP based

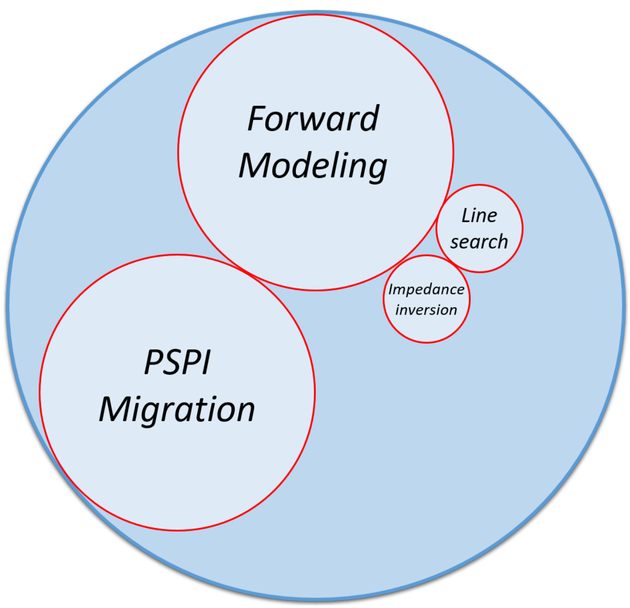


Figure 8.3: With the implementation of the BLIMP to better estimate the gradient (chapter 4), the migration cost went back to its initial standards (chapter 2). However, a small cost is added to apply the impedance inversion of the reflection coefficients based gradient.

gradient. The forward modeling cost to compute the residuals is still the same. Migration cost is reduced back to the same level as the standard steepest-descent method (all the frequencies of the band are migrated at the same time). An impedance inversion (BLIMP) cost is included in the scheme, as it requires depth to time and time to depth conversions.

The step length estimation is reduced to a single forward modeling for each iteration. In others words, I could reduce significantly the routine costs by implementing an impedance inversion of the gradient, and ending up with similar resolution as the inverted model using the monochromatic averaged gradient method.

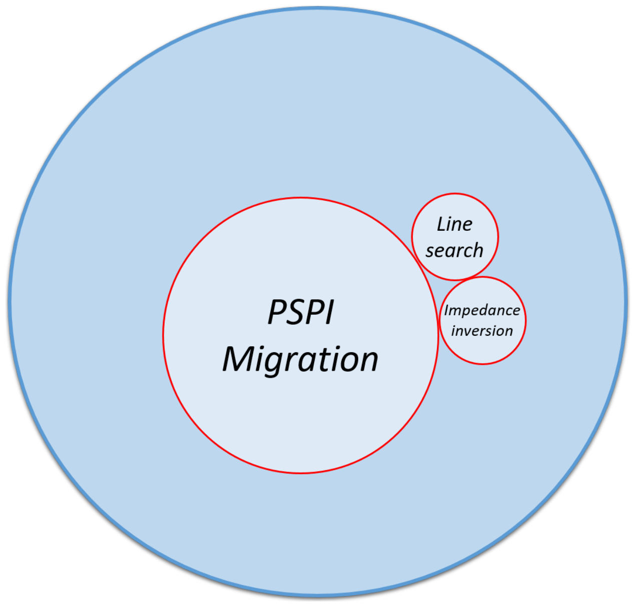


Figure 8.4: Approximating the gradient as the residual difference of the current model, and the impedance inversion of the acquired data migrated using the current model (chapter 5, subsection 5.2.1), reduced the cost of the FWI method to, basically, the cost of the pre-stack depth migration (PSPI), as no forward modeling is required to compute the gradient.

Perhaps the most important development is shown in chapters 5 and 6. There I show that the gradient can be simplified by the difference between current model and the impedance inversion of the acquired data migrated using the current model. This solution came when I started to understand the FWI routine as a combination of seismic processing tools, and assumed linearity properties of them. This way, no forward modeling and source estimation are required to compute the gradient. It was demonstrated for P-wave velocity only, but I am confident that the same strategy can be expanded for any invertible parameter. The inverted model is somewhat comparable with previous inversions, but with a great difference in cost. With this new approximation, the main cost of the process is due to the PSPI migration (figure 8.4). Each iteration has basically the same cost of a pre-stack depth migration of the whole survey. It is possible to make the process even cheaper when I commute the order of the stacking and migration processes during the gradient estimation. This way, the gradient

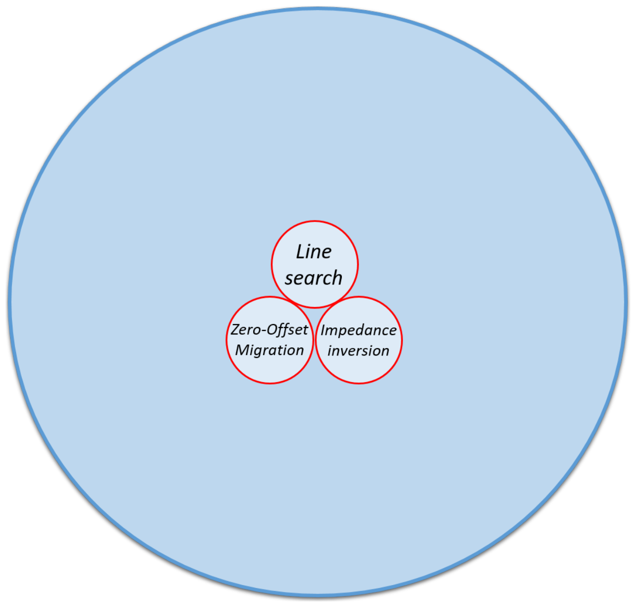


Figure 8.5: The post-stack migration based gradient (chapter 6 reduces significantly the total cost of the FWI method.

is computed by a post-stack depth migration (a zero-offset PSPI was used). I show on figure 8.5 that the total cost of the inversion is abruptly reduced. For the 2D Marmousi acoustic simulation, I could run the whole inversion on a tablet. However, some trade-off between cost and resolution occurs, and must be taken in consideration.

In chapter 7 the gradient started to be calibrated by a well sonic log instead of the Pica et al. (1990)'s algorithm. This allowed the process to be 100% forward modeling-free, fast, cheap, and with increased resolution. Figure 8.6 represents the costs when the post-stack forward modeling-free gradient method is combined with the well calibration. Costs are reduced, source does not need to be estimated, and quality is improved. The process took 8 minutes to run in the Marmousi model on a personal laptop. This means that, if the sonic log is available, anyone or any company can have a cheap and fast FWI routine, and have a

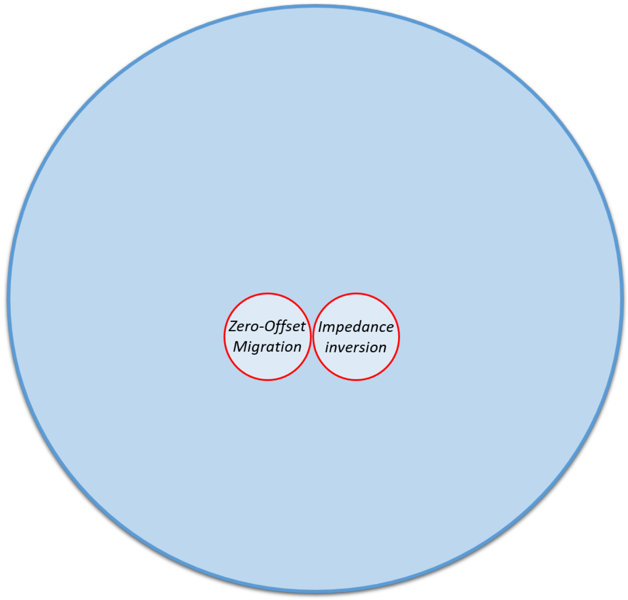


Figure 8.6: The post-stack migration based gradient with well calibration (chapter 7 reduces the total cost of the FWI method to the same as the post-stack migration per iteration.

final model that helps on the interpretation of the seismic data.

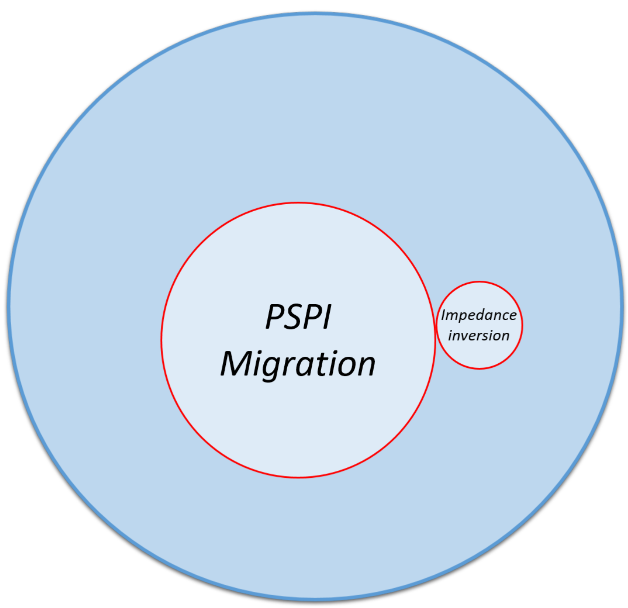


Figure 8.7: The pre-stack migration based gradient with well calibration (chapter 7. The cost is equivalent of the pre-stack migration at each iteration.

Also in chapter 7, resolution is improved be replacing the post-stack migration by a pre-stack one. The total cost of the method is increased, as show on figure 8.7, but the model presented higher resolution. As discussed before, the choice of the FWI method will depends on user trade-off decisions and needs.

In the end, I could test the standard steepest-descent FWI routine and make a few new interpretations of the gradient to end up with a cheap, simple, and high precision strategy.

Bibliography

Bishop, T. N. and Spongberg, M. E. (1984). Seismic tomography: A case study. *SEG Technical Program Expanded Abstracts*, 313:712–713.

Chipman, J. S. (1999). Linear restrictions, rank reduction, and biased estimation in linear regression. *Linear Algebra and its Applications*, 289(1):55 – 74.

Claerbout, J. F. (1971). Toward a unified theory of reflection mapping. *Geophysics*, 36(3):467–481.

Cutler, R. T., Bishop, T. N., Wyld, H. W., Shuey, R. T., Kroeger, R. A., Jones, R. C., and Rathbun, M. L. (1984). Seismic tomography: Formulation and methodology. *SEG Technical Program Expanded Abstracts*, 312:711–712.

Fatti, J. L., Smith, G. C., Vail, P. J., Strauss, P. J., and Levitt, P. R. (1994). Detection of gas in sandstone reservoirs using avo analysis: A 3-d seismic case history using the geostack technique. *Geophysics*, 59(9):1362–1376.

Fei, T. W., Luo, Y., Qin, F., and Kelamis, P. G. (2012). Full waveform inversion without low frequencies: A synthetic study. *SEG Technical Program Expanded Abstracts*, 484:1–5.

Ferguson, R. and Margrave, G. (2005). Planned seismic imaging using explicit one-way operators. *Geophysics*, 70(5):S101–S109.

Ferguson, R. J. and Margrave, G. (1996). A simple algorithm for band-limited impedance inversion. *CREWES Research Report*, 83:21.1–21.10.

Gazdag, J. and Sguazzero, P. (1984). Migration of seismic data by phase shift plus interpolation. *Geophysics*, 49(2):124–131.

Gray, S. H. (1999). Speed and accuracy of seismic migration methods. *Depth imaging of foothills seismic data: Can. Soc. Expl. Geophys*, pages 47–83.

Guarido, M., Lines, L., and Ferguson, R. (2014). Full waveform inversion - a synthetic test using the pspi migration. *CREWES Research Report*, 26:26.1–26.23.

Guarido, M., Lines, L., and Ferguson, R. (2015). Full waveform inversion: a synthetic test using pspi migration. *SEG Technical Program Expanded Abstract*, 279:1456–1460.

Guarido, M., Lines, L., and Ferguson, R. (2016). Fwi without tears: a forward modeling free gradient. *CREWES Research Report*, 26:26.1–26.20.

Lailly, P. (1983). The seismic inverse problem as a sequence of before stack migrations: Conference on inverse scattering, theory and application. *Society of Industrial and Applied Mathematics, Expanded Abstracts*, pages 206–220.

Langan, R. T., Lerche, I., Cutler, R. T., Bishop, T. N., and Spera, N. J. (1984). Seismic tomography: The accurate and efficient tracing of rays through heterogeneous media. *SEG Technical Program Expanded Abstracts*, 314:713–715.

Lindseth, R. O. (1979). Synthetic sonic logs-a process for stratigraphic interpretation. *Geophysics*, 44(1):3–26.

Lines, L. and Treitel, S. (1984). A review of least-squares inversion and its application to geophysical problems. *Geophysical Prospecting*, 32:159–186.

Ma, Y., Hale, D., Meng, Z. J., and Gong, B. (2010). Full waveform inversion with image-guided gradient. *SEG Technical Program Expanded Abstracts*, 198:1003–1007.

Marfurt, K. J. (1984). Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations. *Geophysics*, 49(5):533–549.

Margrave, G., Ferguson, R., and Hogan, C. (2010). Full waveform inversion with wave equation migration and well control. *CREWES Research Report*, 22:63.1–63.20.

Margrave, G., Yedlin, M., and Innanen, K. (2011). Full waveform inversion and the inverse hessian. *CREWES Research Report*, 23:77.1–77.13.

Pan, W., Margrave, G., and Innanen, K. (2013). On the role of the deconvolution imaging condition in full waveform inversion. *CREWES Research Report*, 25:72.1–72.19.

Pica, A., Diet, J. P., and Tarantola, A. (1990). Nonlinear inversion of seismic reflection data in a laterally invariant medium. *Geophysics*, 55(2):R59–R80.

Plessix, R.-E., Baeten, G., de Maag, J. W., Klaassen, M., Rujie, Z., and Zhifei, T. (2010). Application of acoustic full waveform inversion to a low-frequency large-offset land data set. *SEG Technical Program Expanded Abstracts*, 184:930–934.

Pratt, R. G., Shin, C., and Hick, G. J. (1998). Gauss-newton and full newton methods in frequency-space seismic waveform inversion. *Geophysical Journal International*, 133(2):341–362.

Romahn, S. and Innanen, K. (2016). Immi: the role of well calibration in the context of high geological complexity. *CREWES Research Report*, 28:66.1–66.17.

Shuey, R. T. (1985). A simplification of the zoeppritz equations. *Geophysics*, 50(4):609–614.

Tarantola, A. (1984). Inversion of seismic reflection data in the acoustic approximation. *Geophysics*, 49(8):1259–1266.

Treitel, S. (1989). Quo vadit inversio? *Geophysics: The Leading Edge of Exploration*, pages 38–42.

Treitel, S., Lines, L., and Ruckgaber, G. (1995). Seismic impedance estimation. *Geophysical Inversion and Applications - Memorial University of Newfoundland*, 1:6–11.

Versteeg, R. (1991). Analyse du problme de la determination du modle de vitesse pour l'imagerie sismique. *PhD Thesis, University of Paris*.

Vigh, D. and Starr, E. W. (2008). 3d prestack plane-wave, full-waveform inversion. *GEOPHYSICS*, 73(5):VE135–VE144.

Virieux, J. and Operto, S. (2009). An overview of full-waveform inversion in exploration geophysics. *Geophysics*, 74(6):WCC1–WCC26.

Warner, M. and Guasch, L. (2014). Adaptive waveform inversion: Theory. *SEG Technical Program Expanded Abstracts*, 207:1089–1093.

Warner, M., Ratcliffe, A., Nangoo, T., Morgan, J., Umpleby, A., Shah, N., Vinje, V., Stekl, I., Guasch, L., Win, C., Conroy, G., and Bertrand, A. (2013). Anisotropic 3d full-waveform inversion. *Geophysics*, 78(2):R59–R80.

Youzwishen, C. F. and Gary, F. M. (1999). Finite difference modeling of acoustic waves in matlab. *CREWES Research Report*, 11:11.1–11.19.

Zhou, C., Cai, W., Luo, Y., Schuster, G. T., and Hassanzadeh, S. (1995). Acoustic wave-equation traveltime and waveform inversion of crosshole seismic data. *GEOPHYSICS*, 60(3):765–773.