3D elastic time-frequency full-waveform inversion

Zur Erlangung des akademischen Grades eines
DOKTORS DER NATURWISSENSCHAFTEN
von der Fakultät für Physik des
Karlsruher Instituts für Technologie (KIT)

genehmigte

DISSERTATION

von

Dipl.-Geophys. Simone Butzer (geborene Dunkl)
aus Gräfelfing

Tag der mündlichen Prüfung:	9. Januar 2015
Referent:	Prof. Dr. Thomas Bohlen
Korreferent:	Prof. Dr. Hansruedi Maurer

Abstract

3D elastic full waveform inversion (FWI) aims at reconstructing highly-resolved images of the elastic subsurface parameters from seismic data. In contrast to conventional seismic imaging methods, the full information content of the seismograms is exploited by minimising the misfit between observed and modelled waveforms. However, mainly due to the high computational costs of this method, applications nowadays are mostly limited to the 2D or acoustic approximation of the seismic wave equation. In this thesis, I present the theory and implementation of a 3D elastic FWI which allows for the reconstruction of 3D subsurface structures parameterised by seismic velocities and density. The approach is based on the adjoint state method and minimizes the misfit in the conjugate gradient direction. A velocity-stress finite difference scheme in time domain enables a fast and efficient simulation of wavefields. The gradients, by contrast, are calculated in frequency domain from monochromatic particle velocity wavefields resulting in significantly lower storage costs compared to an implementation in the time domain. A discrete Fourier transform is employed to extract monochromatic wavefields on the fly during forward propagation.

The inversion of data of a random medium velocity model resembling crystalline rock shows the ability of the method to reconstruct small-scaled 3D heterogeneities in transmission geometry. Even with a very simple geometry of a line of sources and a line of receivers, differently sized 3D structures adjacent to the source-receiver plane are recovered. In comparison, a 2D FWI yields a similar resolution in the source-receiver plane. However, in this approximation, artefacts are caused by explaining the 3D scattering with a 2D inversion scheme.

To optimise the 3D FWI method and to allow for the inversion of complex surface geometry models, two optimisations were investigated: the use of a diagonal Hessian approximation for preconditioning and the L-BFGS algorithm. The diagonal Hessian approximation is calculated once per frequency stage from partial derivative wavefields in frequency domain. To mitigate computational costs, only a subset of receivers is employed. The L-BFGS method includes information about the misfit curvature by considering gradient and model changes in the previous few iterations. A dimensionless L-BFGS scheme is implemented using a parameter normalisation. This method automatically scales the parameter classes with respect to each other. Both approaches were validated for a simple box model.

These optimisations are used in a 3D FWI applied to a complex 3D layered surface model. Due to surface acquisition geometry and the presence of surface waves, the misfit gradients mainly emphasise very shallow areas. A simple tapering of gradients around sources and receivers was not sufficient for the inversion to succeed. Preconditioning with the diagonal Hessian ap-

proximation, by contrast, enables a successful reconstruction of the seismic velocity models, including the deeper layers. In this approach the approximate Hessian accounts for geometric amplitude effects in the gradient. Preliminary results using the L-BFGS approach result in a comparable image. The main advantage is the significant reduction in computational effort by increasing the convergence.

Contents

Chapter 1

Introduction

Seismic waves are influenced by various physical subsurface parameters, such as wave velocities, density, attenuation and anisotropy. The role of inversion is the inference of this information about the subsurface from recorded seismic data to reconstruct a detailed and correct subsurface model. The seismic inversion is applied in various areas, including exploration seismics, geotechnical surveys or research on structures and processes in the earth's interior, among others. Conventional methods mainly evaluate the travel times of different seismic phases, like direct, refracted or reflected waves. In the 1980's, however, Lailly (1983), Tarantola (1984) and Mora (1987) suggested a new inversion strategy known as adjoint full-waveform inversion (FWI). By iteratively minimising the misfit between observed and modelled seismograms this technique uses the full information content given by the waveforms. It thus promises a significantly improved resolution compared to conventional methods.

A straightforward approach to perform the FWI is a global search of the parameter space, like a neighbourhood algorithm. However, this is only possible for a simple subsurface characterised by few unknown parameters, which strongly limits the field of application. On the assumption of a 3D model described by a huge number of parameters, the global search is far too expensive and cannot be conducted. Thus, in general, FWI is restricted to the use of a local optimisation method, which converge towards the misfit minimum in the vicinity of the starting model. However, due to the high number of model parameters combined with a generally bad data coverage, the problem becomes highly non-linear. Local inversion methods, therefore, bear the danger to end up in a local minimum of the misfit function. A good starting model and thorough inversion strategies are required for a successful inversion.

FWI of realistic models is computationally expensive and only feasible due to major advances in computer technology. The most widely used optimisation methods in seismic FWI are gradient-based algorithms, which update the model in the direction of the steepest descent of the misfit function. Gradient directions can be implemented very efficiently with the adjoint approach (e.g. Tarantola, 1984; Mora, 1987) and these methods are thus applicable even for larger problem sizes. More sophisticated is the class of Newton methods which take the Hessian, the second derivative of the misfit function, into account. They lead to a better convergence and can result in improved subsurface images. Still, the high computational costs make them unattractive for large inverse problems (Pratt, 1999). An alternative are quasi-Newton algorithms, such as the L-BFGS method proposed by Byrd et al. (1995). It estimates the Hessian from changes in the misfit gradient and can therefore be implemented at lower computational costs. Another method successfully applied in FWI is the truncated Newton method, which approximates the

Hessian from second order adjoint formula (Métevier et al., 2012; Fichtner & Trampert, 2011). The adjoint method was originally formulated in the time domain (e.g., Tarantola, 1984; Mora, 1987). Additionally, in the 90's frequency domain FWI was developed by the group of Pratt (e.g., Pratt & Worthington, 1990; Pratt, 1990, 1999). In contrast to time domain FWI, wavefields are modelled and inverted only for few frequencies which corresponds to a limitation of data. This can be of advantage, especially from the numerical point of view. To use advantages of time-domain forward modelling and frequency inversion, a mixed approach was proposed (Sirgue et al., 2008), which is referred to as time-frequency approach in this thesis. Hereby modelling is performed with an efficient forward solver in time domain (e.g., Virieux, 1986; Bohlen, 2002), whereas the inverse problem is solved for discrete frequencies in frequency domain. This can be particularly attractive for 3D FWI.
Of course, an accurate modelling of seismic wavefields is a basic requirement for a successful FWI. This implies a good physical model representation of the subsurface. The sensitivity of the waveforms to different parameter classes offers the opportunity to perform multi-parameter inversions of complex 3D structures. However, with an increase of model parameters, the nonlinearity of the inverse problem is enforced and the computational costs rise. Hence, depending on the problem, it is common to use certain approximations of the wave equation. A main area of FWI application is the marine environment. Here wave propagation is mainly approximated using the acoustic and visco-acoustic wave equation. which only considers compressional waves. For land seismic, the abundance of prominent shear waves and surface waves favours an inversion, involving the according parameters, such as compressional-wave velocity, shear-wave velocity and density. Depending on the subsurface, attenuation and anisotropy need to be included. Still, these parameter classes are very challenging to invert for (Virieux & Operto, 2009) and are often integrated only as passive parameters in the forward simulation.
Many applications are still restricted to the 2D approximation which assumes a 2D model of the subsurface. This is especially the case for elastic applications (e.g. Sheen et al., 2006; Sears et al., 2008; Brossier et al., 2009; Groos et al., 2012; Köhn et al., 2012). The main reason is the dramatic increase in computational costs for a 3D compared to a 2D FWI. Additionally, data sets are often recorded along single profiles. Nevertheless, the 2D approximation is unable to explain 3D scattering arising from 3D heterogeneous subsurface structures. Furthermore, only 2D images of the 3D subsurface are reconstructed. Thus, today, an increasing number of acoustic FWI are performed in 3D and satisfactory results were obtained for real data applications, such as Plessix (2009), Sirgue et al. (2010a), Vigh et al. (2011) and Etienne et al. (2012). A study by Warner et al. (2013) even includes anisotropy. By contrast, studies on the implementation and application of 3D elastic FWI as performed by Epanomeritakis et al. (2008), Fichtner et al. (2009), Castellanos et al. (2011), Guasch et al. (2012) and Butzer et al. (2013) are rare and computationally still very expensive.

The main objective of this thesis is the implementation of a 3D elastic FWI and its application to multi-parameter problems with complex 3D structures. The implementation 3D-AWAIT (**3D**-**A**djoint **WA**veform **I**nversion **T**ool) is based on the conjugate gradient method using the time-frequency approach. It is optimised regarding runtime and storage to enable the solution of the large 3D inverse problem. To improve the performance in case of complex models I tested a preconditioning technique based on a diagonal Hessian approximation and the L-BFGS method. Simple toy examples are used for validation of the code. Additionally, the performance

of 3D FWI is presented for two larger applications, including the inversion of small-scaled 3D heterogeneities in transmission geometry and the inversion of sedimentary layers using a surface acquisition geometry.

Chapter 2 will start with an overview of elastic wave propagation. The concept of FWI and local optimisation methods are introduced. Afterwards I focus on the conjugate gradient approach applied for the 3D elastic wave equation. The gradient calculations for different elastic parameters are derived in time and frequency domain.

Chapter 3 presents the implementation of the theoretical aspects discussed in chapter 1. The modelling of wavefields is performed with a finite-difference solver in time domain, whereas the gradient calculation is conducted in frequency domain. The implementation and the main advantages of this time-frequency approach are discussed. Additionally, the complete workflow and its realisation is presented and visualised by means of a toy example.

Chapter 4 shows the application of 3D elastic FWI on small-scale heterogeneities of crystalline structures, described by a random medium model in transmission geometry. I compare a 2D and a 3D acquisition geometry for a simple checkerboard model and the random medium model. Additionally the influence of 3D scattered waves on the results of a 2D FWI is shown.

Chapter 5 deals with the preconditioning of the gradient with a diagonal Hessian approximation. Both theory and implementation of the diagonal Hessian approximation are explained and tested for a simple box model in transmission geometry. Finally, the approach is applied to a complex surface-geometry model of 3D sedimentary layers.

Chapter 6 extends the conjugate gradient approach with the L-BFGS method. It comprises theoretical considerations, implementation and validation for a simple box example. Finally, first inversion results with L-BFGS of the surface-geometry model are presented.

Chapter 7 gives a summary and conclusion of my thesis.

Chapter 2

Theory of full-waveform inversion

In this chapter I will introduce the theory of full-waveform inversion (FWI). For a general understanding I start with a short section about wave propagation in elastic medium. This is followed by a general introduction of the seismic inverse problem in FWI. I will give an overview about the underlying concept of Newton and gradient methods as local optimisation methods before concentrating on the gradient approach, which will be discussed in detail. From perturbation theory, I derive the expressions for the gradients in time domain, which are calculated with the adjoint method. Additionally, gradient equations in frequency domain and for different parametrisation are formulated.

2.1 Wave propagation in elastic media

FWI aims to find the subsurface model by minimising the misfit between the observed and modelled data. In order to do this, a sufficiently accurate description of seismic wave propagation is essential. In this section, I will give an overview about seismic wave propagation in a three-dimensional elastic medium. For more detailed information I refer to the general literature, like such as Lay & Wallace (1995) and Aki & Richards (1980).
I consider a small subvolume of an elastic continuum. Forces can either act within the subvolume as body forces or on its surface as contact forces. If we consider an internal plane with surface S intersecting the medium, the contact forces at one point P of the plane are described by the traction vector $\mathbf{\Sigma}$ defined as

$$\mathbf{\Sigma}(\mathbf{n}) = \lim_{\Delta S \to 0} \left(\frac{\Delta \mathbf{F}}{\Delta S} \right), \tag{2.1}$$

where $\Delta\mathbf{F}$ denotes the force acting on the small element with surface ΔS around point P and $\mathbf{n}$ is the normal on this surface element. As a general description, the nine component stress tensor τ_{ij} (i=1,2,3; j=1,2,3) is introduced. Its elements define the stress components acting on the Cartesian coordinate planes. For the arbitrary plane with normal $\mathbf{n}$ the i-th component of the traction vector is then calculated as

$$\Sigma_i = \tau_{ij} n_j \tag{2.2}$$

In this and the following theory I use the Einstein notation and sum up over double indices.
Forces acting on an elastic medium also cause deformation and thus spatial gradients of the

displacement $\mathbf{u}(\mathbf{x})$. For small displacement and strain this internal deformation can be described by the symmetric, nine-component infinitesimal strain tensor ε_{ij} given by

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{2.3}$$

The relationship between stress and strain for linear elasticity is described by Hook's law

$$\tau_{ij} = C_{ijkl}\varepsilon_{kl}, \tag{2.4}$$

which is mostly valid for the materials of the subsurface.
The third-order stiffness tensor C_{ijkl} contains 21 independent components to describe the general anisotropic medium. In case of an elastic, isotropic medium the tensor can be reduced to two elasticity coefficients, the Lamé parameters λ and μ:

$$C_{ijkl} = \lambda\delta_{ij}\delta_{kl} + \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}). \tag{2.5}$$

Wave propagation in continuous media follows the equation of motion, which can be derived from Newton's second law. It states that the change in momentum equals the sum of all forces:

$$\int_V \mathbf{f}dV + \int_S \mathbf{\Sigma}(\mathbf{n})dS = \frac{\partial}{\partial t}\int_V \rho\frac{\partial \mathbf{u}}{\partial t}dV. \tag{2.6}$$

$\mathbf{f}$ denotes the body force per unit volume and $\mathbf{\Sigma}$ the traction acting on the surface element dS. Rewriting the equation by using equation 2.2 and Gauss's theorem we find

$$\rho\frac{\partial^2 u_i}{\partial t^2} = f_i + \frac{\partial \tau_{ij}}{\partial x_j}. \tag{2.7}$$

The relation between stress and displacement is then given by Hook's law (equation 2.4). Together with suitable boundary and initial conditions, the equations 2.7 and 2.4 describe the wave propagation in elastic medium caused by an external force described as body force $\mathbf{f}$ and can thus be used for the modelling of seismic waves. This formulation is known as displacement-stress formulation.
For the isotropic medium equation 2.4 simplifies with the use of equation 2.5 to

$$\tau_{ij} - \lambda\Theta\delta_{ij} + 2\mu\varepsilon_{ij} = 0. \tag{2.8}$$

The trace of the stress tensor Θ is given as $\Theta = \tau_{xx} + \tau_{yy} + \tau_{zz}$.
Wave propagation can also be expressed using a velocity-stress formulation. This formulation uses the particle velocity $\mathbf{v} = \partial\mathbf{u}/\partial t$ as wavefield parameter. For the elastic isotropic medium wave propagation due to a force $\mathbf{f}$ can then be described as first order differential equations:

$$\begin{aligned} \rho\frac{\partial v_i}{\partial t} &= f_i + \frac{\partial \tau_{ij}}{\partial x_j}, \\ \frac{\partial \tau_{ij}}{\partial t} - \lambda\frac{\partial \Theta}{\partial t}\delta_{ij} + 2\mu\frac{\partial \varepsilon_{ij}}{\partial t} &= 0, \\ \frac{\partial \varepsilon_{ij}}{\partial t} &= \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right). \end{aligned} \tag{2.9}$$

2.2 The inverse problem

Seismic tomography aims to derive properties of the subsurface from seismic data observed at different receiver locations for a number of shots. Thus, model parameters in model space are determined from measurements in data space. The seismic inverse problem is generally underdetermined as only limited data from restricted receivers and sources is available to deduce a 3D subsurface model. Conventional methods additionally only use few information from the seismograms, like travel times or dispersion curves. The full-waveform inversion, by contrast, uses the full information content given in the waveforms and can consequently offer a much better resolved image of the 3D subsurface structures. By minimising the misfit between the observed and some modelled data it aims to determine the optimal model, which explains the observed data best. The direct way is to apply a global search, like the Monte Carlo approach to find the global misfit minimum. However, this is computationally accomplishable only for a very limited set of model parameters and thus not suited for the general three dimensional multiparameter model. Therefore, local optimisation methods, like Newton methods and gradient methods are commonly used in FWI, which can be implemented very efficiently. These methods search for a minimum of the misfit function in the vicinity of the starting model. However, the seismic inverse problem is generally highly nonlinear and the misfit function is not smooth, but characterised by local minima. A local inversion method bears the danger to end up in one of the local minima, which is a major problem in FWI. Thus, a sufficiently accurate starting model is essential for a successful inversion. Additionally, a multistage procedure, where data information is included piecewise, can help to mitigate the nonlinearity.

2.3 The concept of Newton and gradient methods

In the following, I will describe the idea of Newton and gradient methods for the solution of the optimisation problem. Detailed description of these methods are for example given by Pratt et al. (1998) and Nocedal & Wright (1999). I minimise the misfit function E, which is a measure for the fit between modelled and observed data. A Taylor expansion of the misfit function E around the model parameters $\mathbf{m} = (m_1, ..., m_n)^T$ leads to:

$$E(\mathbf{m} + \delta\mathbf{m}) = E(\mathbf{m}) + \delta\mathbf{m}^T \nabla_m E(\mathbf{m}) + \frac{1}{2}\delta\mathbf{m}^T \mathbf{H} \delta\mathbf{m} + \mathcal{O}(|\delta\mathbf{m}|^3) \tag{2.10}$$

with the misfit gradient as $\nabla_m E(\mathbf{m}) = (\frac{\partial}{\partial m_1} E(\mathbf{m}), ..., \frac{\partial}{\partial m_n} E(\mathbf{m}))^T$. $\mathbf{H}$ is the Hessian operator, defined as the second derivative of the misfit with respect to the model parameters, i.e.,

$$H_{ij} = \frac{\partial^2 E(\mathbf{m})}{\partial m_i \partial m_j} \quad (i = 1, ...n) \ \ (j = 1, ...n). \tag{2.11}$$

The index n is the total number of model parameters. We use a quadratic approximation and neglect terms higher than second order in equation 2.10, which corresponds to a linearisation of the inverse problem. To find the minimum of the misfit function the derivative of equation 2.10 with respect to the model perturbation $\delta\mathbf{m}$ is set zero. This gives us the following model update:

$$\delta\mathbf{m} = -\mathbf{H}^{-1} \nabla_m E \tag{2.12}$$

which is used in the full Newton method. In case of a linear inversion problem, an update with $\delta\mathbf{m}$ would lead into the minimum of the misfit function within one step. Non-linear inversion problems, like FWI are linearised step-wise and iteratively approach the minimum. Then, the Newton method iteratively updates the model in iteration $k+1$ with

$$\mathbf{m}_{k+1} = \mathbf{m}_k - \mathbf{H}_k^{-1}\nabla_m E_k. \tag{2.13}$$

The use of the Hessian, which includes information of the second derivative of the misfit function leads to a good convergence. However, the calculation and inversion of the large $n \times n$ Hessian matrix is highly expensive and not possible for larger scaled problems, as they occure in FWI. Hence, different approximations of the full Newton method developed. The Gauss-Newton method uses an approximation of the Hessian, known as the "approximate Hessian" (Pratt et al., 1998). It is still computationally costly, as it calculates the full approximate Hessian matrix with $n \times n$ elements. Additionally, quasi-Newton methods, like the L-BFGS method and the truncated Newton method were developed. They do not calculate the Hessian matrix directly, but use information about the change of the gradients and model parameters to include information about the second derivative of the misfit function. These methods are computationally less expensive and still offer a superlinear convergence (Nocedal & Wright, 1999).
Another class of optimisation methods are the gradient methods, where the search direction shows in the direction of steepest descent of the misfit function. The gradient method uses the following simplified update

$$\mathbf{m}_{k+1} = \mathbf{m}_k - \alpha_k \mathbf{p}_k \quad \text{with} \quad \mathbf{p}_k = \mathbf{P}\nabla_m E_k. \tag{2.14}$$

In this case, the gradient is not scaled and preconditioned by the inverse Hessian. Consequently, a preconditioning operator $\mathbf{P}$ and the estimation of an appropriate step length α are required for the inversion to succeed. The search direction is given by the preconditioned gradient $\mathbf{p}_k$. The corresponding convergence is low compared to the Newton method. Nevertheless, gradient methods can be implemented very efficiently and require less computational operations per iteration and less storage than Newton and also quasi Newton methods. Thus, they are suitable for larger FWI applications, as they occure in 3D FWI. Some improvement in convergence of the gradient method is gained by using the conjugate gradient $\mathbf{c}_k$ as search direction (Mora, 1987), which is given by

$$\mathbf{c}_k = \mathbf{P}\nabla_m E_k + \beta_k \mathbf{c}_{k-1}. \tag{2.15}$$

This stabilises the inversion by adding some part of the previous gradient. The scalar β can be estimated with the Polak-Ribiére method as

$$\beta_k = \frac{\mathbf{p}_k^T(\mathbf{p}_k - \mathbf{p}_{k-1})}{||p_{k-1}||^2} \text{ for } \beta_k > 0; \quad \beta_k = 0 \text{ for } \beta_k < 0. \tag{2.16}$$

The distinction of cases ensures, that the model is updated in the decent direction of the misfit function. The new model is updated by

$$\mathbf{m}_{k+1} = \mathbf{m}_k - \alpha_k \mathbf{c}_k. \tag{2.17}$$

In the following I will concentrate on the theory of the conventional conjugate gradient method

for FWI in more detail. Its implementation and performance for 3D elastic FWI are shown in the chapters 3 and 4, respectively. For an optimisation of the conventional conjugate gradient approach it is sensible to include some information about the Hessian in the inversion. In chapter 5, I will show the implementation and application of an approximation of the diagonal Hessian for preconditioning. Finally, in chapter 6 the effects of the L-BFGS approach in 3D FWI are investigated.

2.4 Calculation of the misfit gradient

I consider the L_2-norm based misfit function E

$$E(\mathbf{m}) = \frac{1}{2}\sum_s \int_0^T dt \sum_r \delta u_i(\mathbf{x}_s, \mathbf{x}_r, t)^2, \tag{2.18}$$

with the i-th component of the displacement residual $\delta u_i = u_i - u_{i,obs}$ at source position $\mathbf{x}_s$ and receiver position $\mathbf{x}_r$. $\mathbf{u}_{obs}$ denotes the observed data, whereas $\mathbf{u}$ denotes the modelled data. The squared residuals are summed up over all sources s and receivers r and integrated over time t for the full record length T. In this and the following equations I use the Einstein notation to sum up over double indices.
Differentiating E with respect to the model parameter m_j leads to

$$\frac{\partial E(\mathbf{m})}{\partial m_j} = \sum_s \int_0^T dt \sum_r \frac{\partial u_i}{\partial m_j} \delta u_i \tag{2.19}$$

Gradients can be calculated with the adjoint method, which can be derived from perturbation theory (e.g. Tarantola, 1984; Mora, 1987). I consider the linearised forward problem and calculate a small perturbation in data space $\delta\mathbf{u}$ resulting from a small perturbation in model space $\delta\mathbf{m}$, expressed as

$$\delta\mathbf{u} = \mathbf{D}\delta\mathbf{m} \;\text{ and }\; \delta u_i = \int_V dV(\mathbf{x}) \frac{\partial u_i}{\partial m_j} \delta m_j. \tag{2.20}$$

The kernel of the linear operator $\mathbf{D}$ is given by the Fréchet derivatives $\partial u_i / \partial m_j$. The adjoint problem can be described as

$$\delta\mathbf{m}' = \mathbf{D}^* \delta\mathbf{m} \;\text{ and }\; \delta m'_j = \sum_s \int_0^T dt \sum_r \frac{\partial u_i}{\partial m_j} \delta u_i. \tag{2.21}$$

To derive the adjoint equation, I used the fact that the kernels of the linear operator $\mathbf{D}$ and its adjoint $\mathbf{D}^*$ are the same (Tarantola, 1984; Mora, 1987). By comparing equation 2.21 and equation 2.19 it is visible that the gradient of the misfit function describes the transformation from data space to model space corresponding to a small perturbation $\delta\mathbf{u}$ as given in equation 2.21 and that $\nabla_m E = \delta\mathbf{m}'$.
I now turn to the seismic inverse problem and consider a small model perturbation $(\delta\rho, \delta C_{ijkl})$ which causes a perturbation in displacement $(\delta\mathbf{u})$, in stress $(\delta\tau_{ij})$ and in strain $(\delta\varepsilon_{kl})$. To estimate the effects of the disturbed medium on the wavefield, I insert the perturbations into the second order wave equation (2.7) and Hook's law (equation 2.4) and find:

$$(\rho + \delta\rho)\frac{\partial^2}{\partial t^2}(u_i + \delta u_i) - \frac{\partial}{\partial x_j}(\tau_{ij} + \delta\tau_{ij}) = f_i \tag{2.22}$$

$$(\tau_{ij} + \delta\tau_{ij}) - (C_{ijkl} + \delta C_{ijkl})(\varepsilon_{kl} + \delta\varepsilon_{kl}) = 0. \tag{2.23}$$

Taking only first order perturbations into account the wave equations of the perturbed wavefield can be written as

$$\rho \frac{\partial^2 \delta u_i}{\partial t^2} - \frac{\partial \delta\tau_{ij}}{\partial x_j} = \Delta f_i \quad \text{with} \quad \Delta f_i = -\delta\rho \frac{\partial^2 u_i}{\partial t^2} \tag{2.24}$$

and

$$\delta\tau_{ij} - C_{ijkl}\delta\varepsilon_{kl} = \Delta T_{ij} \quad \text{with} \quad \Delta T_{ij} = \delta C_{ijkl}\varepsilon_{kl}. \tag{2.25}$$

Thus, in first order approximation, the model perturbations cause a secondary wavefield, which propagates through the unperturbed medium and is described in the form of an elastic wave equation. The new source terms Δf_i and ΔT_{ij}, also known as "secondary Born sources", contain the model perturbations.
The solution of the elastic wave equation can be expressed in terms of the elastic Green's functions G_{ij} (equation A.2). For the perturbation δu_i this results in (e.g. Wu & Aki, 1985; Mora, 1987)

$$\delta u_i(\mathbf{x}_r, t) = \int_V dV \int_0^T d\tau G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau)\Delta f_j(\mathbf{x}, \tau) - \int_V dV \int_0^T d\tau \frac{\partial}{\partial x_k} G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau)\Delta T_{jk}(\mathbf{x}, \tau). \tag{2.26}$$

In the following I will consider only the isotropic case, where C_{ijkl} can be rewritten with the Lamé parameters λ and μ as given in equation 2.5. In this case, ΔT_{ij} becomes

$$\Delta T_{ij} = \delta_{ij}\varepsilon_{kk}\delta\lambda + (\varepsilon_{ij} + \varepsilon_{ji})\delta\mu = \delta_{ij}\frac{\partial u_k}{\partial x_k}\delta\lambda + \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\delta\mu. \tag{2.27}$$

Hereby we used equation 2.3 to express stress values with displacement derivatives. If I insert the forces ΔT_{jk} and Δf_j from equations 2.24 and 2.27 into equation 2.26 the equation has the form:

$$\delta u_i(\mathbf{x}_r, t) = \int_V dV K_i^\rho \delta\rho(\mathbf{x}) + \int_V dV K_i^\lambda \delta\lambda(\mathbf{x}) + \int_V dV K_i^\mu \delta\mu(\mathbf{x}). \tag{2.28}$$

Comparing this equation to equation 2.20 it becomes clear, that K_i^ρ, K_i^λ and K_i^μ are the Fréchet derivative kernels $\partial u_i/\partial m_j$. They are given by

$$\begin{aligned}
\frac{\partial u_i(\mathbf{x}_r, t)}{\partial \rho(\mathbf{x})} &= K_i^\rho = -\int_0^T d\tau G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau)\frac{\partial^2 u_j(\mathbf{x}, \tau)}{\partial \tau^2} = -\int_0^T d\tau \frac{\partial}{\partial \tau} G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau)\frac{\partial u_j(\mathbf{x}, \tau)}{\partial \tau} \\
\frac{\partial u_i(\mathbf{x}_r, t)}{\partial \lambda(\mathbf{x})} &= K_i^\lambda = -\int_0^T d\tau \frac{\partial}{\partial x_k} G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau)\delta_{jk}\frac{\partial u_p(\mathbf{x}, \tau)}{\partial x_p} = -\int_0^T d\tau \frac{\partial}{\partial x_j} G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau)\frac{\partial u_p}{\partial x_p} \\
\frac{\partial u_i(\mathbf{x}_r, t)}{\partial \mu(\mathbf{x})} &= K_i^\mu = -\int_0^T d\tau \frac{\partial}{\partial x_k} G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau)\left(\frac{\partial u_j(\mathbf{x}, \tau)}{\partial x_k} + \frac{\partial u_k(\mathbf{x}, \tau)}{\partial x_j}\right)
\end{aligned} \tag{2.29}$$

In the expression of K_i^ρ, the time derivatives could be interchanged as a property of convolution. The Fréchet derivatives (equation 2.29) can now be inserted into the gradient equations 2.19 and

I find:

$$\begin{aligned}
\frac{\partial E(\mathbf{m})}{\partial \rho(\mathbf{x})} &= \sum_s \int_0^T dt \sum_r \left[- \int_0^T d\tau \frac{\partial}{\partial \tau} G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau) \frac{\partial u_j(\mathbf{x}, \tau)}{\partial \tau} \right] \delta u_i(\mathbf{x}_s, \mathbf{x}_r, t) \\
\frac{\partial E(\mathbf{m})}{\partial \lambda(\mathbf{x})} &= \sum_s \int_0^T dt \sum_r \left[- \int_0^T d\tau \frac{\partial}{\partial x_j} G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau) \frac{\partial u_p(\mathbf{x}, \tau)}{\partial x_p} \right] \delta u_i(\mathbf{x}_s, \mathbf{x}_r, t) \\
\frac{\partial E(\mathbf{m})}{\partial \mu(\mathbf{x})} &= \sum_s \int_0^T dt \sum_r \left[- \int_0^T d\tau \frac{\partial}{\partial x_k} G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau) \left(\frac{\partial u_j(\mathbf{x}, \tau)}{\partial x_k} + \frac{\partial u_k(\mathbf{x}, \tau)}{\partial x_j} \right) \right] \delta u_i(\mathbf{x}_s, \mathbf{x}_r, t).
\end{aligned} \tag{2.30}$$

To simplify these equations, I define a function $\mathbf{\Psi}(\mathbf{x}, \tau)$ which components are given as

$$\Psi_j(\mathbf{x}, \tau) = \sum_r \int_0^T dt G_{ij}(\mathbf{x}_r, t; \mathbf{x}, \tau) \delta u_i(\mathbf{x}_r, t). \tag{2.31}$$

Then, the equations 2.30 become

$$\begin{aligned}
\frac{\partial E(\mathbf{m})}{\partial \rho(\mathbf{x})} &= - \sum_s \int_0^T d\tau \frac{\partial \Psi_j(\mathbf{x}, \tau)}{\partial \tau} \frac{\partial u_j(\mathbf{x}, \tau)}{\partial \tau} \\
\frac{\partial E(\mathbf{m})}{\partial \lambda(\mathbf{x})} &= - \sum_s \int_0^T d\tau \frac{\partial \Psi_j(\mathbf{x}, \tau)}{\partial x_j} \frac{\partial u_p(\mathbf{x}, \tau)}{\partial x_p} \\
\frac{\partial E(\mathbf{m})}{\partial \mu(\mathbf{x})} &= - \sum_s \int_0^T d\tau \frac{\partial \Psi_j(\mathbf{x}, \tau)}{\partial x_k} \left(\frac{\partial u_j(\mathbf{x}, \tau)}{\partial x_k} + \frac{\partial u_k(\mathbf{x}, \tau)}{\partial x_j} \right).
\end{aligned} \tag{2.32}$$

The wavefield Ψ_j in equation 2.31 describes the adjoint or back-propagated wavefield. We use the spatial reciprocity of the Green's function (equation A.3) and estimate the wavefield at a time $T - \tau$ as

$$\Psi_j(\mathbf{x}, T - \tau) = \sum_r \int_0^T dt G_{ji}(\mathbf{x}, t; \mathbf{x}_r, T - \tau) \delta u_i(\mathbf{x}_r, t). \tag{2.33}$$

We now transform the time t to $t' = T - t$ ($dt' = -dt$) and apply time reciprocity of the Green's function (equation A.4) which results in

$$\Psi_j(\mathbf{x}, T - \tau) = \sum_r \int_0^T dt' G_{ji}(\mathbf{x}, \tau; \mathbf{x}_r, t') \delta u_i(\mathbf{x}_r, T - t'). \tag{2.34}$$

Thus, the back-propagated wavefield at time $T - \tau$ is caused by the time-reversed residual wavefield δu_i acting at receiver position $\mathbf{x}_r$. In this case, the initial condition of $\Psi_j(\mathbf{x}, T) = 0$ is fulfilled. This adjoint field can be modelled with the elastic forward solver. The gradients can then be estimated as a zero-lag cross-correlation between the forward wavefield and the back-propagated wavefield (equation 2.32). The gradient of μ can be slightly reformulated to find the final symmetric gradient expressions in time domain:

$$\begin{aligned}
\frac{\partial E(\mathbf{m})}{\partial \rho(\mathbf{x})} &= - \sum_s \int_0^T d\tau \frac{\partial \Psi_j(\mathbf{x}, \tau)}{\partial \tau} \frac{\partial u_j(\mathbf{x}, \tau)}{\partial \tau} \\
\frac{\partial E(\mathbf{m})}{\partial \lambda(\mathbf{x})} &= - \sum_s \int_0^T d\tau \frac{\partial \Psi_j(\mathbf{x}, \tau)}{\partial x_j} \frac{\partial u_p(\mathbf{x}, \tau)}{\partial x_p} \\
\frac{\partial E(\mathbf{m})}{\partial \mu(\mathbf{x})} &= - \frac{1}{2} \sum_s \int_0^T d\tau \left(\frac{\partial \Psi_j(\mathbf{x}, \tau)}{\partial x_k} + \frac{\partial \Psi_k(\mathbf{x}, \tau)}{\partial x_j} \right) \left(\frac{\partial u_j(\mathbf{x}, \tau)}{\partial x_k} + \frac{\partial u_k(\mathbf{x}, \tau)}{\partial x_j} \right).
\end{aligned} \tag{2.35}$$

2.5 The gradient in frequency domain

It is also possible to perform the gradient calculation in frequency domain. This brings several advantages in FWI, which will be discussed in detail in the next chapter.
Using Fourier transforms the wavefield in time domain $\mathbf{u}(\mathbf{x},t)$ can be transformed into frequency domain $\tilde{\mathbf{u}}(\mathbf{x},\omega)$ and backwards with

$$\tilde{\mathbf{u}}(\mathbf{x},\omega) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty}\mathbf{u}(\mathbf{x},t)e^{-i\omega t}dt \qquad \mathbf{u}(\mathbf{x},t) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty}\tilde{\mathbf{u}}(\mathbf{x},\omega)e^{i\omega t}d\omega \tag{2.36}$$

Thus, we get complex valued wavefields in frequency domain. The gradient in time domain (equations 2.35) is a zero-lag correlation of the form

$$\begin{aligned}\int_{-\infty}^{+\infty}dtA(t)B(t) &= \int_{-\infty}^{+\infty}dt\int_{-\infty}^{+\infty}d\omega\frac{1}{\sqrt{2\pi}}\tilde{A}(\omega)e^{i\omega t}B(t) = \\ &= \int_{-\infty}^{+\infty}d\omega\tilde{A}(\omega)\int_{-\infty}^{+\infty}dt\frac{1}{\sqrt{2\pi}}B(t)e^{i\omega t} = \\ &= \int_{-\infty}^{+\infty}d\omega\tilde{A}(\omega)\tilde{B}(-\omega)\overset{B(t)\in\mathbb{R}}{=}\int_{-\infty}^{+\infty}d\omega\tilde{A}(\omega)\tilde{B}^*(\omega)\end{aligned} \tag{2.37}$$

Thus, in frequency domain, the zero-lag cross-correlation of two real signals A and B in time domain is replaced by a multiplication of one signal with the conjugate of the other signal integrated over the full frequency spectra in frequency domain. This is now applied to the time-domain gradient expressions in 2.35. The time integration $\int_0^T d\tau$ can be extended to $\int_{-\infty}^{+\infty}d\tau$ if the condition of $\mathbf{u}(\mathbf{x},t)=0$ and $\partial\mathbf{u}(\mathbf{x},t)/\partial t = 0$ for $t<0$ and $t>T$ is fulfilled. In the density gradient, the time derivatives each transform to frequency domain by multiplication with a factor of $i\omega$. Spatial derivatives are not affected by the transformation. This results in:

$$\begin{aligned}\frac{\partial E(\mathbf{m})}{\partial\rho(\mathbf{x})} &= \sum_s\int d\omega Re[\omega^2\tilde{u}_i(\mathbf{x},\omega)\tilde{\Psi}_i^*(\mathbf{x},\mathbf{x}_s,\omega)] \\ \frac{\partial E(\mathbf{m})}{\partial\lambda(\mathbf{x})} &= -\sum_s\int d\omega Re\left[\frac{\partial u_p(\mathbf{x},\omega)}{\partial x_p}\frac{\partial\Psi_j^*(\mathbf{x},\omega)}{\partial x_j}\right] \\ \frac{\partial E(\mathbf{m})}{\partial\mu(\mathbf{x})} &= -\frac{1}{2}\sum_s\int d\omega Re\left[\left(\frac{\partial u_j(\mathbf{x},\omega)}{\partial x_k}+\frac{\partial u_k(\mathbf{x},\omega)}{\partial x_j}\right)\left(\frac{\partial\Psi_j^*(\mathbf{x},\omega)}{\partial x_k}+\frac{\partial\Psi_k^*(\mathbf{x},\omega)}{\partial x_j}\right)\right].\end{aligned} \tag{2.38}$$

Hence, the simultaneous inversion of the full frequency spectra corresponds to the time domain inversion. However, when applying frequency domain FWI the gradients are generally only calculated for discrete frequencies (e.g. Pratt, 1999; Sirgue & Pratt, 2004; Brossier, 2011) as

$$\begin{aligned}\frac{\partial E(\mathbf{m})}{\partial\rho(\mathbf{x})} &= \sum_s\sum_{v=1}^{n_f}Re[\omega_v^2\tilde{u}_i(\mathbf{x},\omega_v)\tilde{\Psi}_i^*(\mathbf{x},\mathbf{x}_s,\omega_v)] \\ \frac{\partial E(\mathbf{m})}{\partial\lambda(\mathbf{x})} &= -\sum_s\sum_{v=1}^{n_f}Re\left[\frac{\partial u_p(\mathbf{x},\omega_v)}{\partial x_p}\frac{\partial\Psi_j^*(\mathbf{x},\omega_v)}{\partial x_j}\right] \\ \frac{\partial E(\mathbf{m})}{\partial\mu(\mathbf{x})} &= -\frac{1}{2}\sum_s\sum_{v=1}^{n_f}Re\left[\left(\frac{\partial u_j(\mathbf{x},\omega_v)}{\partial x_k}+\frac{\partial u_k(\mathbf{x},\omega_v)}{\partial x_j}\right)\left(\frac{\partial\Psi_j^*(\mathbf{x},\omega_v)}{\partial x_k}+\frac{\partial\Psi_k^*(\mathbf{x},\omega_v)}{\partial x_j}\right)\right].\end{aligned} \tag{2.39}$$

In this case, a group of n_f discrete frequencies is inverted simultaneously in each iteration. Even the use of a single frequency per iteration is possible (Sirgue & Pratt, 2004; Plessix, 2009; Brossier, 2011). The inversion with a single frequency in frequency domain corresponds to the inversion of monochromatic wavefields in time domain which are of the form

$$u_i(\mathbf{x},t) = \tilde{u}_i(\mathbf{x},\omega_v)e^{i\omega_v t}. \tag{2.40}$$

By using a discrete frequency domain FWI, information is neglected compared to time domain FWI. However, Sirgue & Pratt (2004) found, that a similar result can be achieved as long as the wavenumber spectra is covered continuously during the discrete frequency inversion. Still, an increased redundancy in the wavenumber spectra caused by denser frequency sampling or time domain FWI can mitigate the nonlinearity of the inverse problem (Brossier et al., 2009). This can especially lead to a better performance in case of complex wavefields, such as surface waves (Virieux & Operto, 2009) or complex acoustic wavefields (Kurzmann, 2012).

2.6 The seismic velocity parametrisation

The choice of parametrisation influences the performance of multi-parameter FWI due to different sensitivities and coupling of the parameter classes. So far, we calculated the gradient for the Lamé parameters λ, μ and the density ρ. Other possibilities include, for instance, the seismic-velocities-parametrisation with compressional velocity v_p, shear-wave velocity v_s and the density ρ or the parametrisation with seismic impedances and density. To study the effects of parametrisation it is possible to analyse diffraction patterns of point scatterers as a function of the scattering angle for different parametrisation (e.g. Tarantola, 1986; Forgues & Lambaré, 1997). These studies can give information about the coupling of the different parameters, and thus show how successful the inversion can separate these parameters. The best behaviour for reflection geometry was found with the seismic impedance parametrisation for small offsets and the seismic velocity parametrisation for large offsets. The Lamé parametrisation, by contrast, shows high ambiguities between its parameters especially for small offsets. In transmission geometry the best parametrisation is the seismic velocity parametrisation, however, in this case, the inversion is nearly not sensitive for changes in density. Another parametrisation study performed by Köhn et al. (2012) tested how well perturbations of single parameters included as crosses, triangles and squares in a homogeneous model can be resolved independently in a multiparameter inversion in reflection geometry. In this study they also found the best performance for seismic velocity parametrisation or parametrisation by impedances. Overall, density shows the highest ambiguity and is thus the most difficult parameter to resolve.
In this thesis I will concentrate on the seismic velocity parametrisation. The gradients for v_p, v_s and ρ' can be derived from the gradients of λ, μ and ρ (equation 2.38). The relations between the different parameters are given by

$$\begin{aligned} v_p &= \sqrt{\frac{\lambda+2\mu}{\rho}} \quad v_s = \sqrt{\frac{\mu}{\rho}} \quad \rho' = \rho \\ \lambda &= \rho'(v_p^2 - 2v_s^2) \quad \mu = \rho' v_s^2 \quad \rho = \rho' \end{aligned} \tag{2.41}$$

For the gradients of v_p, v_s and ρ' we find by applying the chain rule

$$\begin{aligned}
\frac{\partial E}{\partial v_p} &= \frac{\partial E}{\partial \lambda}\frac{\partial \lambda}{\partial v_p} + \frac{\partial E}{\partial \mu}\frac{\partial \mu}{\partial v_p} + \frac{\partial E}{\partial \rho}\frac{\partial \rho}{\partial v_p} \\
&= 2\rho v_p \frac{\partial E}{\partial \lambda} \\
\frac{\partial E}{\partial v_s} &= \frac{\partial E}{\partial \lambda}\frac{\partial \lambda}{\partial v_s} + \frac{\partial E}{\partial \mu}\frac{\partial \mu}{\partial v_s} + \frac{\partial E}{\partial \rho}\frac{\partial \rho}{\partial v_s} \\
&= -4\rho v_s \frac{\partial E}{\partial \lambda} + 2\rho v_s \frac{\partial E}{\partial \mu} \\
\frac{\partial E}{\partial \rho'} &= \frac{\partial E}{\partial \lambda}\frac{\partial \lambda}{\partial \rho'} + \frac{\partial E}{\partial \mu}\frac{\partial \mu}{\partial \rho'} + \frac{\partial E}{\partial \rho}\frac{\partial \rho}{\partial \rho'} \\
&= (v_p^2 - 2v_s^2)\frac{\partial E}{\partial \lambda} + v_s^2 \frac{\partial E}{\partial \mu} + \frac{\partial E}{\partial \rho}
\end{aligned} \tag{2.42}$$

Thus, the gradients for the velocity parametrisation can be calculated as linear combinations of the Lamé-gradients. Note, that the density gradient depends on the choice of parametrisation.

Chapter 3

Implementation of 3D FWI

In this chapter I introduce the implementation of the 3D elastic full-waveform inversion (FWI) code 3D-AWAIT (**3D**-**A**djoint **WA**veform **I**nversion **T**ool). The aim of 3D-AWAIT is the multi-parameter reconstruction of 3D subsurface structures in high resolution. Hereby, it inverts for the elastic parameters, like the wave velocities v_p and v_s and density. Anisotropy and attenuation are not taken into account. To minimise the misfit the code employs the conjugate gradient approach. Gradients are calculated using the adjoint method. The theory underlying this implementation was discussed in chapter 2.
3D elastic FWI is a very large-scaled problem and only possible due to computational advancements in recent years. Hence, the implementation has to be optimised regarding runtime and storage costs. This is realised by using a time-frequency approach where the simulation of wavefields is performed in time domain, whereas gradients are calculated in frequency domain. The characteristics of this approach, as well as the pros and cons will be discussed in this chapter.
I will start by introducing the forward solver, which is a key ingredient for a successful and efficient FWI. Afterwards the implementation of the 3D FWI is shown by means of a workflow followed by a detailed discussion of the individual inversion steps.

3.1 Forward modelling

3D elastic FWI requires a forward solver, which accurately simulates seismic wave propagation in the elastic media. Otherwise artefacts and deviations in the inversion result can be caused by fitting modelling errors in the wavefield. Additionally, a computationally efficient forward solver is required, because the inversion process is characterised by a high number of forward modellings and the FWI runtime is thus mainly influenced by the runtime spend for wavefield modelling.
In this implementation I use the elastic version of the time-domain finite-difference solver "SOFI3D" (Bohlen, 2002) for wavefield modelling, which very efficiently and fast solves the 3D elastic wave equation. The code discretizes the velocity stress formulation of the elastic wave equation on a staggered grid (Virieux, 1986; Levander, 1988). It is able to accurately model wave propagation in 3D complex media. In the following, I will give a short overview about finite-difference modelling in general and about the main characteristics of this forward

solver.

3.1.1 The finite-difference approach

To simulate wave propagation, the FD code numerically solves the elastic wave equation by approximating partial derivatives by finite differences. We use the description of wave propagation exited by a point source in the velocity-stress formulation as introduced in section 2.1:

$$\begin{aligned} \rho \frac{\partial v_i}{\partial t} &= f_i + \frac{\partial \tau_{ij}}{\partial x_j}, \\ \frac{\partial \tau_{ij}}{\partial t} - \lambda \frac{\partial \Theta}{\partial t} \delta_{ij} + 2\mu \frac{\partial \varepsilon_{ij}}{\partial t} &= 0, \\ \frac{\partial \varepsilon_{ij}}{\partial t} &= \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right). \end{aligned} \tag{2.9}$$

For the calculation of partial derivatives, the wavefield parameters and model parameters are discretised on a staggered-grid system (Virieux, 1986; Levander, 1988). The 3D staggered-grid system is sketched in Figure 3.1. We use a Cartesian grid (x,y,z)=(i,j,k). The model parameters λ, μ and ρ and the diagonal stress components σ_{ii} are localised on the full grid points, whereas velocities and off-diagonal stress components are calculated on a grid which is half a grid point shifted to the original system. Compared to a conventional Cartesian grid, the staggered grid enables a larger grid spacing for the same level of accuracy.

The derivatives are approximated by centered finite differences. The easiest finite-differences are of second order, where the two adjacent grid points with a grid space dh are used to calculate the derivative between these grid points as follows

$$\frac{\partial f(x)}{\partial x}|_{i+1/2} \approx \frac{f[i+1] - f[i]}{dh} \tag{3.1}$$

The consideration of additional grid points in higher order schemes can increase the accuracy and thus enable a larger grid spacing. However, it needs to be considered, that the number of operations increases and the parameters which need to be interchanged during MPI communication rises. Additionally, high order finite-difference operators can cause problems in case of large discontinuities, like tunnels. The SOFI code uses second order temporal and second to 12th order spatial finite difference operators, which can be chosen according to the problem.
To solve the system of equations 2.9 two steps are performed for each time step:

- velocity updates are calculated using spatial stress derivatives and added to the previous velocity values

- stress value updates are calculated using spatial velocity derivatives and Hook's law and summed to the previous stress values

Hereby the summation of the update values over all time steps accounts for the time derivatives of velocity and stress values.

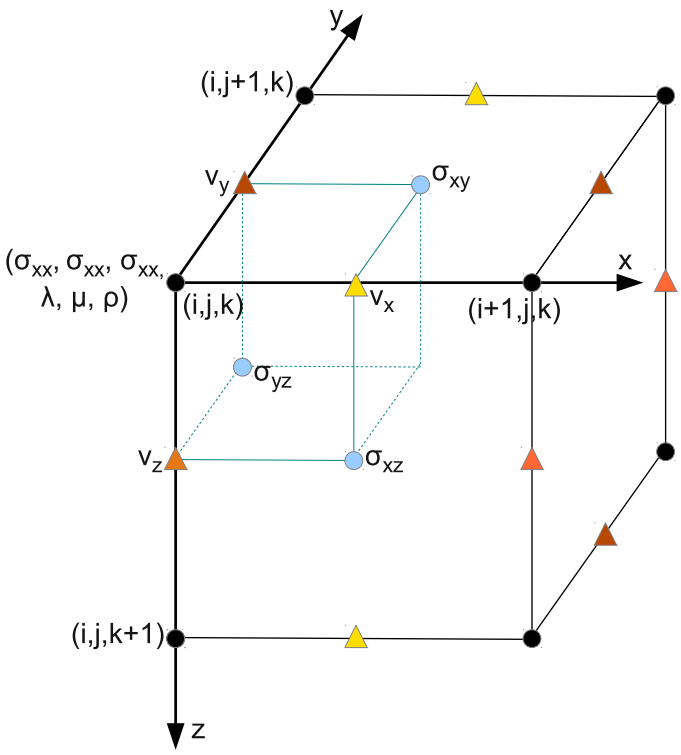

Figure 3.1: Staggered-grid coordinate system used for 3D FD modelling in SOFI3D

3.1.2 Grid dispersion and stability

To mitigate numerical dispersion and grid anisotropy of the wavefield the grid spacing must be chosen sufficiently small. For a fourth-order spatial and second-order temporal scheme the criterion

$$dh < \frac{\lambda_{min}}{6} = \frac{v_{min}}{6 f_{max}} \tag{3.2}$$

ensures a wavefield error of less than 5% (Bohlen, 2002). It depends on the minimum wavelength λ_{min}, defined by the minimum velocity v_{min} and the maximum frequency f_{max}. Higher order FD-operators enable a larger grid spacing.
In order to ensure stability of the simulation, the temporal spacing must satisfy the Courant-Friedrichs-Levy (CFL) stability criterion (Courant et al., 1967). It is related to the maximum velocity v_{max} and is given by

$$dt <= \frac{dh}{h\sqrt{3} v_{max}}. \tag{3.3}$$

The constant factor h amounts to $\frac{7}{6}$ for a Taylor operator of fourth order.

3.1.3 Boundary conditions

C-PML boundary condition

The finite-sized models used in the simulations require the implementation of boundary conditions, which prevent reflections at the boundaries travelling back into the medium. A good suppression of boundary reflections is especially important in full-waveform inversion, where waveforms need to be as accurate as possible. This is generally done by surrounding the model with a boundary area, in which waves are damped. A straightforward method is the use of absorbing boundaries, where velocities within the boundary are multiplied with an exponentially decaying factor (Cerjan et al., 1985). However, for a sufficient damping, the thickness of the absorbing boundary needs to be at least 30 grid points and the model size needs to be increased accordingly. Additionally, problems occure in case of grazing incidence of the waves.
I implemented another technique in SOFI, which is the use of convolutional perfectly matched layers (C-PMLs) (Berenger, 1994; Komatitsch & Martin, 2007), which have shown very good performance in seismic modelling. For a continuous medium, this method is characterised by a zero reflection coefficient between medium and boundary layer for all angles and frequencies. I will shortly discuss the concept of the original PMLs and of C-PMLs and refer to the work of Collino & Tsogka (2001) and Komatitsch & Martin (2007) for more details.
For the original PMLs a boundary in x-direction is considered, with a damping function of $d_x(x) > 0$ within the boundary layer. By introducing a new complex coordinate x' with

$$x' = x - \frac{i}{\omega}\int_0^x d_x(s)ds \qquad \frac{\partial}{\partial x'} = \frac{i\omega}{i\omega + d_x}\frac{\partial}{\partial x} = \frac{\partial}{s_x \partial x} \tag{3.4}$$

an exponentially decaying plane wave solution within the boundary is generated:

$$\mathbf{A} e^{-i(\mathbf{kx}-\omega t)} e^{\frac{k_x}{\omega}\int_0^x d_x(s)ds}. \tag{3.5}$$

The vector $\mathbf{k}$ denotes the wave number. This is done similarly for boundaries in y- and z-directions. In case of a continuous medium, d_i can be chosen high enough to reach strong

damping within the boundary area. After discretisation the reflection coefficient is no longer zero and very high damping functions cause reflections at the PML-boundaries. Still, it needs to be chosen large enough to ensure sufficient damping within the boundary area.
The extent of damping depends on the angle of incidence and is best for waves of normal incidence. To improve the method in case of grazing incidence, convolutional PMLs (C-PMLs) were introduced in elastodynamics by Komatitsch & Martin (2007). A stronger attenuation of waves entering the C-PML-boundary can be achieved by introducing two additional real C-PML coefficients, κ_x and α_x and rewrite s_x in equation 3.4 as

$$s_x = \kappa_x + \frac{d_x}{\alpha_x + i\omega} \tag{3.6}$$

For the implementation of the C-PML boundaries I follow the approach of Komatitsch & Martin (2007). C-PML layers are implemented by introducing memory variables for the spatial derivatives of velocities and stresses to perform the complex coordinate transformation $\partial/\partial x \rightarrow \partial/\partial x'$. Thus, additional storage for the memory variables and the vectors of the C-PML coefficients needs to be provided.
The C-PML coefficients are chosen according to Komatitsch & Martin (2007). Hereby α_i is a linear function, which is proportional to the dominant frequency of the wavefield and d_i is a quadratic function, which is proportional to the maximum velocity at the boundaries and κ_i is a scalar $>= 1$.
In Figure 3.2 I show a comparison of C-PML and absorbing boundary for a simple homogeneous model. The model contains a boundary area of 30 m thickness which equals 30 gridpoints. The source is a vertical point source excited at (x,y,z)=(40 m,40 m,40 m) and is relatively close to the boundary at all sides. Thus, waves enter the boundary area at low angles. A snapshot of the vertical wavefield is plotted for two exemplary times for the absorbing boundary in a) and b) and for the C-PML boundary in c) and d). The amplitudes are normalised to their maximum value and clipped at $\pm 20\%$ to see the boundary reflections. For the absorbing boundaries, I observe a distortion of the wavefronts at time 1 and strong boundary reflections, which reach few percent of the absolute value in the inner model at time 2. The C-PMLs by contrast show no distortion of the wavefronts and very low boundary reflections of less than 0.1% of the maximum v_z amplitude in the inner model.
This example confirms the result by Collino & Tsogka (2001) and Komatitsch & Martin (2007) who showed the very good damping abilities of the (C-)PML layers. In contrast to absorbing boundary conditions, a thickness of 10-20 grid points is sufficient for the C-PML boundary, so that the model size becomes significantly smaller, especially in 3D. Hence, even though the number of operations increases from absorbing to C-PML boundary, the computations can become more efficient.

Free surface condition

Seismic measurements are often carried out along the earth's surface and free surface conditions need to be integrated in the forward modelling. At the free surface, the vertical stress components σ_{xz},σ_{yz} and σ_{zz} are zero. This is implemented implicitly in SOFI by using the mirroring technique described by Levander (1988). The implementation only allows for planar free surfaces.

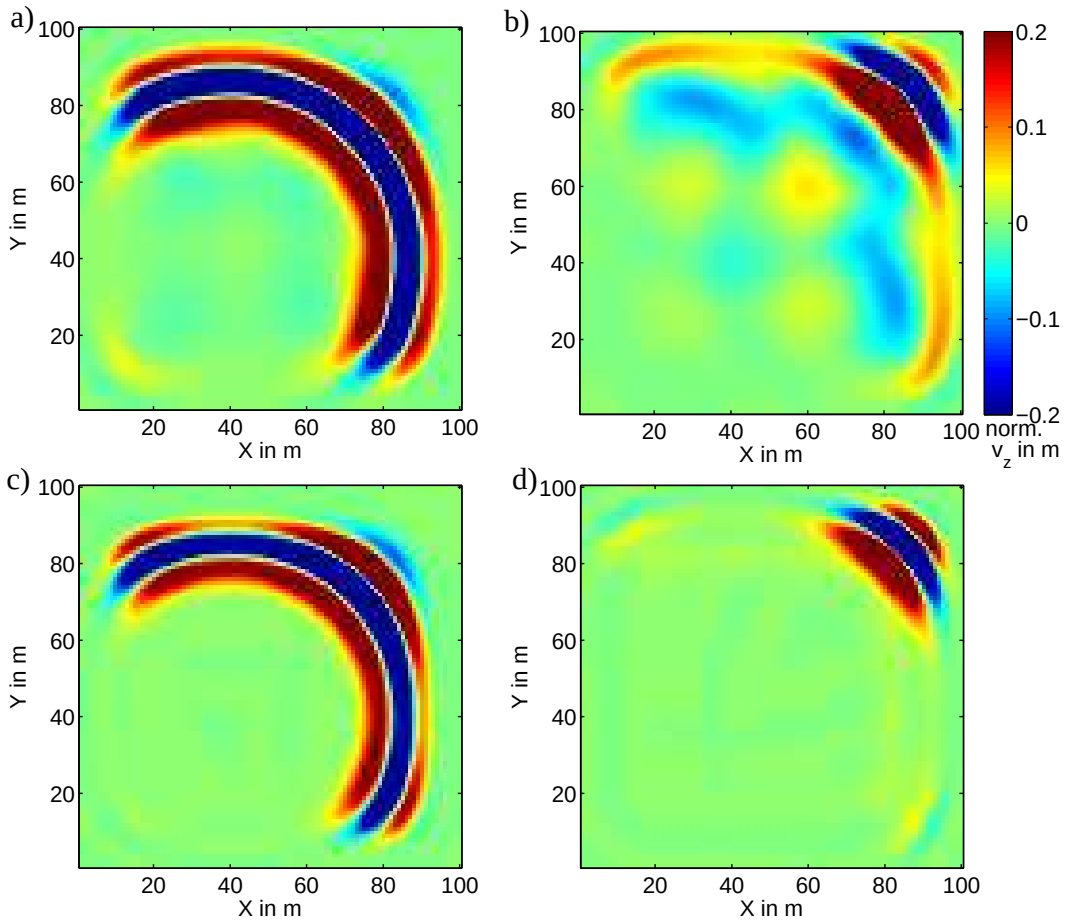

Figure 3.2: Comparison v_z snapshots of PML and absorbing boundary (ABS) for homogeneous model at $z = 40\,\mathrm{m}$, normalised to their maximum value: a) ABS at time 1, b) ABS at time 2, c) C-PML at time 1 and d) C-PML at time 2

3.1.4 Parallelisation

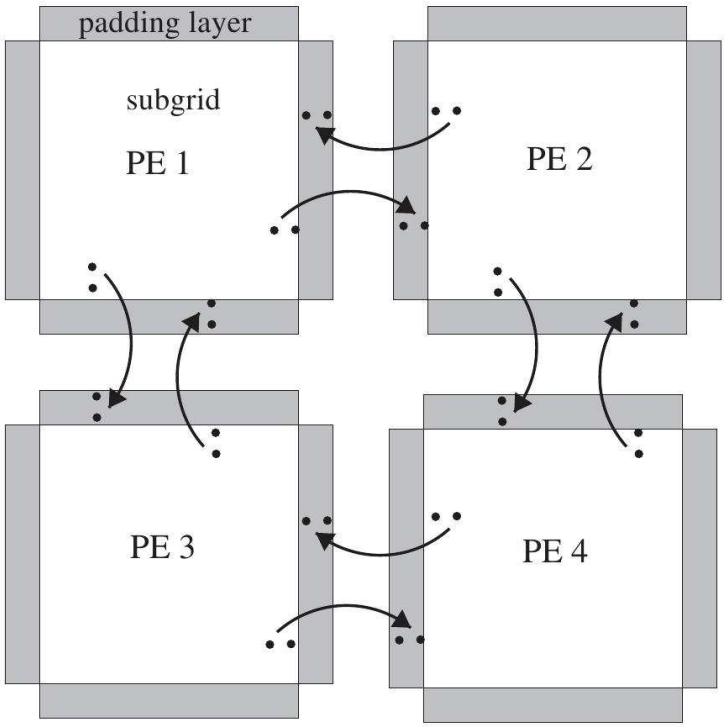

Figure 3.3: Subdivision of model area into sub-models with node to node communication within the padding areas (from: Bohlen (2002))

FWI requires a large number of forward modellings, and thus a fast and efficient forward solver. For 3D models, this cannot be provided by single processors or even on single workstations. Even for relatively small to medium sized models the inversion requires large numbers (typically several hundreds to thousands) of processors. Consequently, the code needs to perform well on large computer clusters.

The SOFI code is optimised for parallel computing architectures, using the message passing interface (MPI) library specification (Bohlen, 2002). The time-domain forward solver allows 3D domain decomposition. In each spatial direction the model is subdivided into sub-models, which are assigned to individual processors. One processor performs the wavefield calculations described in section 3.1.1 in its sub-model. To calculate the finite-difference operators for the spatial derivatives adjacent to the sub-model boundaries, the sub-models overlap in so-called padding layers. Here wavefield parameters are exchanged by node to node communication between the processors (Figure 3.3). The thickness of these padding layers depends on the order of FD operator. In case of 4th spatial order, the sub-models are extended by two grid points. Of course, exchange times increase with increasing order of the FD operators, which needs to be considered for higher order FD-modelling.

The domain decomposition is maintained during the whole inversion process and thus enables a good performance of 3D-AWAIT on large computer clusters like the JUROPA cluster in Jülich and the Cray XE6 at HLRS in Stuttgart.

3.1.5 Single frequency transformation of wavefields

In 3D-AWAIT, I implemented the gradient calculation in frequency domain, even though I apply the time-domain forward solver SOFI. The gradient calculation is performed for few discrete frequencies and requires the frequency-domain forward and adjoint wavefields for these frequencies. I follow the approach by Sirgue et al. (2008) and use the discrete Fourier transformation

$$\tilde{A}_i(\mathbf{x},\omega) = \sum_{l=0}^{nt} \exp(i\omega l\Delta t) A_i(\mathbf{x}, l\Delta t)\Delta t, \tag{3.7}$$

to extract the monochromatic wavefields, where nt is the number of timesteps, Δt the time sampling and A_i a wavefield parameter. The discrete Fourier transformation is performed on the fly, summing up Fourier summands of A_i in each time step. Therefore, no storage of the time domain wavefields at different time steps is required. Furthermore, the number of additional operations is low, as long as the transformation is only performed for few frequencies. Thus, the change from time to frequency domain can be achieved at low extra costs.

3.1.6 Filtering of seismograms

The inversion strategies can require the filtering of wavefields and seismograms. To implement this I added the *cseife* library written by Erhard Wielandt. Filtered wavefields are generated by applying a filter to the source wavelet. Additionally seismograms, like the observed data can be filtered directly within the inversion. The *cseife* library offers different kinds of filters, including lowpass, highpass and bandpass butterworth filters of diverse orders. As a standard, the lowpass butterworth filter of 4th order is used in 3D-AWAIT.

3.2 Workflow

The workflow shown in Figure 3.4 should provide an overview of the 3D-AWAIT inversion code. The steps shown in the workflow are performed within each iteration k. The input in each iteration is the model of the previous iteration $\mathbf{m}_{k-1}$, which corresponds to the starting model in the first iteration, and the observed data. Using this input, the gradient of the misfit function is computed, preconditioning is applied and the conjugate gradient direction is calculated. Finally a line search for the step length calculation is performed and the model is updated. The following sections will provide a detailed descriptions of the individual steps.
The different steps of the inversion will be illustrated by means of a simple toy example, which I shortly introduce here. The model is small with $100 \times 100 \times 120$ grid points in x-, y- and z-direction, respectively and a grid spacing of 1 m. A box of the size of $30\,\text{m} \times 25\,\text{m} \times 20\,\text{m}$ with increased velocity v_p and v_s and density ρ is inserted into homogeneous media, as can be seen in Figure 3.5. The acquisition geometry is a transmission geometry with nine sources and 225 receivers each arranged in a x-y-directed plane as indicated in the plot. The sources are excited as vertical point forces with a dominant frequency of 200 Hz. I use a $\sin^3$ source wavelet.
I used the homogeneous background velocities of $v_p = 6000\,\text{m/s}$, $v_s = 3200\,\text{m/s}$ and a density of $\rho = 2800\,\text{kg/m}^3$ as starting models. The inversion with a total of 43 iterations is separated into four different frequency stages applying five frequencies per stage as given in Table 3.1.

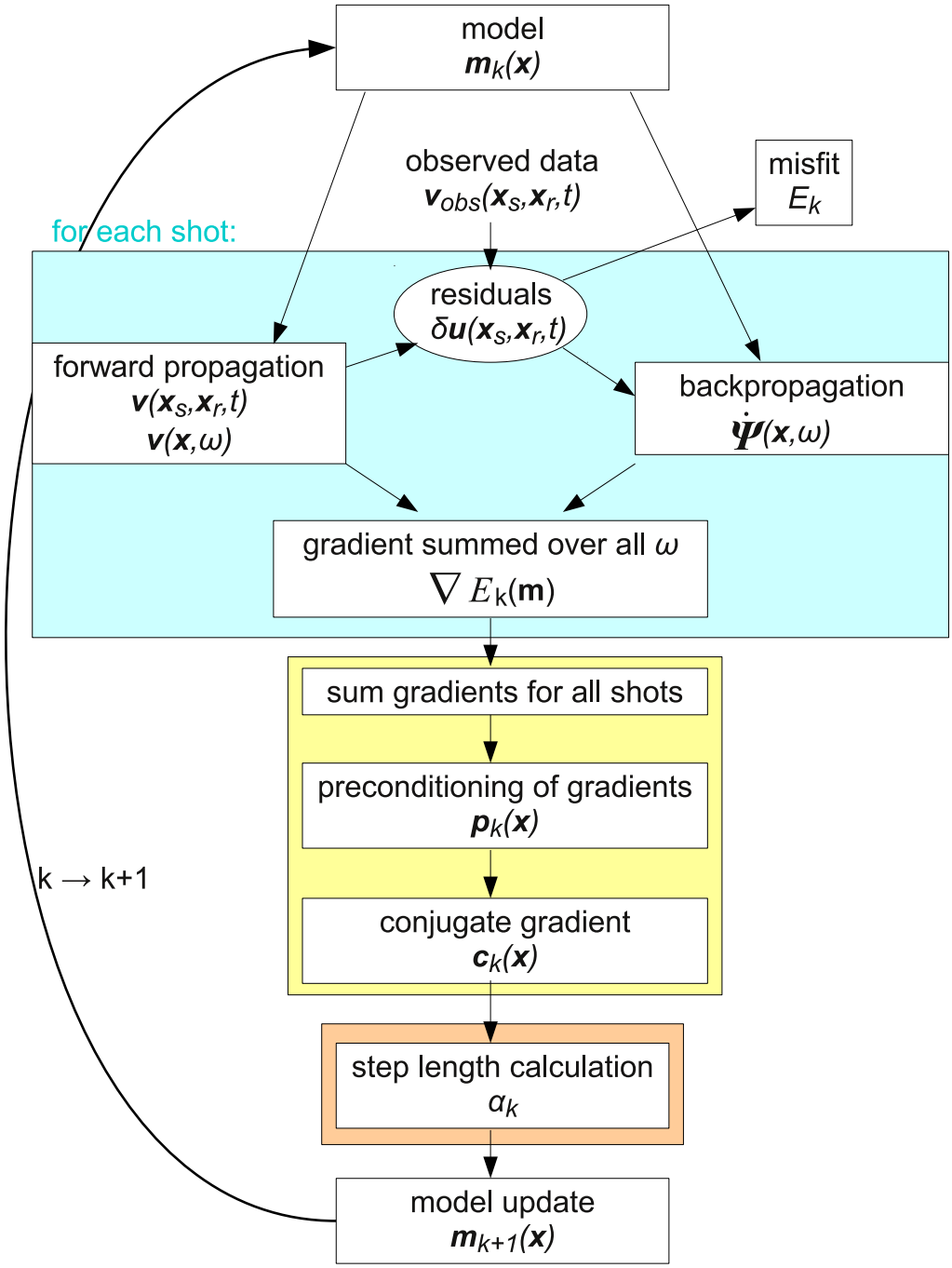

Figure 3.4: The workflow of 3D-Await shows the different steps performed during each iteration.

iteration	frequencies in Hz
1-15	140,150,160,170,180
16-25	180,190,200,210,220
26-35	220,230,240,250,260
36-43	260,270,280,290,300

Table 3.1: Frequency stages used for the toy example.

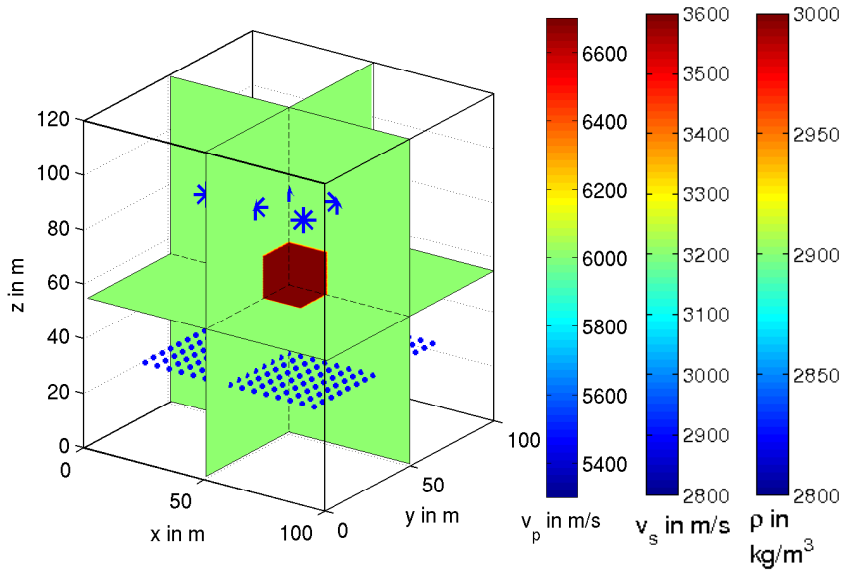

Figure 3.5: A simple box model is used as a toy example. Stars indicate the source positions and crosses indicate the receiver positions.

3.3 The gradient direction

In this section I will introduce the implementation of the gradient calculation in 3D-AWAIT, as sketched in the blue box of the workflow (Figure 3.4). The calculation of the gradient is based on the adjoint method, which theory is explained in section 2.4. The gradients are estimated for the model $\mathbf{m}_k$ in each iteration k. I use a self-adjoint second-order state equation for the inversion algorithm, but calculate the wavefields with the velocity-stress forward solver SOFI3D as introduced previously. This needs to be considered in the implementation.
The implementation is based on a time-frequency approach calculating wavefields in time domain, but performing the gradient calculation in frequency domain as described by Sirgue et al. (2008) and Sirgue et al. (2010b). The advantages of this approach will be discussed in section 3.6. During the forward and backpropagation of wavefields in time domain, monochromatic wavefields are extracted on the fly as described in section 3.1.5. The gradient can then be calculated as a multiplication of forward and conjugate back-propagated wavefield in frequency domain (see equation 2.39). In the following I will describe the steps of gradient calculation in more detail.

3.3.1 Wavefield calculation

For each shot, two forward modellings are performed for the gradient calculation. In a first step, the forward-propagated wavefield is simulated, which travels from source into medium. At each receiver position the seismograms $v(\mathbf{x}_r,t)$ are stored. Additionally, by applying a discrete Fourier transformation (equation 3.7), I calculate the particle velocities $\tilde{v}(\mathbf{x},\omega)$, which are stored in memory for each discrete inversion frequency and at each grid point. At the receiver positions, the L_2-norm based misfit between the modelled seismograms and the observed seismograms is calculated according to equation 2.18 from the displacement values $u_i(\mathbf{x}_r,t)$. The displacements at time step t_p are calculated from the velocities by discrete integration as $u_i(\mathbf{x}_r,t_p)=\sum_1^{t_p} v_i(\mathbf{x}_r,t_p)\Delta t$, where Δt is the time spacing.
The back-propagated wavefield $\Psi(\mathbf{x},T-t)$ was defined in equation 2.34. As a source-time function, the time-reversed displacement residual $\delta u_i(\mathbf{x}_r,T-t')$ is calculated at the receiver positions. This residual is back-propagated from all receivers into the medium simultaneously. Hereby I use the same forward solver, which is based on the first-order wave-equation and therefore calculates the time derivative $\partial\Psi/\partial t=\dot{\Psi}$. For the backpropagation, the time line is reversed and I calculate wavefields starting from time T backwards. Same as for the forward propagation, I extract the frequency wavefields $\tilde{\dot{\Psi}}(\mathbf{x},\omega)$ on the fly. These frequency wavefields are stored in memory for each frequency.
Exemplary frequency domain wavefields for one source-receiver combination of the box model are shown in Figure 3.6. The figure shows the real part of the normalised z-component of the forward- and back-propagated wavefields for 140 Hz (a, b) and 200 Hz (c, d) in the y=50 m plane, which cuts through the source and receiver locations. The forward-propagated wavefield, excited at the source (a, c) shows the frequency domain wavefield propagation in homogeneous medium. The higher frequencies in c) compared to a) are clearly visible. The back-propagated wavefield (b, d) excited at the receiver contains the information about the data residual at this receiver.

3.3.2 Gradient calculation

For each shot the gradients are calculated using the stored forward wavefield $\tilde{v}(\mathbf{x},\omega)$ and back-propagated wavefield $\tilde{\dot{\Psi}}(\mathbf{x},\omega)$ according to the equations 2.39. It would also be possible to reformulate this expressions and to calculate gradients from stresses and particle velocities, which is often used, when using a velocity-stress forward solver (e.g. Shipp & Singh, 2002; Castellanos et al., 2011; Köhn et al., 2012). However, in this case, velocity and stress components need to be stored. The calculation of gradients from particle velocities only (Brossier, 2011) has the main advantage that only three wavefield components instead of nine need to be stored. Hereby the gradient of ρ can be calculated directly as multiplication of velocity components $\tilde{v}_i$ and $\tilde{\dot{\Psi}}_i$. To derive displacements $\tilde{u}_i$ and $\tilde{\Psi}_i$ as required for the computation of the gradients of λ and μ, the frequency domain enables a simple integration with

$$\tilde{u}_i(\mathbf{x},\omega)=\frac{1}{i\omega}\tilde{v}_i(\mathbf{x},\omega). \tag{3.8}$$

The spatial derivatives of $\tilde{u}_i(\mathbf{x},\omega)$ and $\tilde{\Psi}_i(\mathbf{x},\omega)$ are calculated with the use of finite differences. To finally find the total gradient, the gradients for each frequency and each source are summed

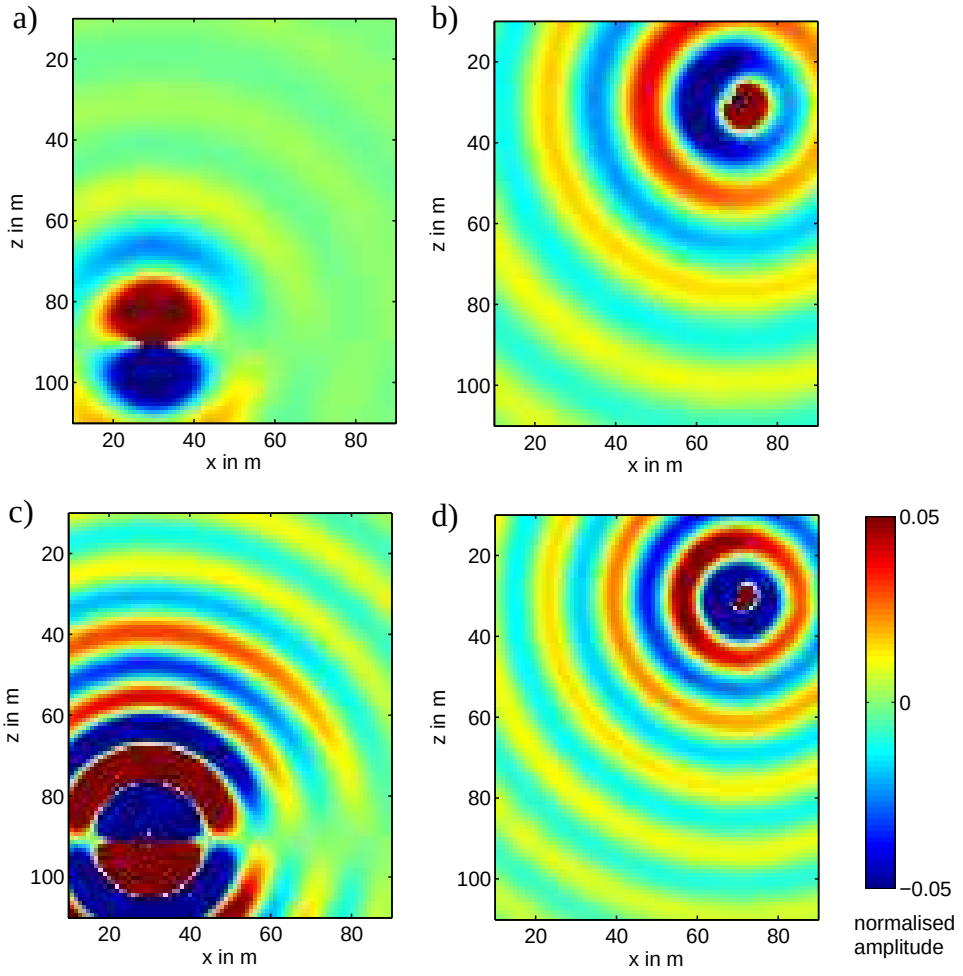

Figure 3.6: Real parts of normalised forward- and back-propagated wavefields (z-component) in frequency domain for source at (x_s, y_s, z_s)=(30 m, 50 m, 90 m) and receiver at (x_r, y_r, z_r)=(70 m, 50 m, 30 m) shown at y=50 m: a) forward wavefield ($\tilde{v}_z$) at 140 Hz, b) back-propagated wavefield ($\tilde{\Psi}_z$) at 140 Hz, c) $\tilde{v}_z$ at 200 Hz and d) $\tilde{\Psi}_z$ at 200 Hz.

up. A change to the seismic velocity parametrisation is implemented, using the chain rule (equations 2.42).

Figure 3.7 shows the normalised gradients for 140 Hz and 200 Hz calculated for the source-receiver combination, which wavefields I showed in Figure 3.6. The v_p gradient at 140 Hz shows the largest wavelengths. The shorter wavelengths of the v_p gradient for the higher frequency of 200 Hz can be clearly seen in b). The wavelengths in the v_s gradients (c,d) are much smaller, which results in the higher nonlinearity but better resolution of the v_s compared to v_p at a given frequency. The density is the most difficult parameter to resolve, which also shows in the somehow messy structure of its gradients (e,f). The box is not yet recognisable in the gradients but becomes visible when summing up the gradients over all source-receiver pairs.
Figure 3.8 presents the total gradients for the first iteration (first frequency stage) normalised to their maximum value. The zones of alternating amplitudes seen in the single source-receiver gradients (Figure 3.7) now vanished in the summed velocity gradients (a,b). In the area of the

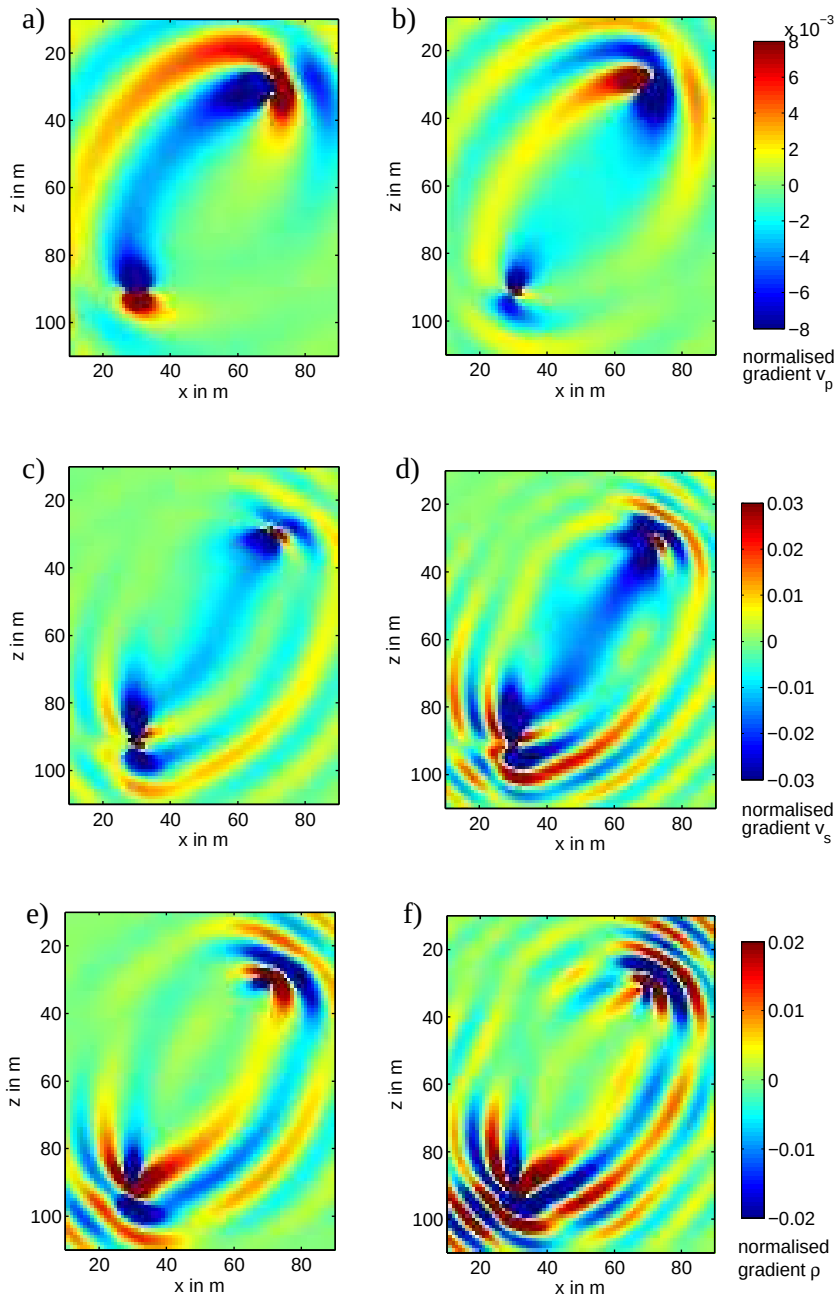

Figure 3.7: Normalised gradients of box model FWI for one source-receiver pair at (x_s, y_s, z_s)=(30 m, 50 m, 90 m) and (x_r, y_r, z_r)=(70 m, 50 m, 30 m) shown at y=50 m for two exemplary frequencies: a) gradient v_p at 140 Hz, b) gradient v_p at 200 Hz, c) gradient v_s at 140 Hz, d) gradient v_s at 200 Hz, e) gradient ρ at 140 Hz and f) gradient ρ at 200 Hz.

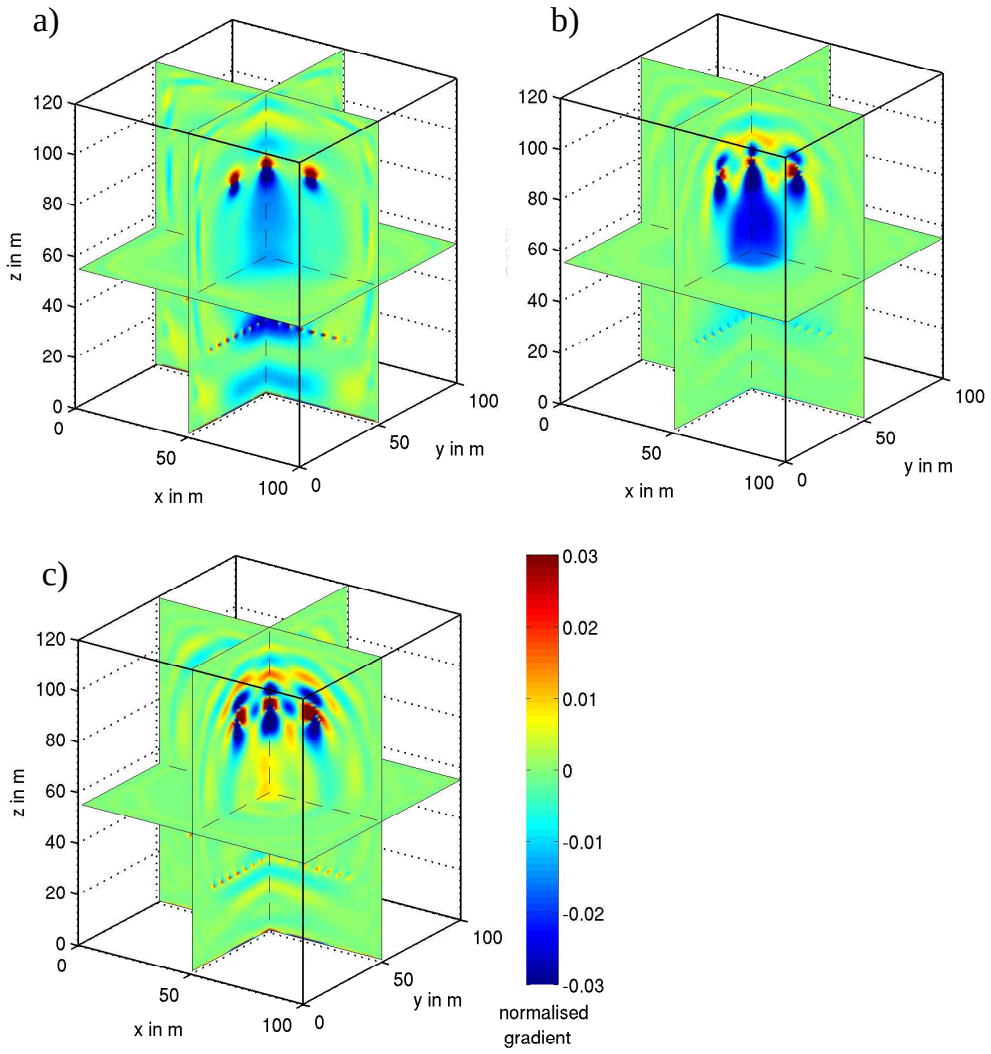

Figure 3.8: The total gradients for the first iteration of the box model normalised to their maximum values, a) gradient v_p, b) gradient v_s and c) gradient ρ

box, the negative gradient values are clearly visible. However, they amount to only 2-3% of the maximum values. The highest amplitudes are found around source and receiver positions, which clearly shows the need for preconditioning. Overall, the v_p gradient is much smoother due to the larger wavelength of the P-wave velocity than the v_s gradient. The density gradient (c) is still very messy, and does not show low density amplitudes in the box area. These difficulties of the density inversion will also be seen in the final inversion result shown later.

3.4 Preconditioning

The gradients are calculated from the forward- and back-propagated wavefields. In a 3D full space, their amplitudes decay with $1/R$ due to geometric spreading, where R is the distance to the source. This amplitude decay of the wavefields becomes noticeable in the gradients, which show high amplitudes around source and receiver positions. Without preconditioning, the model update would thus focus to the area next to sources and receivers, and the inversion fails. A thorough preconditioning which mitigates geometric effects in the gradients is therefore significant for a successful inversion.
A local approach is the damping of the gradient at and adjacent to source and receiver positions by applying spherical tapers at these positions or by tapering the area around the whole source or receiver plane. I use exponential functions of the form

$$D_1(\mathbf{x}) = (1 + ae^{-br})^{-1} \quad \text{or} \quad D_2(\mathbf{x}) = (1 + ae^{-br^2})^{-1}. \tag{3.9}$$

The parameter r is the distance to the source position ($|\mathbf{x} - \mathbf{x}_s|$) or receiver position ($|\mathbf{x} - \mathbf{x}_r|$). For a damping around receiver and source plane, r denotes the distance to these planes. The functions D are characterised by small values adjacent to $\mathbf{x}_s$ or $\mathbf{x}_r$ but approach the value one for large r. Hereby the positive value a defines the minimum value of D, whereas b defines the taper radius, as can be seen in Figure 3.9. Here the functions D_1 and D_2 are plotted for a constant $a = 1000$ and different values of b. The function D_1 allows a smooth taper over a larger distance, whereas the function D_2 is a sharper taper, which is especially suited for very local tapering. In practise, the latter taper can often be used to damp receiver artefacts, which generally extend only few grid points, whereas the D_1 taper is useful for elimination of the more extended source artefacts.
For preconditioning, one damping function is estimated for sources $D_s(\mathbf{x})$ and receivers $D_r(\mathbf{x})$

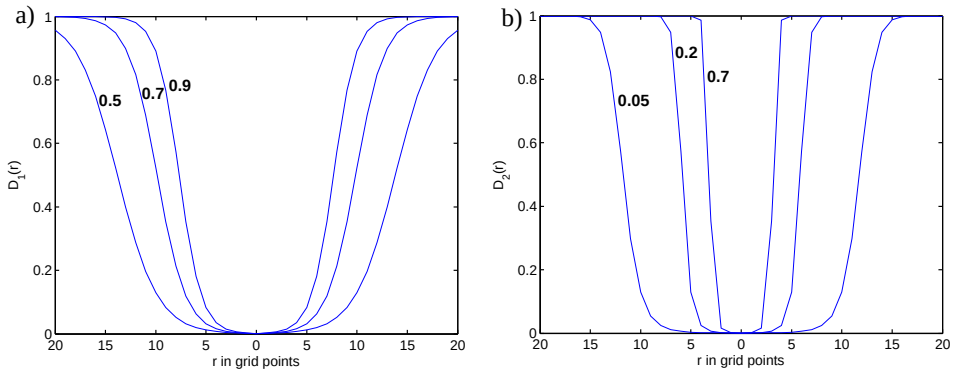

Figure 3.9: Taper functions D_1 (a) D_2 (b) with $a = 1000$ for a source or receiver located at zero distance plotted for different values of b.

each and multiplied to the gradient at each grid point. This damps artefacts locally around source and receiver positions and leaves the remaining gradient unchanged.
The FWI tends to generate artefacts within the C-PML boundaries. Therefore, a tapering of the C-PML boundaries $D_{PML}(\mathbf{x})$ is implemented using the D_2-function, where the distance to the model boundary is used for r.

The total preconditioned gradient can then be estimated as

$$\mathbf{p}_k(\mathbf{x}) = D_s D_r D_{PML} \nabla E_k. \tag{3.10}$$

This local preconditioning technique was applied to the gradients of the box model (Figure 3.8).

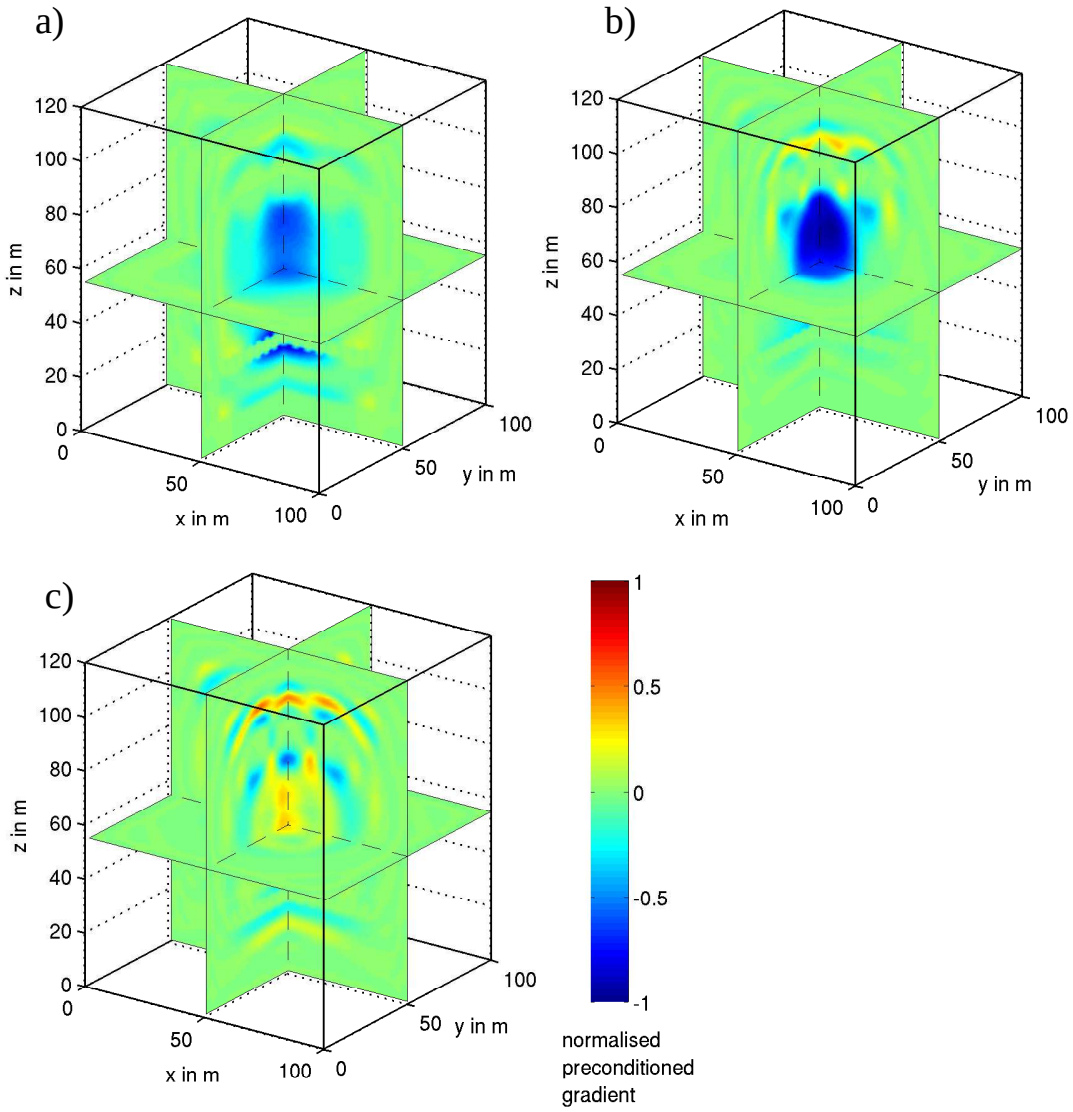

Figure 3.10: The preconditioned gradients for the first iteration of the box model normalised to their maximum values, a) gradient v_p, b) gradient v_s and c) gradient ρ

Hereby the D_1-function with $a = 1000$ and $b = 0.7$ was used for the damping of the sources, and the D_2-function with $a = 1000$ and $b = 0.7$ was applied to damp the local receiver artefacts. Furthermore, the 10 gridpoints C-PML-boundary was tapered with the D_2-function ($a = 1000$ and $b = 0.05$). The preconditioned, normalised gradients are plotted in Figure 3.10. Here the high amplitudes around sources and receivers are removed and the model update of the velocity gradients (a, b) concentrates on the area between source and receiver plane, where the box is

located. The density gradient (c) is still dominated by artefacts.
A main disadvantage is that we get no significant model update next to source and receiver positions. This can lead to problems if the model parameters in source and receiver areas are unknown. Additionally, this local preconditioning only accounts for the high amplitudes around sources and receivers, but does not correct for the geometrical spreading in the remaining model. This is generally sufficient for transmission geometry applications. However, for reflection geometries, where surface waves propagate along the free surface, improved preconditioning methods can be required. A more sophisticated approach is the concept of Hessian preconditioning, which accounts for the geometrical amplitude effects in the gradient. This method will be discussed in detail in chapter 5.

3.5 The model update

In the conjugate gradient method the model in iteration k is updated with $(-\alpha_k \mathbf{c}_k)$ (section 2.3). Hereby the direction of the model update is given by the negative conjugate gradient $\mathbf{c}_k$, whereas the steplength α_k defines its size.

3.5.1 The conjugate gradient

The conjugate gradient direction is estimated from the preconditioned gradient of the current iteration ($\mathbf{p}_k$) and the preconditioned gradient ($\mathbf{p}_{k-1}$) and conjugate gradient ($\mathbf{c}_{k-1}$) of the previous iteration according to the equations 2.15 and 2.16. Thus, the application of the conjugate gradient method requires the storage of $\mathbf{p}_{k-1}$ and $\mathbf{c}_{k-1}$. In the first iteration and after a change of frequencies the preconditioned gradient is used for model update and I define $\mathbf{c}_k = \mathbf{p}_k$.
The conjugate gradient does not offer information about the size of the model update and about the relative model update of the different parameter classes compared to each other. I therefore normalise $\mathbf{c}_k$ to its maximum for each parameter and scale them by multiplication with some average of the respective model parameter, i.e.

$$c_k^{v_p}(\mathbf{x}) \rightarrow \frac{c_k^{v_p}(\mathbf{x})}{max(c_k^{v_p}(\mathbf{x}))} v_p^0 \qquad c_k^{v_s}(\mathbf{x}) \rightarrow \frac{c_k^{v_s}(\mathbf{x})}{max(c_k^{v_s}(\mathbf{x}))} v_s^0 \qquad c_k^{\rho}(\mathbf{x}) \rightarrow \frac{c_k^{\rho}(\mathbf{x})}{max(c_k^{\rho}(\mathbf{x}))} \rho^0 \tag{3.11}$$

3.5.2 Steplength calculation

In Newton or Gauss-Newton methods, the Hessian operator accounts for the size of the model update and steplengths close to one can be expected. By contrast, gradient methods require a good estimation of α_k to reach a good convergence of the inversion. In spite of this, I expect the line search for α_k to be relatively cheap and do not want to afford too many forward modellings for the steplength estimation. I decided to use a parabola method (e.g. Kurzmann et al., 2009) which is relatively cheap, but offers reasonable results.
Hereby I update the model with two test steplengths and estimate the corresponding misfit values for a subset of shots (N_s(step)). Additionally to the misfit value of the current iteration with steplength zero, these values define a parabola. Its minimum gives us some optimal steplength. Thus, this method requires $(2 \times N_s(\text{step}))$ additional forward modellings.

The parabola method only offers a rough approximation and several problems can occure and need to be accounted for in the implementation. I defined the following conditions:

1. in case, the parabola extremum is maximum, I use the test steplength with the smallest misfit

2. a maximum steplength is defined, which cannot be exceeded

3. if a steplength of zero is estimated or the steplength is negative, the model is updated using a small ratio of the first test-steplength and a warning is given

Of course, the choice of the test steplengths is crucial. They can generally be larger at the beginning of the inversion to estimate the rough model, whereas smaller steplengths are essential in the later part of the inversion, when smaller details are reconstructed. A model update of few percent of the scaled gradients (equation 3.11) is generally a good choice at the start of the inversion, which can be decreased down to few per mill during the inversion process.
Only one steplength is estimated simultaneously for all parameter classes, which I invert for. Hereby, the scaling of the gradients (equation3.11) accounts for the different update sizes. Additionally it can be sensible to decrease the model update for some parameter classes by using only a ratio of the (test) steplength in the model update.
After the estimation of α_k the model can finally be updated and I proceed to the next iteration.

3.6 Time-frequency method

3D elastic full-waveform inversion is computationally very demanding and its applications are still limited by long runtimes and high storage costs. Thus, its implementation needs to account for these issues, to achieve an optimal computational performance. The time-frequency approach offers great advantages for 3D elastic waveform inversion which arise from modelling of 3D wavefields in time domain but performing the inversion for discrete frequencies in frequency domain.
I use a time-domain finite-difference forward solver, which is computationally more efficient in 3D compared to a frequency-domain solver (Nihei & Li, 2007; Sirgue et al., 2008). Operto et al. (2007) found, that direct frequency modelling requires $O(n^6)$ operations and a high memory consumption of $O(35n^4)$. By contrast, a finite-difference time-domain forward solver uses $O(n^3 N_s N_t)$ operations and less storage space of only $O(n^3)$. Hereby, n^3 is the number of gridpoints, N_s the number of sources and N_t the number of timesteps. The lower number of operations and less storage combined with the efficient parallelisation of the time domain solver makes it very attractive.
The calculation of wavefields in time-domain also offers the advantage of straightforward time-windowing. Hereby, only data within a particular time window are inverted simultaneously. This is not yet implemented in 3D-AWAIT.
However, gradient calculations in time domain (equations 2.35) are performed during back-propagation, but need the information about the full forward wavefield to perform the zero-lag cross-correlation between forward- and back-propagated wavefield. Even though time stepping for inversion can be chosen larger than required for modelling, the storage of these wavefields

takes a considerable amount of storage space (Kurzmann, 2012). By contrast, for the inversion in frequency domain only a few discrete frequencies per iteration can be sufficient for the model reconstruction and wavefields are stored only for these frequencies. This leads to a storage decrease of two to three orders of magnitude compared to time-domain FWI and significantly affects the performance of 3D FWI. Other approaches to decrease storage costs in time domain include the recalculation of the wavefield from values stored at checkpoints (Symes, 2007) or the reconstruction of the forward wavefield during backpropagation from the last wavefield snapshot and wavefield values stored along the boundary (Gauthier et al., 1986). The main drawback of these approaches is the increased number of operations.
Another advantage in frequency-domain inversion is that it is straightforward to successively invert higher frequencies, which mitigates the cycle skipping problems. This so-called multi-scale approach will be discussed in section 3.7.1. When performing a frequency inversion, the wavefields used for gradient calculation are automatically filtered by the choice of frequencies per iteration. Still, in our implementation, the misfit is calculated in time domain, so that additional filtering of the seismograms is required to get a representative misfit value.
The main drawback of the discrete frequency domain inversion is the loss of redundancy compared to time-domain FWI which can lead to problems in case of highly nonlinear problems such as surface waves (Virieux & Operto, 2009) or complex acoustic wavefields (Kurzmann, 2012). Of course, this can be mitigated by using a denser frequency sampling. However for high numbers of frequencies per iteration, the storage space increases and the higher number of discrete Fourier transformations slows the inversion down. Thus, the time-frequency method becomes less efficient. Still, successful applications showed that the discrete frequency inversion in general offers very good results for a reasonable number of 3-5 frequencies per iteration (e.g. Pratt, 1999; Sirgue & Pratt, 2004; Brossier et al., 2009).

3.7 The full inversion process

So far I described the different steps to estimate the model update in one iteration, i.e. to calculate the final conjugate gradient direction and the step length. This section will give an overview about the whole inversion process. The seismic inverse problem is highly nonlinear and requires an iterative approach to fit the modelled and observed data. Consequently, the previously explained steps are repeated for a number of iterations until the inversion converges to sufficiently low misfit values. In this case a model is found which explains the observed data well.
As a local inversion method, FWI bears the danger to end up in one of the numerous local minima of the objective function. It is possible, that the final model does explain the observed data well, even though it is not the correct model. To mitigate this problem, a better starting model can be required. Additionally, different inversion strategies, like the multi-stage inversion can help to avoid the local minima and consequently be required for a successful inversion. The multi-stage inversion will be explained in the first part of this section.
Another important aspect of the elastic FWI is the multi-parameter inversion, which inverts for several parameters simultaneously. Different issues of this topic will be discussed in the second part of this section.

3.7.1 Multi-scale inversion

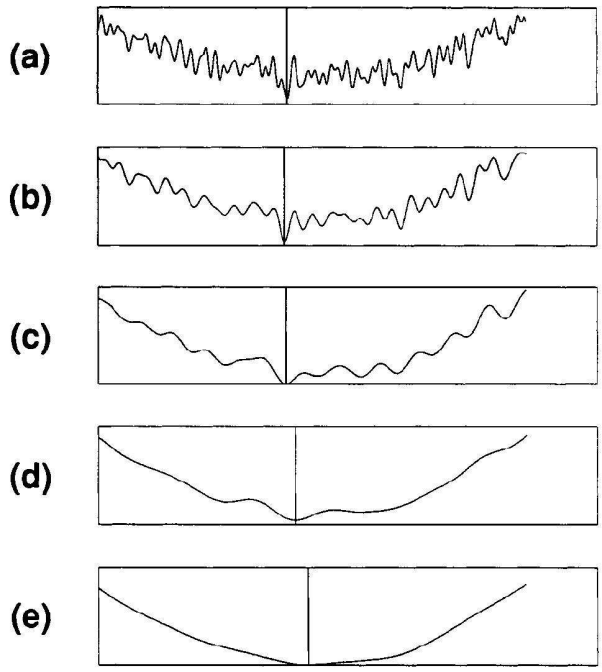

Figure 3.11: Illustration of the misfit function for scale lengths increasing from a) to e) [from: Bunks et al. (1995)].

In the so-called multi-stage approach we do not invert all data simultaneously, but separate the inversion in stages, in which the data is limited to some frequency band, to a certain time window or to a particular offset range. This can help to mitigate the nonlinearity of the inversion and to improve the result. Of particular importance is the multi-scale approach, in which the inversion is divided into different frequency stages. Hereby we start from low frequencies and increase the frequencies during inversion (Bunks et al., 1995). Especially for more complex problems, this strategy is essential to reach the global misfit minimum. The main reason is that the form of the misfit function is strongly influenced by the frequency content of the data. This is well illustrated in Figure 3.11 from Bunks et al. (1995). The misfit function is plotted for different scale lengths, which increase from a) to e). With increasing scale, and thus with decreasing frequency, the misfit function becomes smoother and the number of local minima decreases. Therefore, to reach the global minimum, a model can be much farther away from the real model for low frequencies, whereas a much closer model is required for high frequencies. This can also be understood when looking at the data. Phase differences between the modelled and the observed data need to be smaller than half a period, so that the phases can be assigned correctly to each other. Otherwise cycle skipping can occure and the inversion might end up in a local minimum. This condition is easier to fullfill for lower frequencies. Hence, a multi-scale approach in FWI can mitigate the problem of cycle-skipping and help to reach the global minimum.

Starting model and lowest frequency

Still, the starting model data needs to be close enough to the real data in the first frequency stage. Hence, the starting model and the starting frequency are crucial for a successful FWI. This can especially cause problems in field data applications, where the lowest frequency is defined by the frequency content of the data. In these applications, the starting frequency generally corresponds to the lowest available frequency (Warner et al., 2013). Furthermore, in frequency inversion it needs to be considered that frequencies below this starting frequency are not used in inversion, even if lower frequency data is available. In general, less information in the starting model, like a smoother starting model means, that lower starting frequencies are required. In my synthetic applications I choose the starting frequencies low enough that the frequency content below the starting frequency in the data is explained for the most part. In transmission geometry FWI is able to resolve structures down to one wavelength. Thus, it is possible to use a homogeneous starting model, when starting from frequencies, which correspond to a wavelength of the size of the largest structures.

Frequency intervals

The choice of the frequency interval needs to ensure a continuous coverage of the wavenumber spectra. Sirgue & Pratt (2004) showed, that this can generally be fulfilled for frequency intervals larger the sampling interval of $1/T$ (T: length of time series), when considering not only one trace but a range of offsets. Hereby larger offsets enable the coverage of the wavenumber spectra with a lower frequency sampling. For a simple 1D model, the relevant frequencies can be calculated with

$$f_{n+1} = \frac{f_n}{\alpha_{min}}, \tag{3.12}$$

where α_{min} depends on the offset-to-depth ratio and is smaller for larger maximum offsets (Sirgue & Pratt, 2004). Still, the use of additional frequencies increases the redundancy of the data which generally results in improved images, especially in case of higher nonlinearity of the inversion.

Frequency stages

The multi-scale inversion is separated into different frequency stages, in which one to few frequencies are used. Brossier et al. (2009) tested three different approaches:

- sequential approach: single frequency inversion using one frequency at a time
- Bunks approach: starting from one low frequency, one higher frequency per stage is added, so that the number of frequencies per stage increases
- simultaneous approach: a group of typically 3-5 frequencies is inverted in each stage, with overlapping frequency bands

Overall, they found the best performance for the simultaneous inversion of several frequencies, with overlapping frequency bands for their reflection geometry application. Thus, the entrainment of the lower, already inverted frequencies in the Bunks approach does not seem to be

improve the inversion. Pratt (1999) also showed an improved performance, when using five instead of three frequencies during one frequency stage. Thus a higher number of frequencies can improve the inversion and a single frequency approach does not seem to be favourable, even though successful applications show that single frequency inversions are possible.
Fortunately, the use of frequency groups with few frequencies per stage instead of a single frequency in a time-frequency FWI does not increase the computational costs by much. By contrast, when applying a frequency domain forward solver, these frequencies need to be simulated additionally.

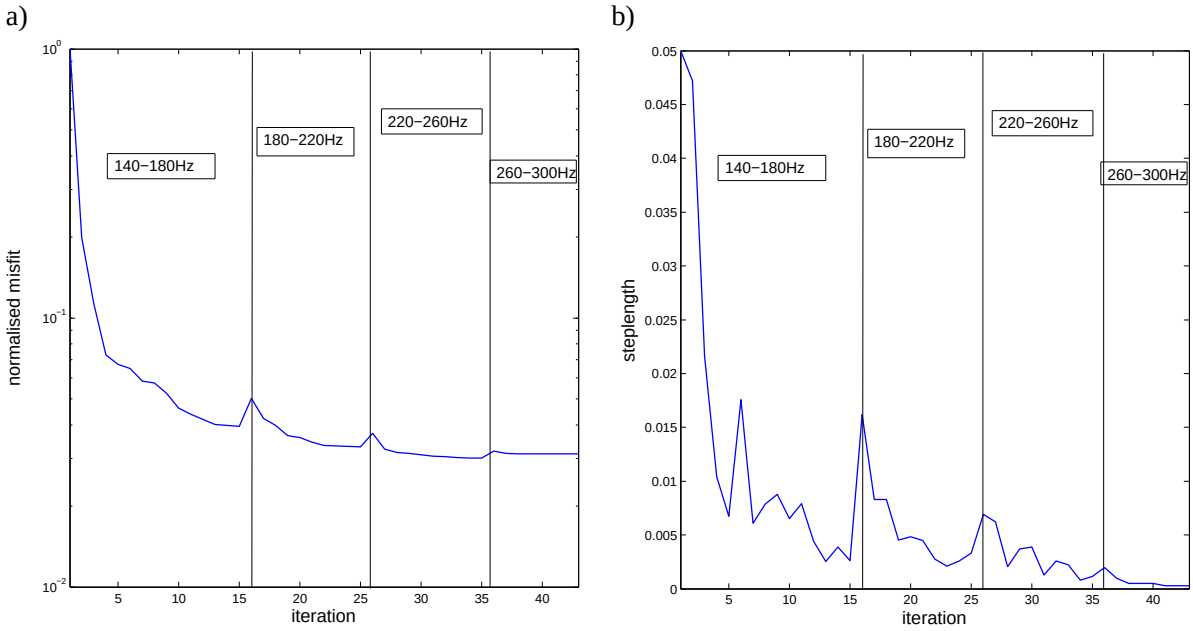

Figure 3.12: a) Misfit behaviour of the toy example shows convergence within the four frequency stages and b) steplength used for update plotted for each iteration.

Inversion process of the toy example

For the toy example, I used four groups of overlapping frequencies given in Table 3.1. The proceedings of the inversion can be seen in the misfit curve shown in Figure 3.12a). Additionally, the steplengths of each iteration are plotted in Figure 3.12b). By multiplying them to the normalised and scaled gradients, they define the size of the total model update. The test steplengths for the steplength calculation were decreased from 2% to 0.3%. The inversion starts with large steplengths and rapid convergence within the first frequency stage to find rough models. In the following stages, the steplengths are smaller and the misfit decreases slower, as finer changes within the model are made. The last frequency stage only gives very little model update and the misfit curve stagnates, so that I did not continue the inversion for higher frequencies.

The good convergence can also be seen in the fit of the seismograms presented exemplarily for one shot and receiver combination in Figure 3.13. In Figure 3.13 a), the waveforms of the starting model and the observed data, both filtered below 180 Hz are plotted, which show clear

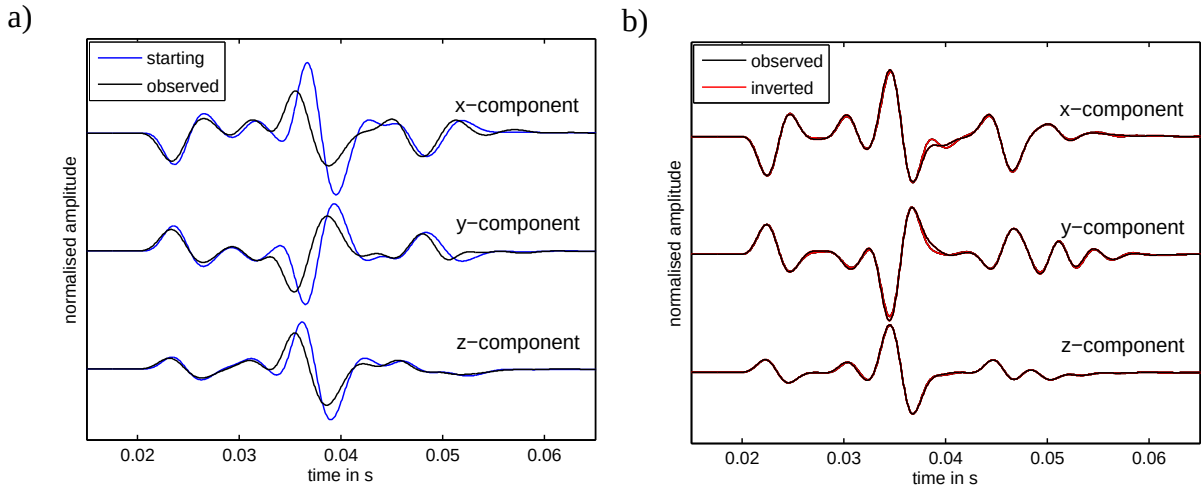

Figure 3.13: Multi-component waveforms normalised to one trace for exemplary receiver at (x_r, y_r, z_r)=(80 m, 30 m, 30 m) and source at (x_s, y_s, z_s)=(30 m, 70 m, 90 m), a) starting data versus observed data filtered below 180 Hz and b) observed versus final inverted data filtered below 300 Hz.

differences caused by the box. The residuals were successfully fitted during the inversion, and the observed and final inverted data, plotted in Figure 3.13 b) hold a good agreement.

3.7.2 Multi-parameter inversion

Wave propagation in the subsurface is influenced by various parameters. Here, I already approximate the wave propagation with the elastic wave equation, which does not consider anisotropy and attenuation. Still, 3D elastic FWI is a highly complex multi-parameter problem. Compared to the acoustic case, where often only the P-velocity is used as inversion parameter, the number of model parameters and consequently the nonlinearity of the inverse problem increases dramatically. The choice of parametrisation in the elastic case was already discussed in section 2.6 and I will stick to the choice of the seismic velocities v_p and v_s and density ρ in my applications. From these parameters, ρ is generally the most difficult parameter to resolve and especially in transmission geometry it is often not possible. This can also be seen in the simple toy example. In future applications in this thesis, I will therefore invert for the seismic velocities only and keep ρ constant during inversion.
A main difficulty of the inversion of v_p and v_s is caused by the different wavelengths of these parameters. Due to the smaller velocities, and thus smaller wavelengths of the S-wave, they require a lower starting frequency, but offer higher resolution for lower frequencies. Therefore, the elastic case generally requires lower starting frequencies or better starting models than the acoustic case. Additionally, if the inversion fails for high frequencies, I might be able to get a very good resolved v_s model but only a smooth v_p model.
Of course, a major difficulty of the multi-parameter inversion is the scaling of the individual parameters. So far, I obtained good results by normalising the gradients to their maximum and individually scaling them with some representative value as described in section 3.5.1. Still, results might be improved by Newton methods which offer a scaling of the individual parameter updates with respect to each other.

The final results for the multi-parameter FWI of the toy example are presented in Figure 3.14. The high velocity box could be nicely reconstructed in the velocity models v_p (a) and v_s (b). Hereby, the box is sharper and better confined in v_s than in v_p due to its smaller wavelengths. The inversion of density (Figure 3.14c) was not successful. Only the edge of the box is indicated by a zone of higher density, whereas low density artefacts are caused within the box and surrounding the box. I repeated the same inversion for a box varying in density only and inverting for density only. This mono-parameter inversion could successfully fit modelled and observed data. A better reconstruction of the box could be achieved (Figure 3.14d), where the outline of the box is clearly visible. Still, low density artefact surrounding the box and no significant density updates within the box show the low sensitivity of the inversion to density and the difficulties which arise during density inversion in transmission geometry.

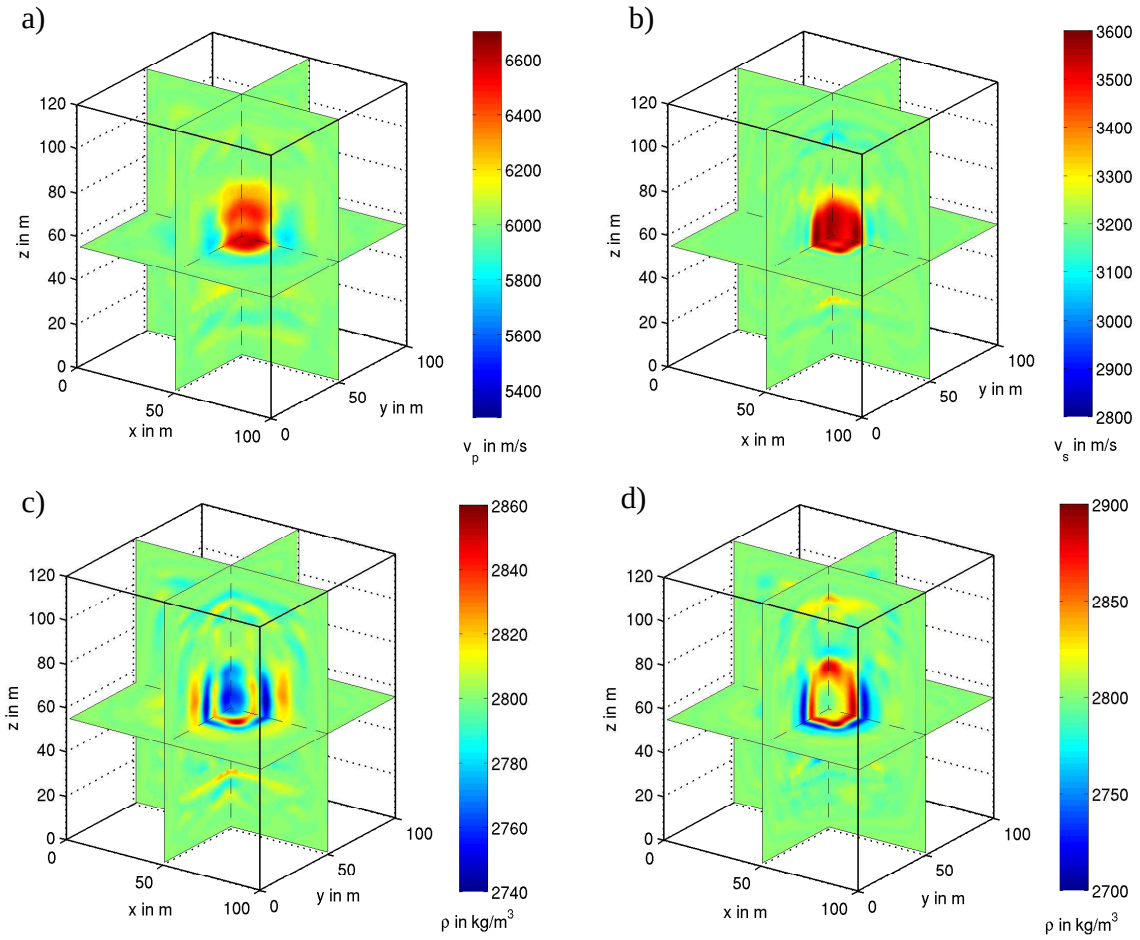

Figure 3.14: In a)-c): Final multi-parameter inversion results of the box-model with a) P-velocity, b) S-velocity and c) density, in d) final result for mono-parameter density inversion of box model.

Chapter 4

Reconstruction of small-scale heterogeneities in transmission geometry

4.1 Motivation

In this chapter I show the application of 3D FWI for the reconstruction of 3D heterogeneous crystalline rock structures. Two different transmission geometries are compared: an "optimal" 3D geometry with planes of sources and receivers and a more realistic 2D geometry, resembling two boreholes. The crystalline rock model is presented by a complex random medium model, which is characterised by differently sized 3D structures. Additionally a checkerboard test is performed to test the resolution in both geometries. This study tests the ability of 3D FWI to resolve differently sized 3D structures. Additionally we address the question, whether it can be beneficial to perform a 3D FWI even for a simple 2D cross-hole geometry. The 3D FWI results are compared to a 2D FWI of the 3D heterogeneous medium data to show the effects of 3D scattering on the 2D FWI result. The study was published in Butzer et al. (2013).

4.2 Inversion setup

Model and acquisition geometry

The model size is $240\,\mathrm{m} \times 160\,\mathrm{m} \times 280\,\mathrm{m}$, that is $300 \times 200 \times 350$ gridpoints in X-, Y- and Z-direction, respectively. The grid spacing is $dh = 0.8\,m$. Average material parameters of the real models were adapted with $v_p = 6300\,\mathrm{m/s}$, $v_s = 3500\,\mathrm{m/s}$ and $\rho = 2800\,\mathrm{kg/m^3}$. These average parameters correspond to the crystalline crust found at the German Continental Drilling site (KTB) drilling site at 3000 m to 6000 m depth (Kneib, 1995). The models do not contain a free surface. The seismic velocities show checkerboard or random medium structures whereas the density is kept homogeneous and constant during inversion.
The acquisition geometries are shown in Figure 4.1. The grey area corresponds to checkerboard or random medium structure, which changes to homogeneous medium (white area) towards source and receiver planes. For the 3D acquisition geometry (Figure 4.1a) 416 receivers and 12 sources, marked as crosses and stars, respectively, were regularly arranged in X-Y oriented

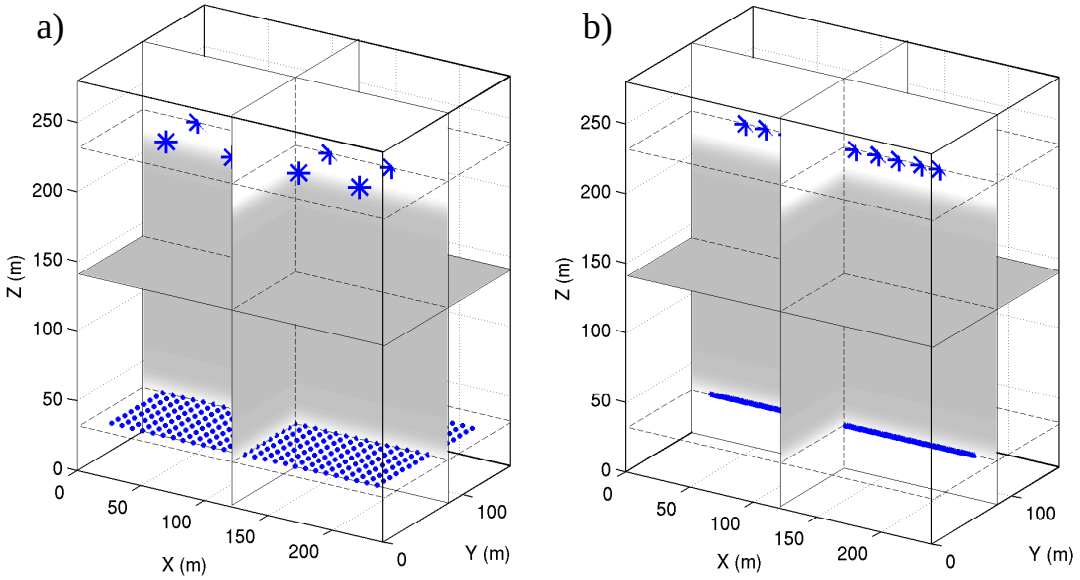

Figure 4.1: Acquisition geometries. Source and receiver positions are indicated by stars and crosses, respectively. a) 3D geometry with 12 sources and 416 receivers; b) 2D geometry with 10 sources and 100 receivers.

source and receiver planes. This configuration is used to test the performance of the 3D FWI in favourable geometry. For the 2D acquisition geometry only a line of 10 sources and a 200 m distant line of 100 receivers at $Y = 80$ m was used, as indicated in Figure 4.1b. Compared to the 3D geometry, additional sources and receivers were added in these lines to increase the coverage. With this geometry, the reconstruction of 3D structures using a 2D geometry is tested and compared to 2D FWI results. We applied point forces in Z-direction with a central source frequency of 200 Hz, emitting both P- and S-waves. Generally, we used all three receiver components in the 3D inversion and the X- and Z-component in the 2D inversion. The source wavelet is assumed to be known during inversion.

Inversion parameters

Overall, seven different inversion tests, varying in model, acquisition geometry, number of receiver components and dimensionality of the FWI as listed in Table 4.1, are presented here.

We inverted for v_p and v_s and chose homogeneous starting models, hereby adapting the average seismic velocities. The gradients were tapered around source and receiver planes and along the model boundaries to avoid artefacts in these areas. No regularisation was used. The inversion parameters, like the number of iterations and the frequency range of the FWI, are listed in Table 4.1. For the 3D FWI with 3D geometry up to three frequencies per iteration were used, which were successively increased in 20 Hz steps. In case of the 3D FWI with 2D geometry, the number of frequencies was increased to five, to refine the frequency sampling to 10 Hz. This improves the convergence and the quality of the results. The 2D FWI is performed with an elastic FWI code developed by Köhn (2011). This code is based on the same adjoint-state method but is a pure time-domain implementation which considers the full frequency content. The data was low-pass filtered in 20 Hz steps during inversion to obtain a better comparability.

	model	FWI	acquisition geometry	inverted frequencies (Hz)	S-wave-lengths (m)	P-wave-lengths (m)	number of iterations
Test 1	3D checker	3D	3D, 3C	$160 \rightarrow 320$	$22 \rightarrow 11$	$39 \rightarrow 20$	69
Test 2	3D checker	3D	2D, 3C	$160 \rightarrow 320$	$22 \rightarrow 11$	$39 \rightarrow 20$	75
Test 3	3D random	3D	3D, 3C	$140 \rightarrow 320$	$25 \rightarrow 11$	$45 \rightarrow 20$	73
Test 4	3D random	3D	2D, 3C	$100 \rightarrow 320$	$35 \rightarrow 11$	$63 \rightarrow 20$	89
Test 5	3D random	3D	2D, 1C	$100 \rightarrow 320$	$35 \rightarrow 11$	$63 \rightarrow 20$	73
Test 6	2.5D random	2D	2D, 2C	< 320	> 11	> 20	93
Test 7	3D random	2D	2D, 2C	< 220	> 16	> 2948	80

Table 4.1: Inversion tests varying in method, models, acquisition geometry and number of components (C) with applied inversion parameters.

4.3 Preliminary investigation - the checkerboard test

The checkerboard model (Figure 4.2) consists of alternating cubes of 20 m edge length and $\pm 5\%$ velocity variation in both v_p and v_s. The checkerboard is chosen antisymmetric to the 2D geometry source-receiver plane at $Y = 80\,m$, indicated with a white dashed line in Figure 4.2. Note that the Y-Z slice shown here is 12 m distant from this plane.

Inversion parameters used for FWI with 3D geometry (test 1) and 2D geometry (test 2) are given in Table 4.1. We used a minimum frequency of 160 Hz (S-wavelength 22 m) and increased it in 20 Hz steps to 320 Hz (S-wavelength 11 m, P-wavelength 20 m).

The models of P-wave and S-wave velocity inverted using data acquired with 3D and 2D geometries are plotted in Figure 4.3. When using data of a 3D source-receiver geometry (Figure 4.3a), the inversion can resolve the cubes well in v_s. The inversion is not able to resolve the sharp edges, as this would require the use of higher frequencies. The frequency range and number of iterations is not sufficient for a satisfactory resolution of v_p (Figure 4.3b), but a general checkerboard pattern is visible. The minimum P-wavelength we used is about the size of the checkerboard cubes. Thus, we can observe a minimum resolution at the scale of a wavelength (λ). This is much smaller than the width of the first Fresnel zone of $\sqrt{\lambda L} = 63$ m, where $L = 200$ m denotes the minimum propagation distance. The 3D FWI of data acquired with a 2D source-receiver geometry is capable to reconstruct the checkerboard pattern in the area adjacent to source-receiver plane in v_s (Figure 4.3c). Structures up to 40 m apart from the plane, which corresponds to about two wavelengths, could be recovered. Furthermore, the antisymmetric pattern of the checkerboard could be resolved. However, resolution is less than for the 3D geometry, which is due to the lower wave path coverage. The v_p model (Figure 4.3d) could not be inverted successfully for this simple geometry and in this frequency range. This example indicates that 3D FWI is capable to recover 3D structures in the vicinity of the direct propagation path and also structures outside the first Fresnel zone of the direct rays. However, it also indicates difficulties, that might arise during multiparameter inversion for simple geometries.

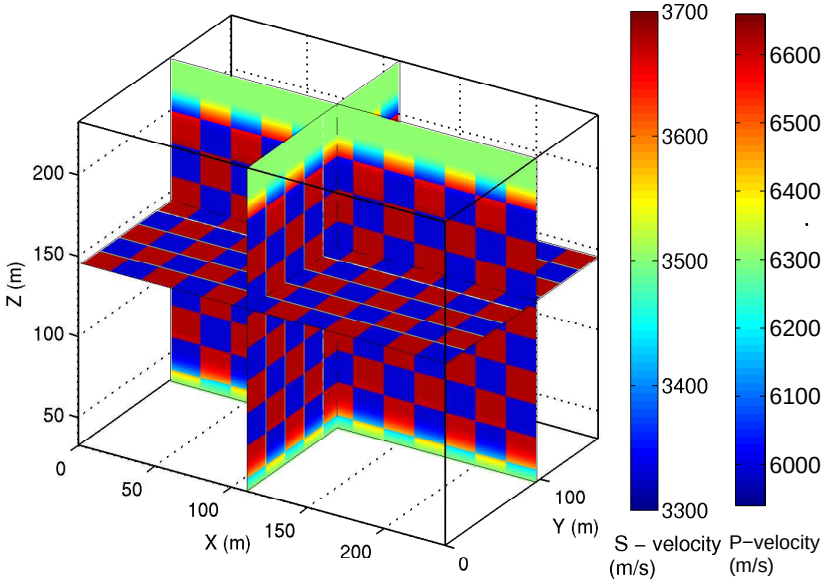

Figure 4.2: Checkerboard model v_s and v_p with 20 m edge length and $\pm 5\%$ velocity perturbations. The source-receiver plane of the 2D acquisition geometry at $Y = 80$ m is indicated by a white dashed line.

4.4 The random medium application

The crystalline crust can be characterised with statistical parameters of a random medium model as derived in studies by Kneib (1995) and Mueller & Shapiro (2001) for the German Continental Drilling site (KTB). In our tests we use a random medium (Figure 4.4) described by an exponential autocorrelation function in space with a correlation length of 30 m and a standard deviation of 5% for the Gaussian distributed seismic velocities. The average velocities are the same as in the previous section with $v_p = 6300$ m/s and $v_s = 3500$ m/s and a constant v_p/v_s ratio of 1.8. The model features differently sized 3D structures and is used here to test resolution of the FWI results for different scales.

The 3D scattering which is caused by the 3D heterogeneous structures contains all phases of first order scattering, that is PP, PS, SS and SP waves. Thus we aim to reconstruct the P-wave and S-wave velocities. Nevertheless, the resolution of the structures depends on the wavelength, which is 1.8 times larger for v_p than for v_s. Thus a lower resolution in v_p than in v_s is to be expected.

4.4.1 Results 3D full-waveform inversion

Inversion parameters are given in Table 4.1, test 3. The maximum S-wavelength of 25 m corresponds to half the size of the largest model structures. The smallest wavelengths are 20 m and 11 m of P- and S-waves, respectively. For the 2D acquisition geometry (test 4) lower frequencies of down to 100 Hz corresponding to S-wavelengths of about 35 m were required for the inversion to succeed.

The misfit was successfully decreased to final values of 1.5% of the initial misfit for the 3D

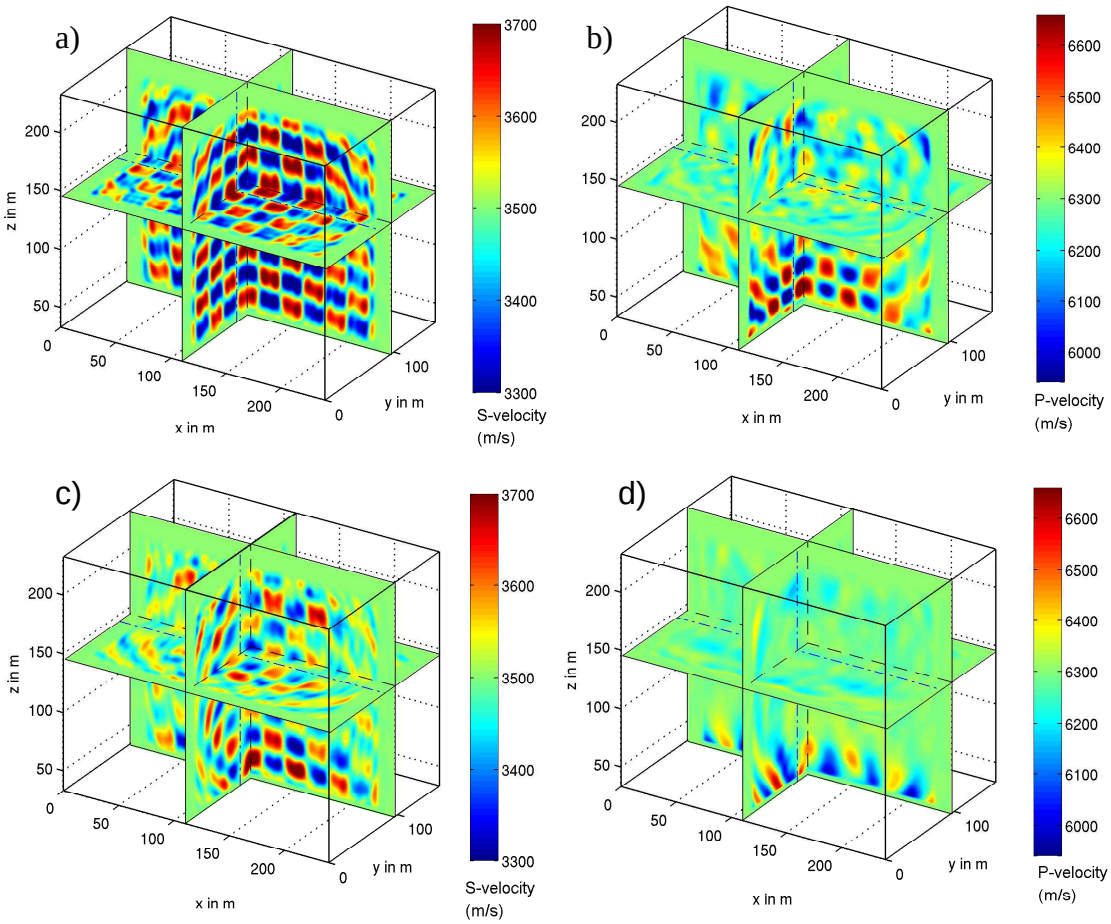

Figure 4.3: Checkerboard inversion results: a) 3D geometry (Figure 4.1a), test 1, S-wave velocity, b) 3D geometry (Figure 4.1a), test 1, P-wave velocity, c) 2D geometry (Figure 4.1b), test 2, S-wave velocity and d) 2D geometry (Figure 4.1b), test 2, P-wave velocity

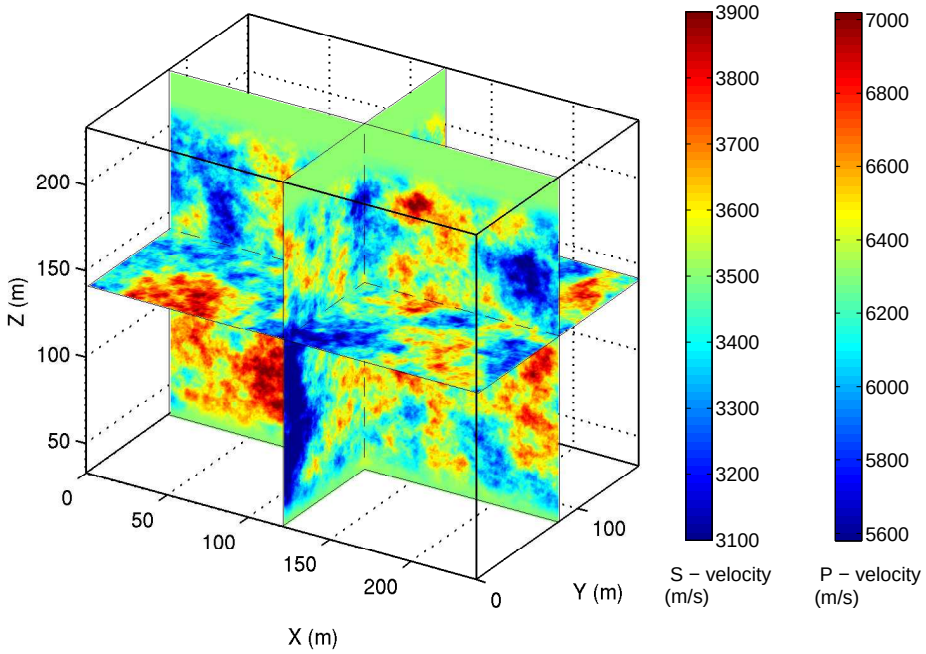

Figure 4.4: The random medium velocity model.

geometry data and 2.1% for the 2D geometry data. Figure 4.5 shows the observed, initial and inverted 3-component waveforms at one exemplary receiver and for one shot. Starting from a homogeneous model, the main phases of P- and S-waves as well as the scattered waves in the observed data can be well explained by the final model for all components.

In Figure 4.6a and Figure 4.6b the final inverted models of v_s and v_p for the 3D acquisition data are plotted. Compared to the true model in Figure 4.4, the differently sized 3D structures of the random medium are well resolved in v_s. Structures with an extension of about a wavelength could be recovered. Resolvability along the Z-axes is lower than in lateral direction. The v_p model is smoother due to its larger wavelengths. The quality decreases towards the model boundaries where wavefield coverage is lower. Overall, the seismograms computed for the homogeneous starting model in transmission geometry can explain the low frequency observed data sufficiently for the inversion to succeed. The lack of low frequencies, surface acquisition geometries or complex phenomena, such as the involvement of surface waves or multiple reflections, might require the improvement of the starting model and the application of further multi-stage strategies such as time windowing (e.g. Sears et al., 2008; Brossier et al., 2009; Kurzmann, 2012).

24The v_s model obtained for the 2D acquisition geometry data is shown for comparison in Figure 4.6c. Maximum wave path coverage is along the source-receiver plane at $Y = 80\,\mathrm{m}$. In this plane the main structures are recovered by the inversion. However, even though we used a denser spacing of sources and receivers within this plane and a refined frequency sampling, the quality is less than for the FWI with 3D geometry data. Similar to the checkerboard test, we observe that the 3D FWI succeeds to reconstruct 3D structures outside the source-receiver plane. Using multi-component 3D scattered waves, some structures up to two wavelengths apart from the plane are recovered. The v_p result in Figure 4.6d shows a smoother version of the v_s

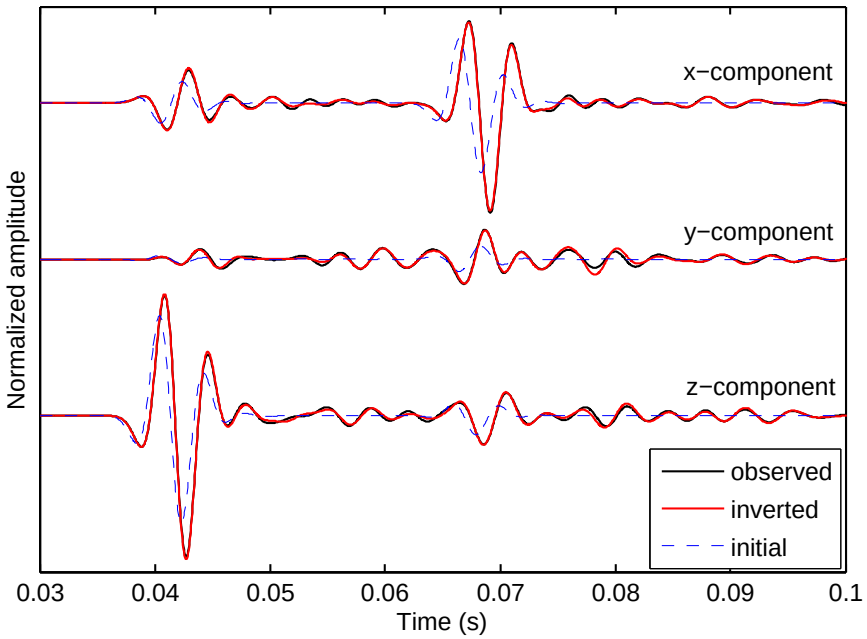

Figure 4.5: Observed, initial and inverted waveforms of 3D FWI with 3D acquisition geometry data (test 3) at one representative receiver (136 m, 72 m, 32 m) and for one shot at (96 m, 80 m, 232 m).

result. In contrast to the failed v_p checkerboard inversion with 2D geometry (Figure 4.3d) the inversion could recover the general random structures, even though its resolution is limited due to the large wavelengths.

Highest amplitudes of the S-wave in the source-receiver plane are found on the X-component seismograms. We repeated the 3D FWI with 2D source-receiver geometry using only the X-component of the observed data to see the influence of multi-component data (test 5). The final v_s model is plotted in Figure 4.7. Within source-receiver plane the reconstructed structures look quite similar to the three component result in Figure 4.6c. The main difference is, that gradients and model updates of the X-component inversion are symmetric to source-receiver plane. Structures antisymmetric to source-receiver plane cannot be resolved correctly without the additional information about the Y-component waveforms.

4.4.2 Comparison of 3D and 2D inversion results

In many applications 2D FWI is applied to cross-well seismic experiments (e.g. Pratt et al., 2005; Zhang et al., 2012). In these studies structures outside the acquisition plane are neglected but may distort the inversion results. To study the possible site effects of 3D structure we compare the 3D FWI results with the results of a 2D FWI. For the 2D FWI we used the same 2D acquisition geometry as previously. We thus try to resolve the plane at $Y = 80$ m of the random medium. Two tests were performed: test 6 and test 7 (Table 4.1). As a reference result a 2D FWI of 3D modelled data of the 2.5D random medium was performed (test 6). The 2.5D model extends model parameters at Y=80 m of the random medium constant in Y-direction. The data of this model are thus not influenced by 3D scattering and a 2D approximation is valid. In test 7

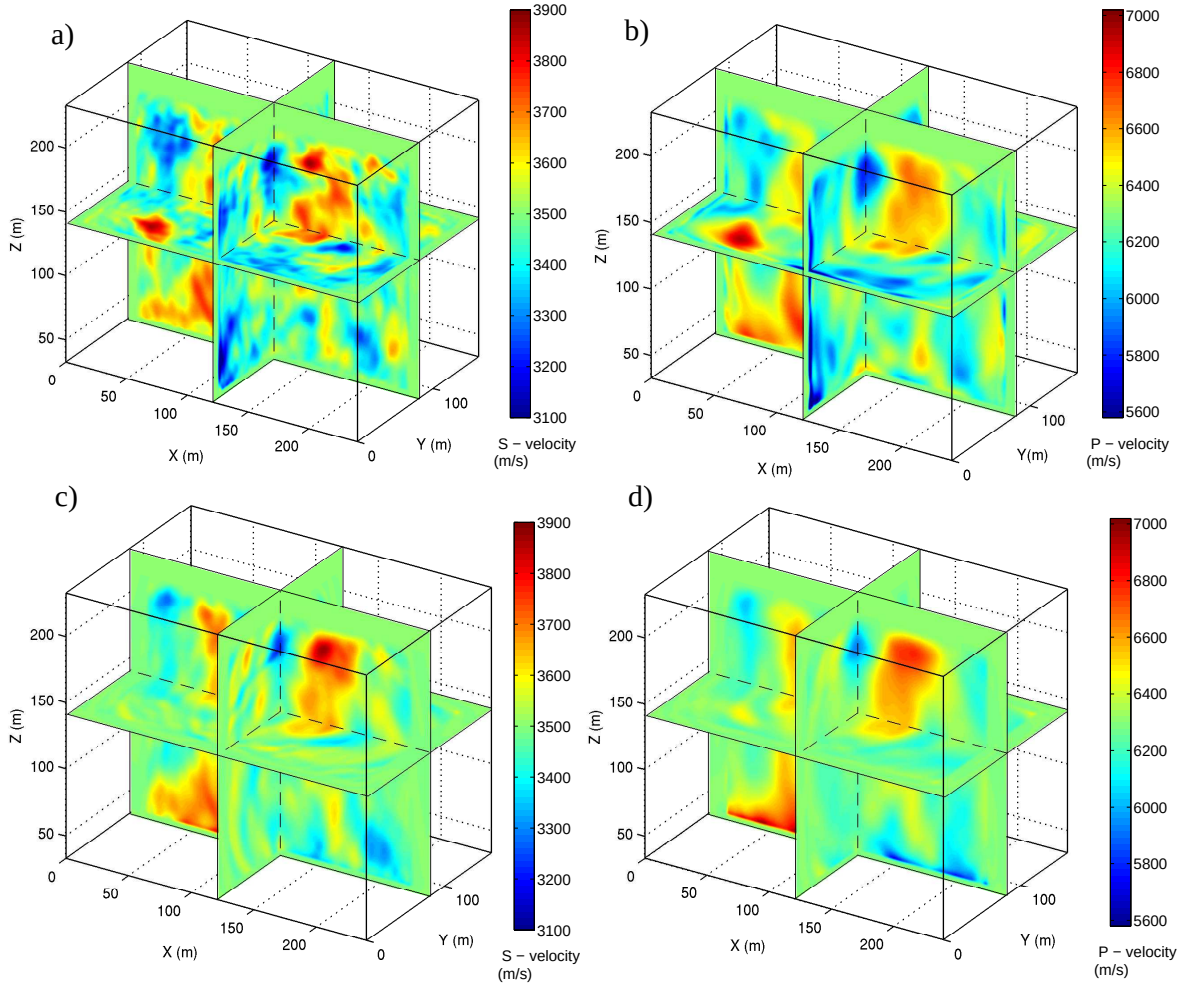

Figure 4.6: 3D FWI results for the random medium model using different acquisition geometries: a) model v_s inverted with 3D geometry (test 3), b) model v_p inverted with 3D geometry (test 3), c) model v_s inverted with 2D geometry (test 4) and d) model v_p inverted with 2D geometry (test 4).

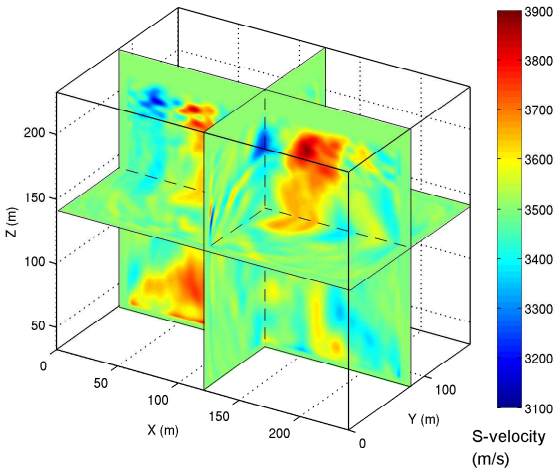

Figure 4.7: 3D FWI result (v_s) for the random medium model using the 2D acquisition geometry and inverting only X-component.

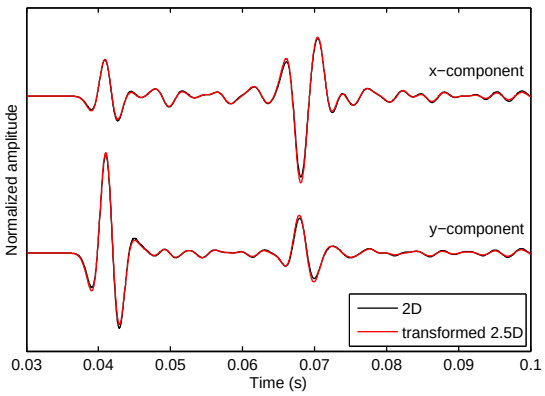

Figure 4.8: Waveform comparison of 2D data calculated for 2D random medium compared to 3D to 2D transformed data of the 2.5D random medium at receiver (111 m, 80 m, 32 m) for source location (96 m, 80 m, 232 m).

the data of the 3D random medium were inverted, which contain the 3D scattered waves.
To invert the 3D data with a 2D method, a transformation of observed data from 3D to 2D was applied to account for the differences between line source and point source wavefields. We convolved seismograms with $\frac{1}{\sqrt{t}}$ as phase correction and multiplied amplitudes with $\sqrt{t}$ as amplitude correction (Pica et al., 1990). The transformed 3D modelled data for the 2.5D model agree very well with the 2D data modelled for the plane at $Y = 80\,m$. This is shown for one shot and receiver in Figure 4.8. It proves, that the transformation works well for the given scenarios.
We applied a 2D FWI to the transformed data of the 2.5D model and of the 3D model. In both cases the 2D inversion successfully minimised the misfit to 1-2% of the initial value.
The 2D FWI results for v_s and v_p are shown in Figure 4.9. The slice of the true model (Figure 4.9a) is smoothed, to enable a better comparison. The 2D FWI of the 2.5D model data (Figure 4.9b and c) successfully recovered the random medium structure in v_s and v_p. Resolution is again lower in v_p. Its quality and resolution is comparable to the 3D FWI result achieved with 3D source-receiver geometry (Figure 4.6a and b). The main features in v_s can still be reconstructed when inverting the data of the 3D heterogeneous random medium with 2D FWI (Figure 4.9d). However, smaller structures suffer in resolution and artefacts are caused by 3D site-scattering. This can also be seen in the v_p model in Figure 4.9e. The artefacts become more dominant for higher frequencies and we therefore show the result for a maximum inversion frequency of 220 Hz here, where large scale structures are still reliable. The artefacts can be explained when looking at the effects of 3D scattering on the waveforms. Figure 4.10a shows a comparison between the 2.5D random medium data without 3D scattering versus the 3D random medium data with 3D scattering for X-, Y-, and Z-component at one exemplary receiver. 3D scattering has a significant influence on the wavefields. In Figure 4.10b the observed, initial and inverted waveforms of the 2D FWI filtered with 220 Hz are plotted and it is evident, that the 2D FWI tries to fit the 3D scattered waveforms. However, the 2D FWI is unable to interpret the 3D scattering correctly, leading to artefacts and the lack of resolution. The quality and resolution of 2D FWI thus suffers from the presence of 3D small scale heterogeneities.

A 3D FWI with 2D source-receiver geometry (test 4) has the advantage to gain information about 3D structures outside source-receiver plane, as was shown in the previous section. A comparison of the final model achieved with 3D FWI (Figure 4.6b) and the 2D FWI result of the 3D random medium (Figure 4.9c) shows, that the resolution within source-receiver plane is comparable. However, the 3D FWI does not suffer from artefacts caused by the 3D scattering.

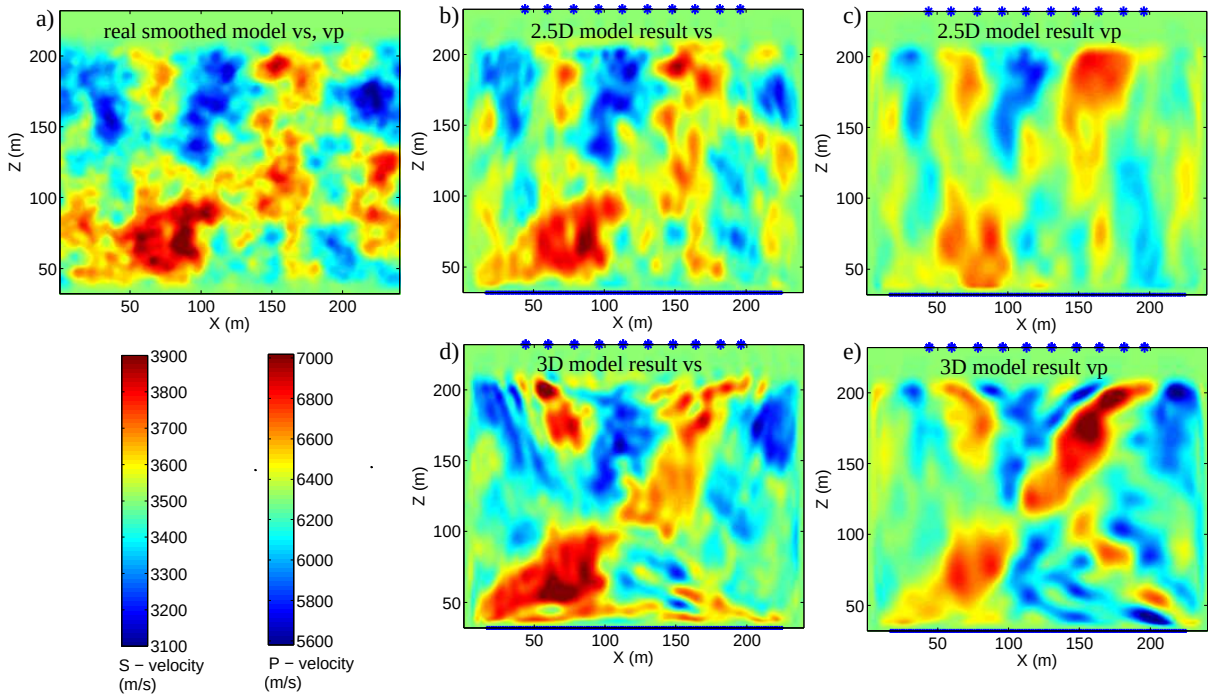

Figure 4.9: Model reconstruction using 2D FWI; sources (stars) and receivers (crosses) are marked. a) real smoothed model v_p and v_s, b) v_s for 2.5D random medium data (test 6) as reference, c) v_p for 2.5D random medium data (test 6), d) v_s for 3D random medium data (test 7) and e) v_p for 3D random medium data (test 7).

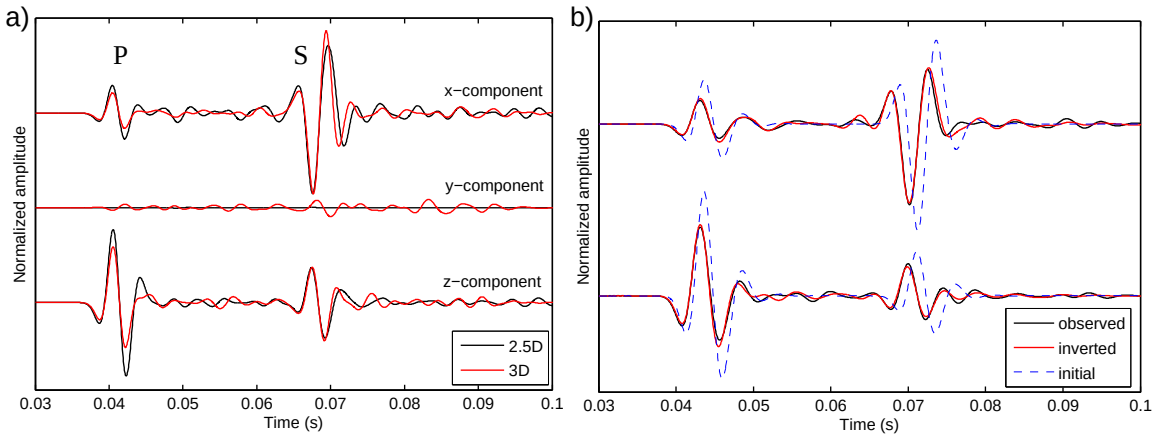

Figure 4.10: Waveforms (low-pass filtered with 220 Hz) at receiver (111 m, 80 m, 32 m) for source location (96 m, 80 m, 232 m): a) The comparison of waveforms for the 3D and the 2.5D random medium model shows the influence of 3D scattering. b) Observed, initial and final inverted waveforms of the 2D FWI with 3D random medium data (test 7).

Table 4.2 gives a comparison of the computation times for 2D FWI and 3D FWI. The 2D FWI requires only about 64 core hours in total for our example and was performed on a 8 CPU workstation. The 3D FWI requires 28.000 core hours and can be realised nowadays on high performance computers.

3D FWI	800 cores	73 iterations	$300 \times 200 \times 350$	35 hours
2D FWI	8 cores	67 iterations	300×350	8 hours

Table 4.2: Comparison of run times for a 2D and a 3D elastic FWI using 10 sources, 28 forward modellings per iteration and 3300 timesteps.

4.5 Conclusion

The performance of the 3D FWI was shown for a random medium and a checkerboard example in transmission configuration inverting for v_p and v_s. Using a 3D acquisition geometry the 3D elastic FWI can successfully recover the 3D checkerboard and the differently sized random medium velocity structures starting from a homogeneous initial model, if sufficiently low frequencies are included in the inversion. Using an acquisition geometry of two boreholes 3D structures of checkerboard and random medium are recovered up to one to two wavelengths adjacent to the source-receiver plane, however with less resolution. Inverted models for the compressional velocity are generally smoother due to its larger wavelengths. The consideration of only one component oriented parallel to the source-receiver plane, causes a symmetric shape of the inversion result and structures antisymmetric to this plane cannot be reconstructed correctly. The results indicate, that the application of 3D FWI to multi-component cross-well data might be used to detect 3D structures in an area around the acquisition plane. For comparison a 2D FWI was performed. In the presence of 3D small-scale heterogeneities the 2D FWI results for shear and compressional velocity suffer from artefacts due to 3D scattering effects, especially when inverting at higher frequencies. Only the main structures in the acquisition plane were reconstructed.

Chapter 5

Applying a diagonal Hessian approximation for preconditioning

5.1 Motivation

The misfit gradients offer information about structures of the true model. However, the amplitude distribution of the misfit gradients is affected by acquisition geometry, the illumination by the different wavetypes propagating through the medium and by the amplitude decay of the wavefields. Due to geometric amplitude spreading of forward- and back-propagated wavefields, the amplitudes of the gradients decrease with distance from sources and receivers. This becomes especially evident in the vicinity of source and receiver positions, where high amplitudes in the gradients occur. Additionally, we have areas of low wave path coverage, such as towards the boundaries, where amplitudes are smaller. In reflection geometries, where sources and receivers are located at the surface, we generally observe an amplitude decay with depth. On the one hand, this is caused by the geometric spreading of the wavefields. On the other hand, less energy penetrates the deeper parts due to reflection and diffraction of wavefields at the model structures. Additionally, if a free surface is considered, the high amplitude surface wave causes high amplitudes of the gradients near the surface, which strongly decay with depth.
The gradient preconditioning corrects some of these amplitude dependencies of the gradient and is generally required for a successful FWI with gradient methods. It aims to optimise the inversion by improving its convergence and by regularising the inversion to find a physically plausible model (Fichtner, 2011). Hereby it can, for example, correct for source and receiver artefacts and enhance the image with depth.
Different approaches of preconditioning exist for FWI. Empirically, the gradients can be radially tapered around source and receiver positions to remove high amplitude artefacts in these areas, as discussed in section 3.4. A main disadvantage hereby is, that we do not get a significant model update close to source and receiver positions. This can cause problems, if model parameters in this area are unknown. Additionally, it only corrects the geometrical amplitude decay in the gradients locally. This can be sufficient for transmission geometry applications (see chapter 4). However, more complex examples, like reflection geometry applications, might require additional preconditioning. Another empirical method uses a gain function to scale the gradient with depth (e.g. Köhn, 2011) and to enhance deeper structures for reflection geometries.

A preconditioning to correct for geometric amplitude spreading based on wavefield parameters is also possible (Fichtner et al., 2009; Kurzmann, 2012). Hereby, the distribution of the maximum amplitudes of forward- and back-propagated wavefields is used. This method takes into account model inhomogeneities and thus offers an improved correction compared to a radial taper, especially in the vicinity of sources and receivers.
A different type of preconditioning is physically motivated and based on an approximation of the diagonal Hessian. Hence, it includes information about the second derivative of the misfit function in order to optimise the inversion process. To understand this idea we compare the update of the Newton method ($-\mathbf{H}_k^{-1}\nabla_m E_k$) and the update of the gradient method ($-\alpha_k \mathbf{P}\nabla_m E_k$) as introduced in section 2.3. The gradient method approximates the inverse Hessian $\mathbf{H}_k^{-1}$ by some preconditioning operator $\mathbf{P}$ scaled with a steplength α_k. Hence, it is reasonable to include some information about $\mathbf{H}_k$ in $\mathbf{P}$ to design an efficient, physically funded preconditioning operator, which can be seen as an optimisation of the gradient-based FWI. The Hessian preconditioning could be proven in several successful applications, such as Shin et al. (2001); Operto et al. (2006); Brossier et al. (2009); Warner et al. (2013). Hereby different approximations are used for calculation to ensure computational efficiency, as for example the pseudo-Hessian introduced by Shin et al. (2001) or the calculation for only a subset of shots (Operto et al., 2006).
In this chapter I will present the concept of Hessian preconditioning for the time-frequency method. Different approximations of the diagonal approximate Hessian are made, to allow its computation at reasonable costs. I will start with the theory, discuss its implementation in 3D-AWAIT. After validation of the approach with the inversion of a simple box model I will apply the method for the inversion of a complex surface geometry model.

5.2 Theory

5.2.1 The Hessian operator

The Hessian operator was already shortly introduced in section 2.3. Here I want to take a closer look at its calculation and its role in the inversion. The elements of the Hessian operator H_{jl} are defined as the second derivative of the misfit function $E(\mathbf{m})$ with respect to the model parameters m_j and m_l.

$$H_{jl} = \frac{\partial^2 E(\mathbf{m})}{\partial m_j \partial m_l} \quad (j = 1,...n)\ \ (l = 1,...n). \tag{2.11}$$

The indices j and l run over all n model parameters, which results in a very large matrix $\mathbf{H}$. I use the L_2-norm based misfit function E as given in equation 2.18. Its second derivative can be expressed as

$$H_{jl} = \sum_s \int_0^T dt \sum_r \frac{\partial u_i}{\partial m_j}\frac{\partial u_i}{\partial m_l} + \sum_s \int_0^T dt \sum_r \frac{\partial^2 u_i}{\partial m_j \partial m_l}\delta u_i = (H_a)_{jl} + R_{jl}. \tag{5.1}$$

The i-th displacement component is denoted by $u_i(\mathbf{x}_s, \mathbf{x}_r, t)$, whereas $\delta u_i(\mathbf{x}_s, \mathbf{x}_r, t)$ is the component of the displacement residual. To calculate the Hessian in equation 5.1 we sum over all data points, i.e. over all sources s, receivers r, components i and integrate over time. The calculation of the full Hessian requires two terms. The first term $\mathbf{H}_a$ is calculated as a zero-lag

cross-correlation of the first order partial derivative wavefields, and is known as "approximate Hessian". The second term **R** contains the second order partial differential wavefields. This term is often difficult and expensive to compute (Pratt et al., 1998) and most applications of Newton methods constrain to the Gauss-Newton method, which only takes into account the approximate Hessian $\mathbf{H}_a$. The full Newton method, by contrast, takes into account the full Hessian matrix.
Compared to gradient methods, Newton methods offer a much better convergence and can result in improved reconstruction of the subsurface by using the information contained in the Hessian. The role of the Hessian is well described in Pratt et al. (1998); Virieux & Operto (2009) and Brossier (2011). $\mathbf{H}_a$ is dominated by its diagonal elements, which are calculated as zero-lag autocorrelation of the partial derivative wavefields. These elements predict the geometrical amplitude spreading contained in the gradient and can thus be used to find an improved spatial scaling of the gradient. In surface seismic, the scaling of the gradient with the diagonal elements of $\mathbf{H}_a$ enhances the influence of deeper parts of the model. I will use an approximation of this diagonal part of $\mathbf{H}_a$ for preconditioning. The off-diagonal elements of the approximate Hessian account for the limited-bandwidth effects caused by the acquisition geometry. They additionally allow a correct scaling of the different inversion parameters, which can be very favourable for multiparameter inversion. Overall, applying the inverse Hessian to the gradient can be seen as a sharpening and focusing of the gradient, which leads to a better performance. Further improvements can be achieved with the use of the full Hessian, in which the second term **R** takes multi-scattering events into account.
The main drawback of the Gauss-Newton and especially the full Newton method are the extensive computational costs of runtime and storage to calculate and invert the large Hessian matrix. A good alternative is given by quasi-Newton methods, which I will discuss in chapter 6.
The use of an approximation of the diagonal $\mathbf{H}_a$ for preconditioning in the gradient method, which I will present in this chapter, is not equal to a Gauss-Newton method and therefore does not share all benefits. Still I will show, that it can improve our inversion sheme and enable the inversion of complex models and that it can be implemented at reasonable computational costs in a 3D elastic FWI.
Same as for the calculation of the gradients, we use a time-frequency approach for the Hessian calculation. This essentially means, that the wavefields required for the Hessian are simulated in time domain, but the Hessian calculation is performed in frequency domain for nf discrete frequencies. The approximate Hessian in equation 5.1 in frequency domain for discrete frequencies can be expressed as

$$(H_a)_{jl} = \sum_s \sum_r \sum_{v=1}^{nf} \frac{\partial \tilde{u}_i}{\partial m_j} \left(\frac{\partial \tilde{u}_i}{\partial m_l} \right)^* , \tag{5.2}$$

where $\tilde{u}_i(\mathbf{x}_s, \mathbf{x}_r, \omega_v)$ is the displacement in frequency domain and * denotes its complex conjugate.

5.2.2 The partial derivative wavefields

For the calculation of the approximate Hessian (equation 5.2) the explicit calculation of the partial derivative wavefields, also known as Fréchet derivative kernels, is required. This is a main difference to the calculation of the gradient, where the adjoint method avoids these

expensive calculations by introducing the adjoint wavefield. In section 2.4 the Fréchet derivative kernels for the model parameters λ, μ and ρ were already derived as (see equation 2.29)

$$\begin{aligned}
\frac{\partial u_i(\mathbf{x}_r,t)}{\partial \rho(\mathbf{x})} &= K_i^{\rho} = -\int_0^T d\tau \frac{\partial}{\partial \tau} G_{ij}(\mathbf{x}_r,t;\mathbf{x},\tau)\frac{\partial u_j(\mathbf{x},\tau)}{\partial \tau},\\
\frac{\partial u_i(\mathbf{x}_r,t)}{\partial \lambda(\mathbf{x})} &= K_i^{\lambda} = -\int_0^T d\tau \frac{\partial}{\partial x_j} G_{ij}(\mathbf{x}_r,t;\mathbf{x},\tau)\frac{\partial u_p}{\partial x_p},\\
\frac{\partial u_i(\mathbf{x}_r,t)}{\partial \mu(\mathbf{x})} &= K_i^{\mu} = -\int_0^T d\tau \frac{\partial}{\partial x_k} G_{ij}(\mathbf{x}_r,t;\mathbf{x},\tau)\left(\frac{\partial u_j(\mathbf{x},\tau)}{\partial x_k} + \frac{\partial u_k(\mathbf{x},\tau)}{\partial x_j}\right).
\end{aligned} \tag{2.29}$$

The Green's functions $G_{ij}(\mathbf{x}_r,t;\mathbf{x},\tau)$ only depend on the time difference $t-\tau$ and with the use of spatial reciprocity (equation A.3) they can be written as

$$G_{ij}(\mathbf{x}_r,t;\mathbf{x},\tau) = G_{ji}(\mathbf{x},t-\tau;\mathbf{x}_r,0). \tag{5.3}$$

These Green's receiver functions thus describe the j-th component of the displacement wavefield, which is generated at the receiver position $\mathbf{x}_r$ by a delta pulse point force in i-direction. I can thus calculate the partial derivative wavefields from equation 2.29 with the following formulas

$$\begin{aligned}
\frac{\partial u_i(\mathbf{x}_r,t)}{\partial \rho(\mathbf{x})} &= -\int_0^T d\tau \frac{\partial}{\partial \tau} G_{ji}(\mathbf{x},t-\tau;\mathbf{x}_r,0)\frac{\partial u_j(\mathbf{x},\tau)}{\partial \tau},\\
\frac{\partial u_i(\mathbf{x}_r,t)}{\partial \lambda(\mathbf{x})} &= -\int_0^T d\tau \frac{\partial}{\partial x_j} G_{ji}(\mathbf{x},t-\tau;\mathbf{x}_r,0)\frac{\partial u_p}{\partial x_p},\\
\frac{\partial u_i(\mathbf{x}_r,t)}{\partial \mu(\mathbf{x})} &= -\frac{1}{2}\int_0^T d\tau \left(\frac{\partial}{\partial x_k} G_{ji}(\mathbf{x},t-\tau;\mathbf{x}_r,0) + \frac{\partial}{\partial x_j} G_{ki}(\mathbf{x},t-\tau;\mathbf{x}_r,0)\right)\left(\frac{\partial u_j(\mathbf{x},\tau)}{\partial x_k} + \frac{\partial u_k(\mathbf{x},\tau)}{\partial x_j}\right).
\end{aligned} \tag{5.4}$$

These equations contain a convolution of the Green's receiver functions with the forward-propagated wavefields. To find the partial derivatives for discrete frequencies, I transform the equations 5.4 using the convolution theorem, i.e.,

$$C(t) = \int d\tau A(\tau - t)B(\tau) \quad \rightarrow \quad \tilde{C}(\omega) = \sqrt{2\pi}\tilde{A}(\omega)\tilde{B}(\omega) \tag{5.5}$$

Hereby, the Fourier transform as defined in equation 2.36 is used. I now apply this theorem to the equations 5.4. The time derivatives transform to the Fourier domain with a factor of $i\omega$. This results in

$$\begin{aligned}
\frac{\partial \tilde{u}_i(\mathbf{x}_r,\omega_v)}{\partial \rho(\mathbf{x})} &= \omega_v^2 \tilde{G}_{ji}(\mathbf{x},\omega_v;\mathbf{x}_r,0)\tilde{u}_j(\mathbf{x},\omega_v),\\
\frac{\partial \tilde{u}_i(\mathbf{x}_r,\omega_v)}{\partial \lambda(\mathbf{x})} &= -\frac{\partial}{\partial x_j}\tilde{G}_{ji}(\mathbf{x},\omega_v;\mathbf{x}_r,0)\frac{\partial \tilde{u}_p(\mathbf{x},\omega_v)}{\partial x_p},\\
\frac{\partial \tilde{u}_i(\mathbf{x}_r,\omega_v)}{\partial \mu(\mathbf{x})} &= -\frac{1}{2}\left(\frac{\partial}{\partial x_k}\tilde{G}_{ji}(\mathbf{x},\omega_v;\mathbf{x}_r,0) + \frac{\partial}{\partial x_j}\tilde{G}_{ki}(\mathbf{x},\omega_v;\mathbf{x}_r,0)\right)\left(\frac{\partial \tilde{u}_j(\mathbf{x},\omega_v)}{\partial x_k} + \frac{\partial \tilde{u}_k(\mathbf{x},\omega_v)}{\partial x_j}\right).
\end{aligned} \tag{5.6}$$

These equations can finally be used to calculate the Hessian preconditioner.

5.2.3 The calculation of the diagonal Hessian preconditioner

We only use the diagonal of $\mathbf{H}_a$ in equation 5.2, which is

$$(H_a)_{jj} = \sum_s \sum_r \sum_{v=1}^{nf} \frac{\partial \tilde{u}_i}{\partial m_j} \left(\frac{\partial \tilde{u}_i}{\partial m_j} \right)^* . \tag{5.7}$$

So far, the index j runs over all model parameters. Because we only use the diagonal, we can also reformulate this in one expression for λ, μ and ρ each:

$$\begin{aligned} H_a^{\rho}(\mathbf{x}) &= \sum_s \sum_r \sum_{v=1}^{nf} \frac{\partial \tilde{u}_i(\mathbf{x}_r, \omega_v)}{\partial \rho(\mathbf{x})} \left(\frac{\partial \tilde{u}_i(\mathbf{x}_r, \omega_v)}{\partial \rho(\mathbf{x})} \right)^*, \\ H_a^{\lambda}(\mathbf{x}) &= \sum_s \sum_r \sum_{v=1}^{nf} \frac{\partial \tilde{u}_i(\mathbf{x}_r, \omega_v)}{\partial \lambda(\mathbf{x})} \left(\frac{\partial \tilde{u}_i(\mathbf{x}_r, \omega_v)}{\partial \lambda(\mathbf{x})} \right)^*, \\ H_a^{\mu}(\mathbf{x}) &= \sum_s \sum_r \sum_{v=1}^{nf} \frac{\partial \tilde{u}_i(\mathbf{x}_r, \omega_v)}{\partial \mu(\mathbf{x})} \left(\frac{\partial \tilde{u}_i(\mathbf{x}_r, \omega_v)}{\partial \mu(\mathbf{x})} \right)^* . \end{aligned} \tag{5.8}$$

The diagonal approximate Hessian for λ, μ, and ρ are calculated from the partial derivative wavefields given in equation 5.6 multiplied with their complex conjugate value and summed up over all data points, i.e., all source-receiver combinations and frequencies.
As preconditioning operator ($P^{\rho}(\mathbf{x})$, $P^{\lambda}(\mathbf{x})$ and $P^{\mu}(\mathbf{x})$) for the different parameters I use the inverse of the diagonal of $\mathbf{H}_a$, which is straightforward for a diagonal matrix, and I get

$$\begin{aligned} P^{\rho}(\mathbf{x}) &= (H_a^{\rho}(\mathbf{x}) + \varepsilon_{\rho})^{-1}, \\ P^{\lambda}(\mathbf{x}) &= (H_a^{\lambda}(\mathbf{x}) + \varepsilon_{\lambda})^{-1}, \\ P^{\mu}(\mathbf{x}) &= (H_a^{\mu}(\mathbf{x}) + \varepsilon_{\mu})^{-1}. \end{aligned} \tag{5.9}$$

The coefficients ε_{ρ},ε_{λ} and ε_{μ} are water levels, which are summed to the Hessian approximations for reasons of stability. Otherwise, very small elements of $\mathbf{H}_a$ in areas of very low wave coverage cause a strong increase of the gradient in these areas which can result in artefacts.
I employ the seismic velocity parametrisation with v_p, v_s and ρ' and thus require the preconditioning operators for these parameters. For their calculation, I transform the partial derivative wavefields with respect to λ, μ, and ρ as calculated in equation 5.6 to wavefield derivatives with respect to the seismic velocities. The transformation is similar to the gradient transformation described in section 2.6. This results in

$$\begin{aligned} \frac{\partial \tilde{u}_i}{\partial v_p} &= \frac{\partial \tilde{u}_i}{\partial \lambda}\frac{\partial \lambda}{\partial v_p} + \frac{\partial \tilde{u}_i}{\partial \mu}\frac{\partial \mu}{\partial v_p} + \frac{\partial \tilde{u}_i}{\partial \rho}\frac{\partial \rho}{\partial v_p} = 2\rho v_p \frac{\partial \tilde{u}_i}{\partial \lambda}, \\ \frac{\partial \tilde{u}_i}{\partial v_s} &= \frac{\partial \tilde{u}_i}{\partial \lambda}\frac{\partial \lambda}{\partial v_s} + \frac{\partial \tilde{u}_i}{\partial \mu}\frac{\partial \mu}{\partial v_s} + \frac{\partial \tilde{u}_i}{\partial \rho}\frac{\partial \rho}{\partial v_s} = -4\rho v_s \frac{\partial \tilde{u}_i}{\partial \lambda} + 2\rho v_s \frac{\partial \tilde{u}_i}{\partial \mu}, \\ \frac{\partial \tilde{u}_i}{\partial \rho'} &= \frac{\partial \tilde{u}_i}{\partial \lambda}\frac{\partial \lambda}{\partial \rho'} + \frac{\partial \tilde{u}_i}{\partial \mu}\frac{\partial \mu}{\partial \rho'} + \frac{\partial \tilde{u}_i}{\partial \rho}\frac{\partial \rho}{\partial \rho'} = (v_p^2 - 2v_s^2)\frac{\partial \tilde{u}_i}{\partial \lambda} + v_s^2 \frac{\partial \tilde{u}_i}{\partial \mu} + \frac{\partial \tilde{u}_i}{\partial \rho}. \end{aligned} \tag{5.10}$$

Inserting these partial derivative wavefields into equation 5.8, I can calculate the Hessian approximations $H_a^{v_p}(\mathbf{x})$, $H_a^{v_s}(\mathbf{x})$ and $H_a^{\rho'}(\mathbf{x})$. Finally, the preconditioning operators for the seismic velocity parametrisation $P^{v_p}(\mathbf{x})$, $P^{v_s}(\mathbf{x})$ and $P^{\rho'}(\mathbf{x})$ can be calculated similar to equation 5.9.

5.3 Implementation

The workflow presented in Figure 5.1 illustrates the steps for the estimation of the Hessian preconditioner, which will be described in the following. The approximate Hessian does contain information about the model of the k-th iteration ($\mathbf{m}_k$), and about the acquisition geometry. However, it does not include information about the observed data, and thus about the true subsurface model. The Hessian approximation is calculated only once at the beginning of each frequency stage and then applied to the gradients within this frequency stage.

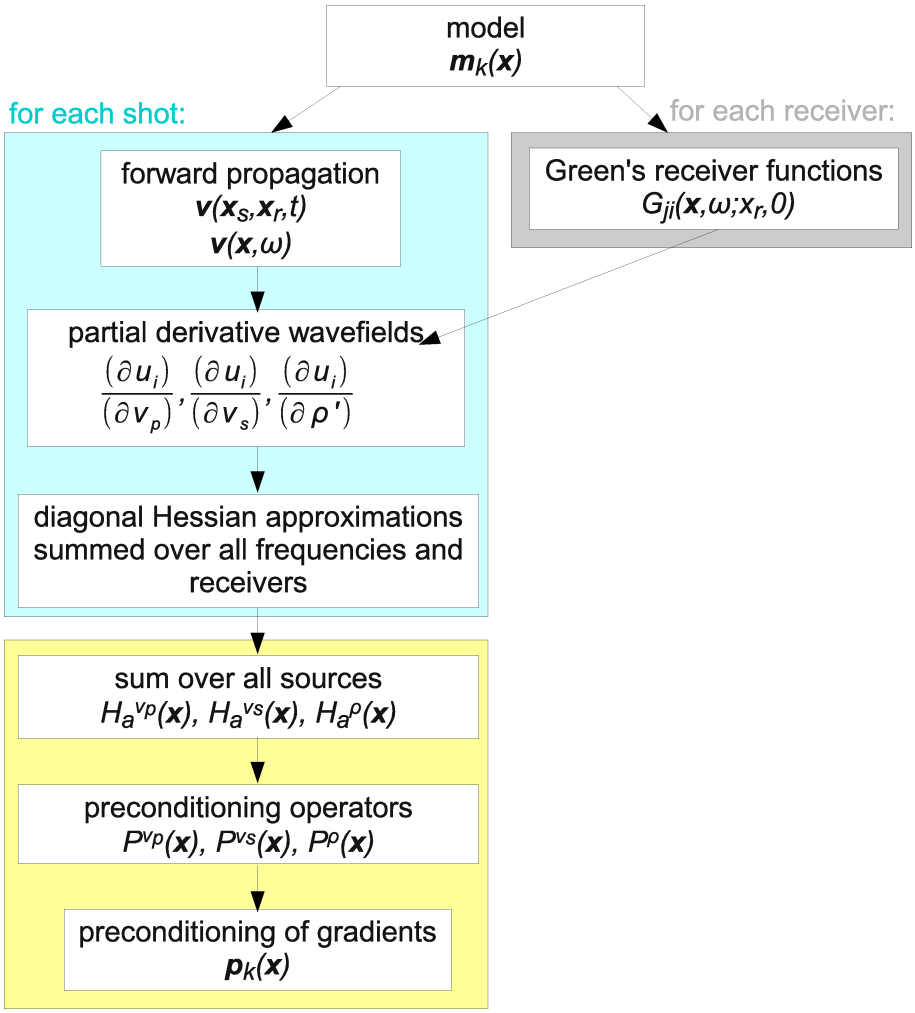

Figure 5.1: Workflow for the calculation of the Hessian preconditioning operator at iteration k as implemented in 3D-AWAIT for the seismic velocity parametrisation.

5.3.1 Computation of the Hessian approximation

The Hessian requires the knowledge of the partial derivative wavefields, which are calculated according to equation 5.6 in frequency domain for each discrtete frequency and each source-receiver combination.

Green's receiver functions

In a first step, the Green's receiver functions are simulated. This needs to be done for each receiver individually by propagating a δ-pulse from the receiver into the medium. Due to the characteristics of the velocity stress solver, I use a Heaviside step function (Θ) as a force-time function, defined as

$$\Theta(t - t_{step}) = \begin{cases} 0 \text{ for } t < t_{step} \\ 1 \text{ for } t \geq t_{step}, \end{cases}$$

with the $\delta(t - t_{step})$-function as its derivative. Using the Θ-function I simulate Green's receiver functions of displacement $G_{ij}(\mathbf{x}_r, t; \mathbf{x}, t_{step})$. The time shift by t_{step} is applied to avoid filtering artefacts. I use the same t_{step} for the calculation of the forward wavefield. Of course, the correctness of the Green's receiver functions is only warranted for frequencies which fullfill the dispersion criterion (equation 3.2) and can cause strong dispersion for higher frequencies. However, this is considered in the range of the inversion frequencies, in which the functions are used.
To estimate the Green's receiver functions in frequency domain, we apply a discrete Fourier transform (equation 3.7) to the function $G_{ij}(\mathbf{x}_r, t; \mathbf{x}, t_{step})$ as described in section 3.1.5. The functions $G_{ij}(\mathbf{x}_r, \omega_v; \mathbf{x}, 0)$ are stored in memory for each discrete frequency and each receiver.
The calculation of the Green's receiver functions causes the most part of additional computational costs which occur during the calculation of the Hessian preconditioner. I therefore use two main approximations here. First, I only propagate the wavefield for a subset of receivers. This is justified, because the dominating artefacts in the gradient are located at the source positions, whereas the small receiver artefacts are far easier to correct. Second, I only propagate one component of the δ-pulse. The component I use corresponds to the force direction of the forward source. This means, that I only calculate one component of the partial derivative wavefield (e.g., for a δ-pulse in z-direction this corresponds to $\partial u_z / \partial m_j$). This generally leads to a good correction at the source, but some artefacts remain at the receivers, where other wavefield components are abundant. Using these two approximations, the Green's receiver functions can be calculated and stored at reasonable computational costs.

Partial derivative wavefields and Hessian computation

The forward wavefields for each shot are already computed and transformed to the Fourier-domain in the framework of the gradient calculation (section 3.2). After the calculation of each shot, I calculate the partial derivative wavefields according to equation 5.6 for this shot. For each receiver and each discrete frequency the forward wavefield is multiplied with the corresponding Green's receiver function. The forward wavefield is stored as velocities v_i and integrated with $(i\omega)^{-1}$ (equation 3.8) to find the corresponding displacements u_i. Spatial derivatives of forward field and Green's receiver functions are calculated with the use of finite-differences, similar to the implementation of gradient calculation (section 3.3). The partial derivative wavefields are then multiplied with their conjugate value and summed up (equation 5.8). When this is done for all shots, this results in an approximation of the diagonal Hessian operators $H_a^{v_p}(\mathbf{x})$, $H_a^{v_s}(\mathbf{x})$ and $H_a^{\rho}(\mathbf{x})$.

5.3.2 Hessian preconditioning

Finally, the preconditioning operators $P^{v_p}(\mathbf{x})$, $P^{v_s}(\mathbf{x})$ and $P^{\rho}(\mathbf{x})$ are calculated by adding a water level to the Hessian approximations and by using their reciprocal value (equations 5.9). At the moment, the water level is chosen empirically. Hereby, it is chosen large enough to avoid large artefacts in areas of low illumination, i.e., in large depth or near the boundaries. These operators are then used for gradient preconditioning:

$$\mathbf{p}_k(\mathbf{x}) = P(\mathbf{x})\nabla E_k. \tag{5.11}$$

Additional local preconditioning at the receiver positions according to section 3.4 is required due to the reduced number of Green's receiver functions used for the Hessian calculation. Furthermore, I typically taper the C-PML boundaries.

Overall, compared to the Gauss-Newton method, I make the following approximations:

- use of diagonal elements of $\mathbf{H}_a$ only,
- employ only a subset of receivers for its calculation,
- only one component of partial derivative wavefields by backpropagating only one component δ-pulse and
- estimate Hessian only once for each frequency stage.

In the following I will show two examples of 3D FWI with Hessian preconditioning: a simple toy example to validate the approach and a surface geometry example to show its performance in a more complex application.

5.4 Transmission geometry - the subdivided box model

Even though the implementation of the Hessian preconditioning is of main interest for its application to FWI of complex models, I choose a relatively simple test in transmission geometry to show its effects on the gradients and to prove its performance in the inversion.

5.4.1 Model and inversion setup

I consider a box model with the size of $160\times160\times184$ grid points corresponding to 128 m$\times$128 m $\times$147.2 m in x-, y- and z-direction. The true models for the seismic velocities v_p and v_s are shown in Figure 5.2. The seismic velocity models contain a box, which is divided into four differently-sized parts with different positive and negative velocity variations. The v_p/v_s ratio is not constant. The model does not contain a free surface. As starting models, the homogeneous background velocities outside the box are used. The density model is homogeneous with 2800 kg/m^3 and kept constant during inversion.
Sources and receivers are arranged within x-y-planes, as indicated in Figure 5.2. We use 12 (3×4) sources in 92 m depth and 169 (13×13) receivers in 24 m depth. The sources are vertical directed point forces with $\sin^3$-wavelets as source time functions and a source duration of

0.0033 s. This source wavelet comprises frequencies up to about 400 Hz.

In total, I performed 80 iterations, divided into 4 different frequency stages ranging from

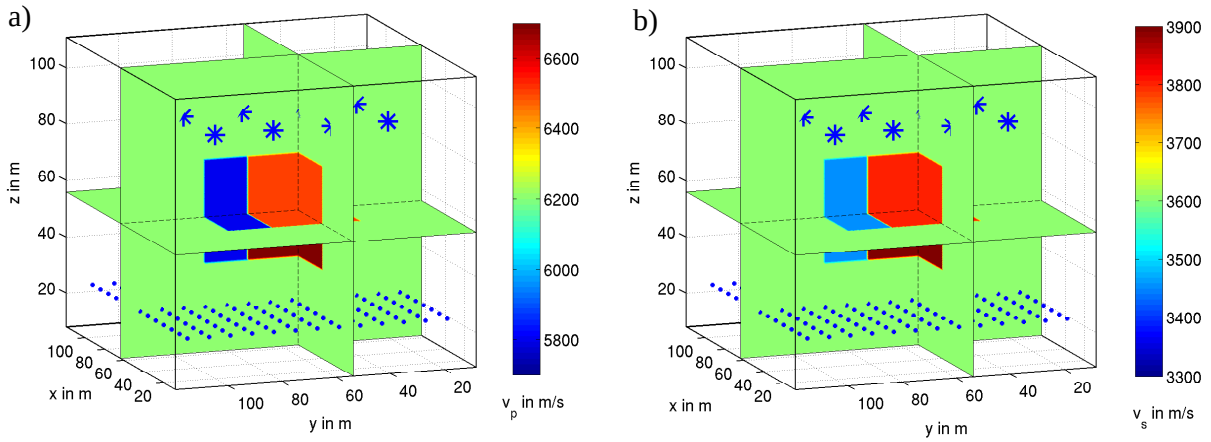

Figure 5.2: Real box model of v_p (a) and v_s (b) with indication of sources (stars) and receivers (crosses).

160 Hz to 320 Hz and use five frequencies in each stage (see Table 5.1). The minimum wavelengths for v_p and v_s corresponding to the maximum frequency in each stage are also given. They correspond to the maximum resolution which is expected in this inversion. The gradients are generally normalised to their maximum value and scaled by the homogeneous background velocities. To account for the different wavelength of v_p and v_s I additionally apply empirical scaling factors to improve the multiparameter inversion. As I expect a predominant influence of v_s for the first frequency stage I apply a scaling factor of 0.7 to reduce the v_p update. In the third and fourth frequency stage the v_s update is scaled by 0.7 and 0.4, respectively. Here only small changes are required to get the fine structures at small wavelengths, whereas the larger P-wavelengths still allow for rougher changes in the model.

At the beginning of each frequency stage, the diagonal Hessian preconditioning matrices $\mathbf{H}_a^{v_p}$

iteration	frequencies in Hz	$\lambda_{min}(v_p)$ in m	$\lambda_{min}(v_s)$ in m
1-20	160,170,180,190,200	31	18
21-40	200,210,220,230,240	26	15
41-60	240,250,260,270,280	22	13
61-80	280,290,300,310,320	19	11

Table 5.1: Frequency stages used for the transmission geometry box example with minimum wavelengths $\lambda_{min}(v_p)$ and $\lambda_{min}(v_s)$.

and $\mathbf{H}_a^{v_s}$ were calculated according to the workflow (Figure 5.1) described in the last section. In the framework of the Hessian calculation a vertically directed δ-pulse was propagated from each receiver to find the receiver Green's functions. These Hessian approximations were used for preconditioning of the seismic velocity gradients within the whole frequency stage.

With the use of four sources for the steplength estimation the general conjugate gradient approach requires 32 forward modellings per iteration. The calculation of the Hessian preconditioning matrices amounts to 169 additional forward modellings once in each frequency stage.

Note, that in this example I calculate the Green's receiver functions for all receivers. Thus, for the full inversion 676 forward modellings are added to the 2560 forward modellings of the general gradient approach, which is an increase of about 26% of runtime in this example.

5.4.2 Effects of Hessian preconditioning

Figure 5.3 shows the effects of the diagonal Hessian preconditioning on the gradients of v_p and v_s for the first iteration, and thus for the first frequency stage. The gradients before preconditioning, normalised to their maximum values, are shown in Figure 5.3a) for v_p and b) for v_s. Note, that the colorbar is strongly clipped to make the general gradient structure observable. The high amplitudes around sources and receivers are clearly visible. Without preconditioning, the model update is only significant within these areas, and the inversion fails. Figure 5.3c) and d) shows the logarithm of the normalised diagonal Hessian approximation $\mathbf{H}_D$ for v_p and v_s. The Hessian matrix covers several orders of magnitude and, like the gradient, it shows extremely high values at source and receiver positions. The influence of the geometric amplitude decay of the wavefield is clearly visible. Areas with no or very low wavepath coverage show very low values. This is, for example, visible in the blue areas of the v_p Hessian. The application of the inverse Hessian in such areas would lead to an enormous enhancement of the gradient, even though we have no or very little information in our data. To avoid this, the water level ε is added to the Hessian approximation, which I determine manually once for each Hessian approximation.
The preconditioning operators $P(\mathbf{x})$ are calculated according to equation 5.9. The normalised gradients after preconditioning are shown in Figure 5.3e) and f) for v_p and v_s, respectively. The high amplitudes near source and receiver positions are well corrected. Very small receiver artefacts are still visible, which results from my approximation to use only the vertical component of the partial derivative wavefields for the Hessian calculation. Hence, some additional damping at receiver positions might be required and is applied locally during inversion. The highest amplitude in the gradient now concentrates on the box area. The structure of the preconditioned gradient within the box area is still very similar to the structure of the gradient before preconditioning. Thus, for this simple transmission geometry application, the effects of Hessian preconditioning are mainly limited to the source and receiver area. Unfortunately, some artefacts above the source plane are visible in the gradients. However, the inversion results show, that these gradient artefacts do not manifest in the model.

5.4.3 Inversion results

In figure 5.4 the data fit of observed, starting and final inverted data is plotted for the x-, y-z-component for one representative source and receiver normalised to one trace. The data is lowpass-filtered with a corner frequency of 320 Hz, which is the maximum frequency used for the inversion. It is visible, that the seismograms of the homogeneous starting model are already relatively close to the data. The final inverted data and the observed data show a nearly perfect fit.
The final models of v_p and v_s are shown as two 2D slices: a horizontal (x-y) slice in Figure 5.5 and a vertical (y-z) slice in Figure 5.6. For comparison, the real models are plotted. Overall, the box compartments could be successfully reconstructed by the FWI. In the horizontal slice of the final inverted models in Figure 5.5 b) and d) the three sub-boxes are successfully recovered

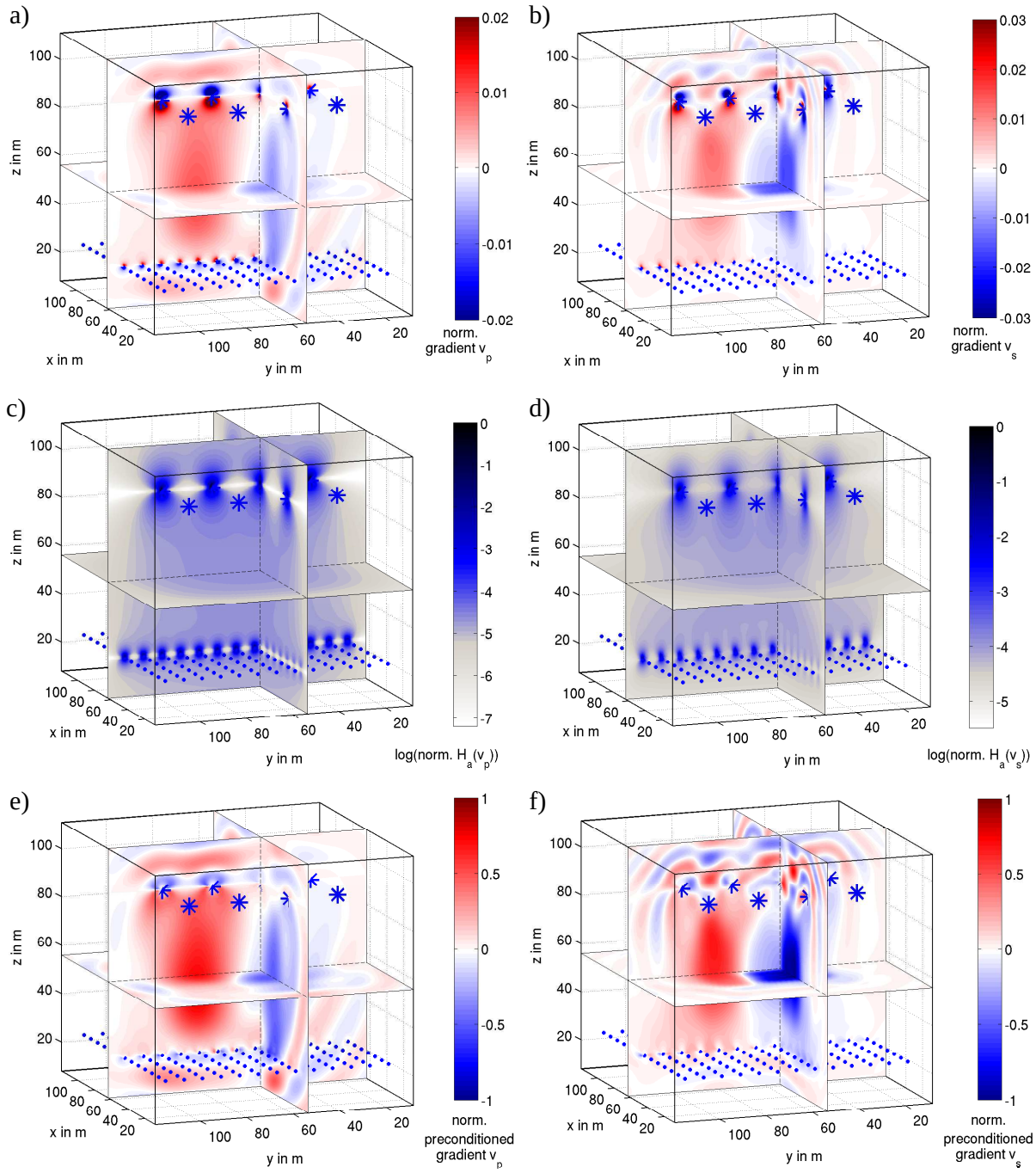

Figure 5.3: Effects of Hessian preconditioning on the gradients of v_p and v_s in the box model: a) normalised gradient v_p, b) normalised gradient v_s, c) logarithm ($\log_{10}$) of normalised Hessian $\mathbf{H}_D(v_p)$, d) logarithm of normalised Hessian $\mathbf{H}_D(v_s)$, e) normalised preconditioned gradient v_p and f) normalised preconditioned gradient v_s.

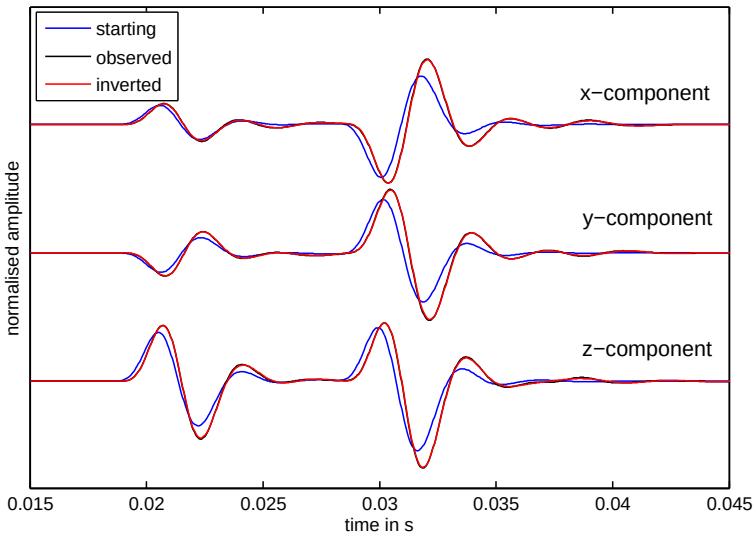

Figure 5.4: Comparison of starting, observed and final inverted multi-component waveforms filtered below 320 Hz for exemplary receiver at (x_r, y_r, z_r)=(64 m, 64 m, 24 m) and source at (x_s, y_s, z_s)=(96 m, 32 m, 92 m) normalised to one trace.

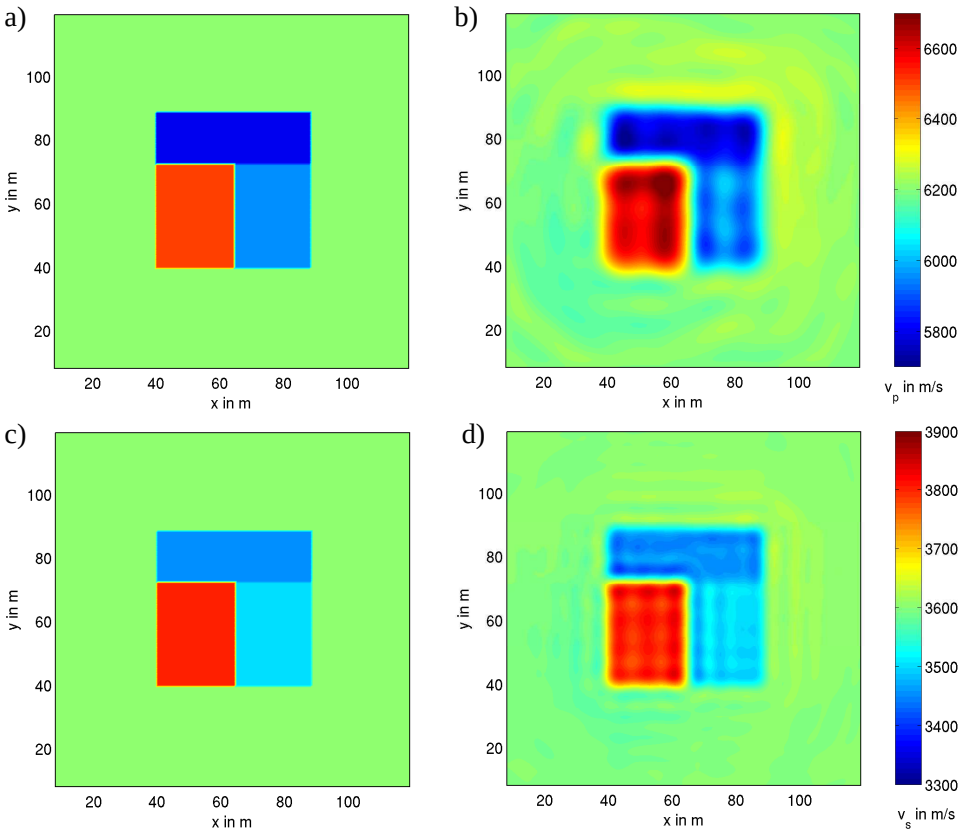

Figure 5.5: Results of box model inversion for horizontal slice at $z = 60$ m with a) real model v_p, b) inverted model v_p, c) real model v_s and d) inverted model v_s.

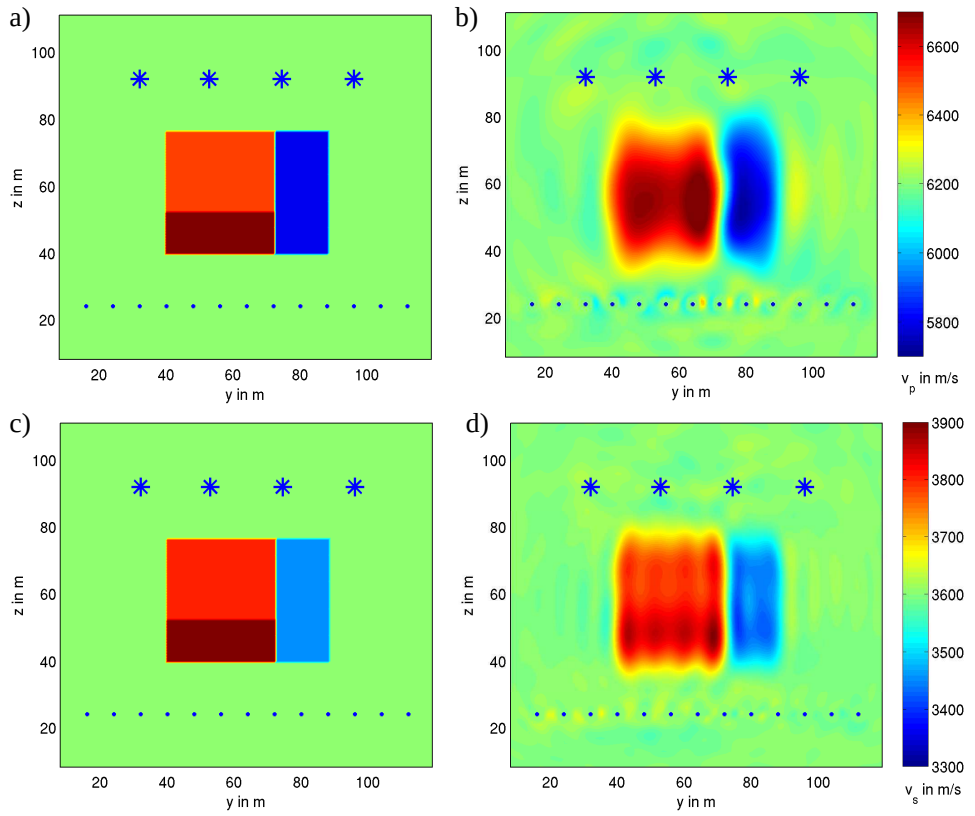

Figure 5.6: Results of box model inversion for vertical slice at $x = 56\,\text{m}$ with a) real model v_p, b) inverted model v_p, c) real model v_s and d) inverted model v_s.

by the inversion. Due to the smaller wavelengths of the shear wave, the shape of the sub-boxes is clearer in v_s, compared to v_p.
The results plotted in the vertical slice are not as well resolved. The 12 m thick high-velocity compartment on the left side is only partly resolved in v_s (Figure 5.6d) and is not visible in the v_p model (Figure 5.6b). This is not unexpected when looking at the minimum wavelengths of 11 m for v_s and 19 m for v_p within this high-velocity zone. Higher frequencies would be required for a better reconstruction.

5.4.4 Comparison with local preconditioning

For comparison, I repeated the inversion with the same data and inversion parameters, such as frequency stages, number of iterations and and test steplengths but used a local preconditioning. Circular tapers were applied to damp the high amplitudes around sources and receivers, as described in section 3.4. Figure 5.7 shows the evolution of the misfit for both methods of preconditioning. The misfit values are normalised to the initial misfit and plotted logarithmically. The steps in the misfit curve are caused when adding new data at the start of a frequency stage. Overall, both curves show a very good convergence in all four frequency stages. Finally, the misfit in the frequency band up to 320 Hz could be reduced to 0.4% for the Hessian preconditioning and 0.3% of the initial misfit in case of local preconditioning. Comparing the two methods, the misfit curves behave nearly similar, even though misfit values are slightly lower

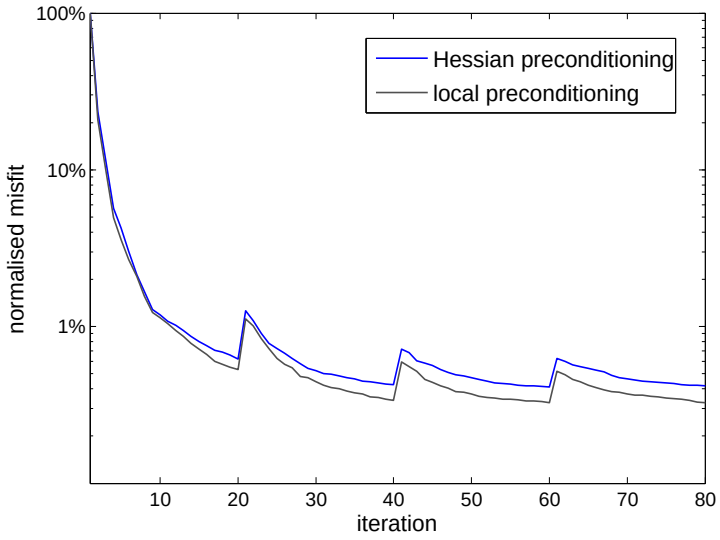

Figure 5.7: Misfit proceedings for box model normalised to the initial misfit. FWI- preconditioned with Hessian approximation compared to a local tapering.

for local preconditioning. Thus, no improvement of the convergence could be achieved in this example with the use of the Hessian. A comparison of the final models also results in no clear differences between the two approaches. Note, however, that this is a very simple example, where sources and receivers are located in homogeneous and known material. A more distinct change is expected for more complex problems.

Overall, this simple example showed, that the method of Hessian preconditioning can successfully account for high amplitudes around sources and receivers in the gradients and allow a satisfactory inversion. Still, the effects of Hessian preconditioning in this example are mainly concentrated around sources and receivers. Applying a local tapering in these areas results in a similar convergence and in a similar inversion result, which is computed much cheaper. Thus, well-posed inverse problems might not profit from preconditioning with the diagonal Hessian approximation.

5.5 Surface geometry - the 3D layered model

Most seismic applications are conducted with sources and receivers located at the surface in so-called surface or reflection acquisition geometries. Differently to transmission geometry experiments, where the direct wave already carries a lot information about the underground structures, the subsurface in reflection geometry experiments is commonly inferred from reflected, refracted and diffracted waves. Furthermore, due to the existence of a free surface, the wavefield also contains high amplitude surface waves. This complex wavetype increases the nonlinearity of the inversion. However, it does contain useful information about the shallow subsurface with a particular sensitivity to the shear wave velocity (Forbriger, 2001).
The energy of the seismic wavefield strongly decreases with depth due to the geometric amplitude spreading, due to the loss off energy, which is reflected and diffracted back to the surface and due to the strong amplitude decay of the surface waves with depth. Therefore, the FWI, especially the reconstruction of deeper structures, requires a careful preconditioning. In this

section I will present an inversion of a 3D layered model with surface acquisition geometry. For preconditioning, the approximate Hessian, as discussed in this chapter, is applied.

5.5.1 Model and acquisition geometry

I consider a 3D layered onshore surface model as shown in Figure 5.8 for v_p (a) and v_s (b) and ρ (c). The velocity model consists of homogeneous sedimentary layers over a basin-shaped homogeneous half-space. The size is $320\times320\times160$ grid points which corresponds to 256 m$\times$256 m $\times$128 m. At $z = 0$m, the model contains a free surface. The structures in v_p and v_s are similar with v_p/v_s ratios between 1.67 and 1.80. The third layer (white) is a low velocity layer. I do not invert for density in this example, but use the real density model as starting model and keep it constant during inversion. In order to avoid the introduction of hard contrasts in the inversion of the seismic velocities due to the density, the density model is built as a smoothed version of the seismic velocity structures. In order to get this model I used the SMOOTH3D function of Seismic Unix, which is based on a damped least squares approach (Liu, 1994) with an operator length of 32 m.
The acquisition geometry can be seen in Figure 5.9. 49 vertically directed point sources (red stars) are located in 3 m depth and in total 3025 receivers (black crosses) are located in 1.6 m depth. Only a subset of 100 receivers (blue crosses) is used for the calculation of the Hessian.
As source time function, I apply a $\sin^3$ function with a dominant frequency of 20 Hz. Its spectra is plotted in Figure 5.9. It contains frequencies up to about 50 Hz and a main advantage of this wavelet is, that it also comprises the very low frequencies.
Seismograms of the vertical component are plotted for one exemplary shot along a x-profile and a y-profile in Figure 5.11a) and b), respectively. The waveforms show the predominant influence of the high amplitude Rayleigh wave, whereas direct, reflected and refracted body waves only show with low amplitude and cannot be clearly identified. The waveforms along the x-profile and y-profile look relatively similar.

5.5.2 Inversion setup

I inverted for the seismic velocities v_p and v_s. The starting models for the seismic velocities are smoothed versions of the real models (Figure 5.12). They result from applying the SMOOTH3D Seismic Unix function with an operator length of 32 m to the real models. Thus, only rough structures of the real model are preserved. The low velocity zone is smoothed out for the greater part.

The first frequency band used during inversion is from 2 Hz to 4 Hz. Starting at these low frequencies was necessary for the inversion to succeed. The seismograms along a y-profile for one representative source filtered below 4 Hz are plotted in Figure 5.13. It is obvious, that there are already substantial differences between the data of the smoothed starting model and the observed data in the first frequency band.

The different frequency stages are given in Table 5.2. In total, 170 iterations were performed using six frequency stages, with frequencies ranging from 2 Hz and 22 Hz. In each frequency stage six frequencies were employed and the frequency interval was increased to higher stages. The minimum wavelengths for P- and S-wave are also given in the table. A resolution down to $\lambda/2$ can be reached in the ideal case.

For preconditioning I apply the Hessian preconditioning. Its effects are discussed in more

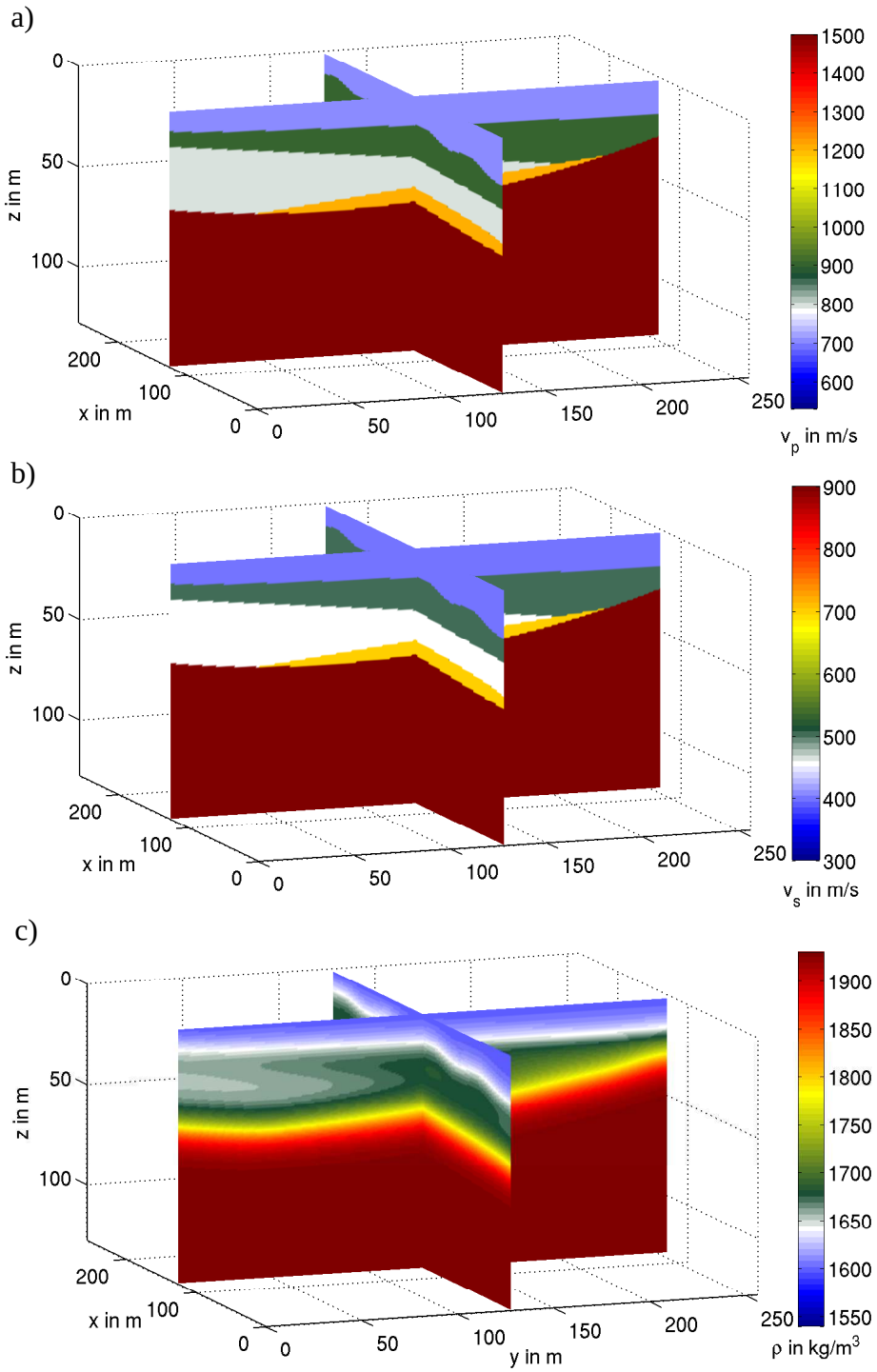

Figure 5.8: Real 3D surface models: a) v_p, b) v_s, c) ρ.

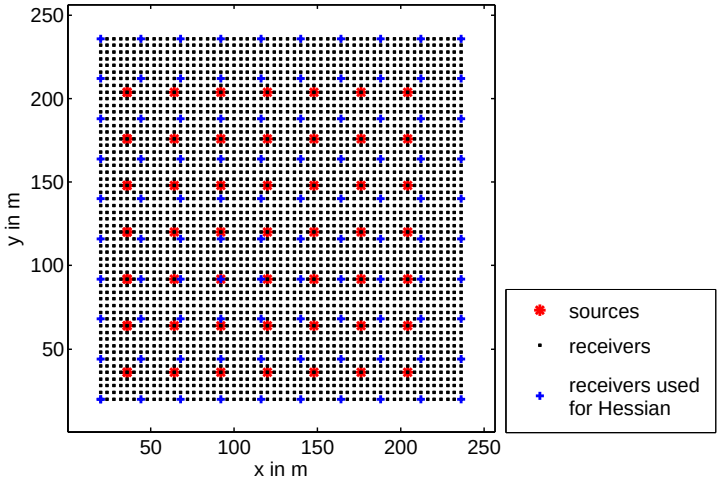

Figure 5.9: Top view of the surface acquisition geometry showing sources, all receivers and the subset of receivers used for the Hessian calculation.

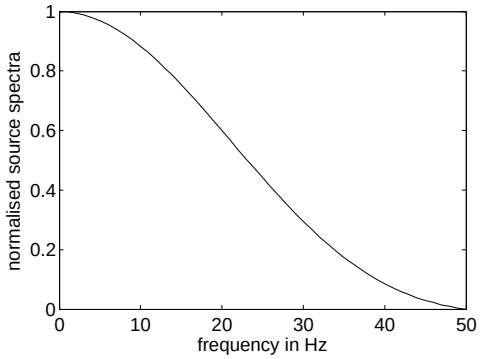

Figure 5.10: Source spectra of a 20 Hz $\sin^3$-wavelet used for the surface application.

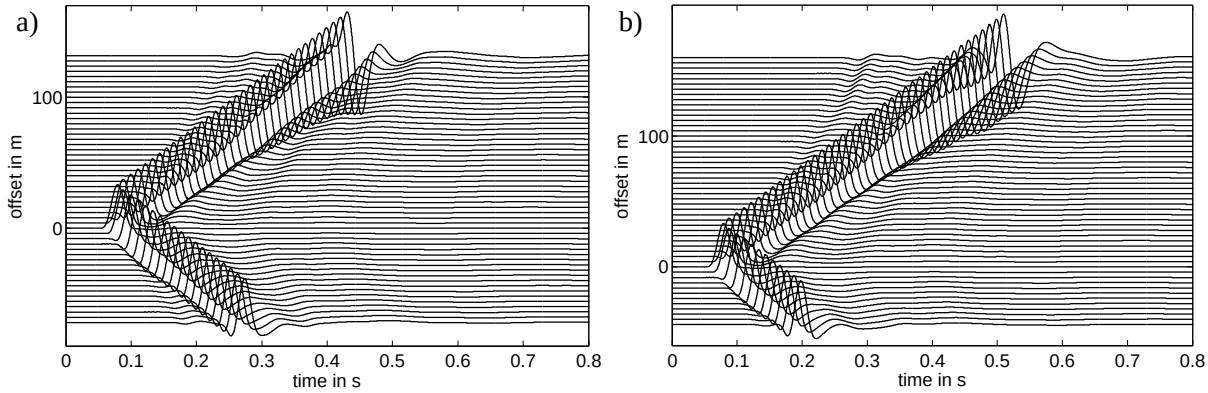

Figure 5.11: Observed unfiltered data of vertical component (trace normalised) shown for shot at x=92 m and y=64 m for receivers along two perpendicular profiles: a) x-profile at y=64 m and b) y-profile at x=92 m.

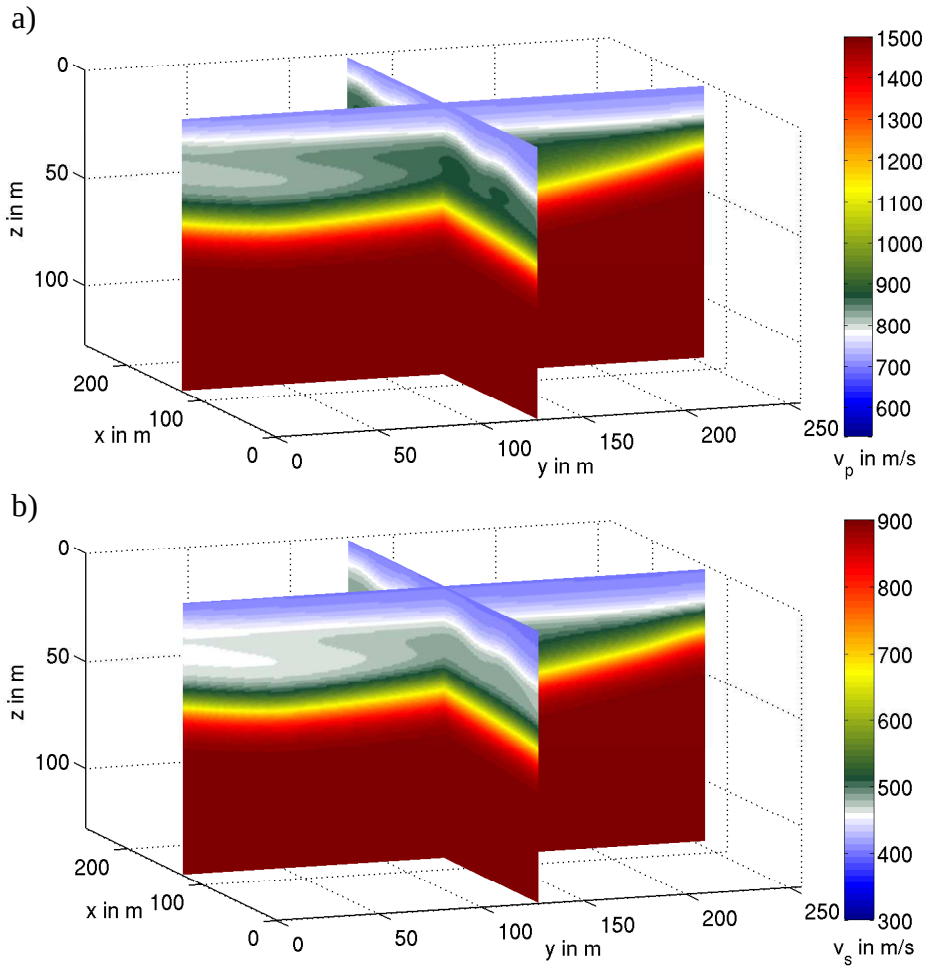

Figure 5.12: Smoothed versions of the real surface models used as starting models for FWI: a) v_p, b) v_s.

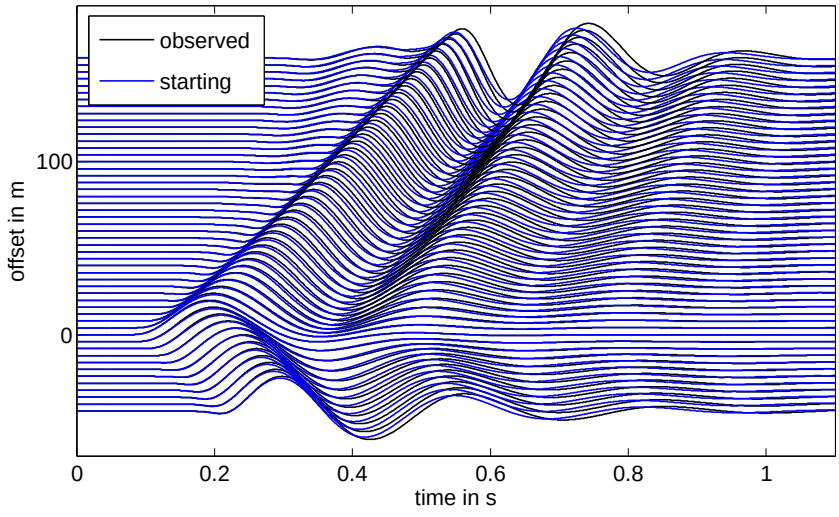

Figure 5.13: Vertical component of observed versus starting data lowpass-filtered with 4 Hz (trace normalised) shown for source at x=92 m and y=64 m along y-profile at x=92 m.

iteration	frequencies in Hz	$\lambda_{min}(v_p)$ in m	$\lambda_{min}(v_s)$ in m
1-29	2.0 2.4 2.8 3.2 3.6 4.0	175	100
30-62	4.0 4.5 5.0 5.5 6.0 6.5	108	62
63-85	6.5 7.0 7.5 8.0 8.5 9.0	78	44
86-114	9.0 9.6 10.2 10.8 11.4 12.0	58	33
115-141	12.0 13.0 14.0 15.0 16.0 17.0	41	24
142-170	17.0 18.0 19.0 20.0 21.0 22.0	32	18

Table 5.2: Frequency stages used for the surface geometry application with minimum wavelengths $\lambda_{min}(v_p)$ and $\lambda_{min}(v_s)$.

detail in the next section. Additionally, the vertical C-PML-boundaries are not updated to avoid artefacts in these areas. To prevent implausible contrasts from the inner model to the boundary, all boundary parameters in a certain depth are set to an average value of the grid point adjacent to the boundary. This average value is calculated as the mean value of all directly neighbouring gridpoints.
Only a subset of the receivers is used for the Hessian calculation. Therefore, an additional local preconditioning is applied at the receiver positions. I do not apply any additional smoothing to the gradients and models during the inversion.

5.5.3 Effects of Hessian preconditioning

The Green's receiver functions for the Hessian are calculated for a subset of 100 receivers and with vertically directed δ-pulse forces, only. Thus, 100 additional forward modellings are performed for an approximate Hessian estimation compared to the gradient calculation. The Hessian approximations are calculated once at the beginning of each frequency stage and then applied for gradient preconditioning in this stage.
The effects of the Hessian preconditioning on the gradients are shown for two exemplary iterations in the Figures 5.14 and 5.15. In Figure 5.14 a) and b) the gradients for v_p and v_s, normalised to their maximum value, are plotted for the first iteration and the lowest frequency band (2 Hz - 4 Hz). The high amplitude values adjacent to the sources are clearly visible. Still, due to the strongly clipped colorbar, gradient structure in greater depth can also be seen. The corresponding Hessian matrices (Figure 5.14 c) and d)) are plotted logarithmically. They cover several orders of magnitude, and contain information about the acquisition geometry and the structure of the starting model. The calculation of the preconditioning matrix $P(\mathbf{x})$ (equation 5.9) and its application to the gradients results in the preconditioned gradients presented in Figure 5.14 e) and f). It is visible, that the source artefacts are now successfully removed and that the main model update concentrates on the middle and deeper layers of the basin. Still, except for the area around the sources the effects of the Hessian preconditioning are small and the structure of the preconditioned gradients is very similar to the unpreconditioned gradients. The preconditioned gradients still show some artefacts in the boundary and at the receiver positions, so that additional preconditioning in these areas is useful.
The effects of Hessian preconditioning can be more substantial as I present in the second example for the third frequency band (6.5 Hz - 9 Hz). In this case, the gradients (Figure 5.15 a) and b)) show a very strong amplitude decay with depth. This amplitude decay is probably mainly caused by the presence of the surface wave. A model update without a correction of the

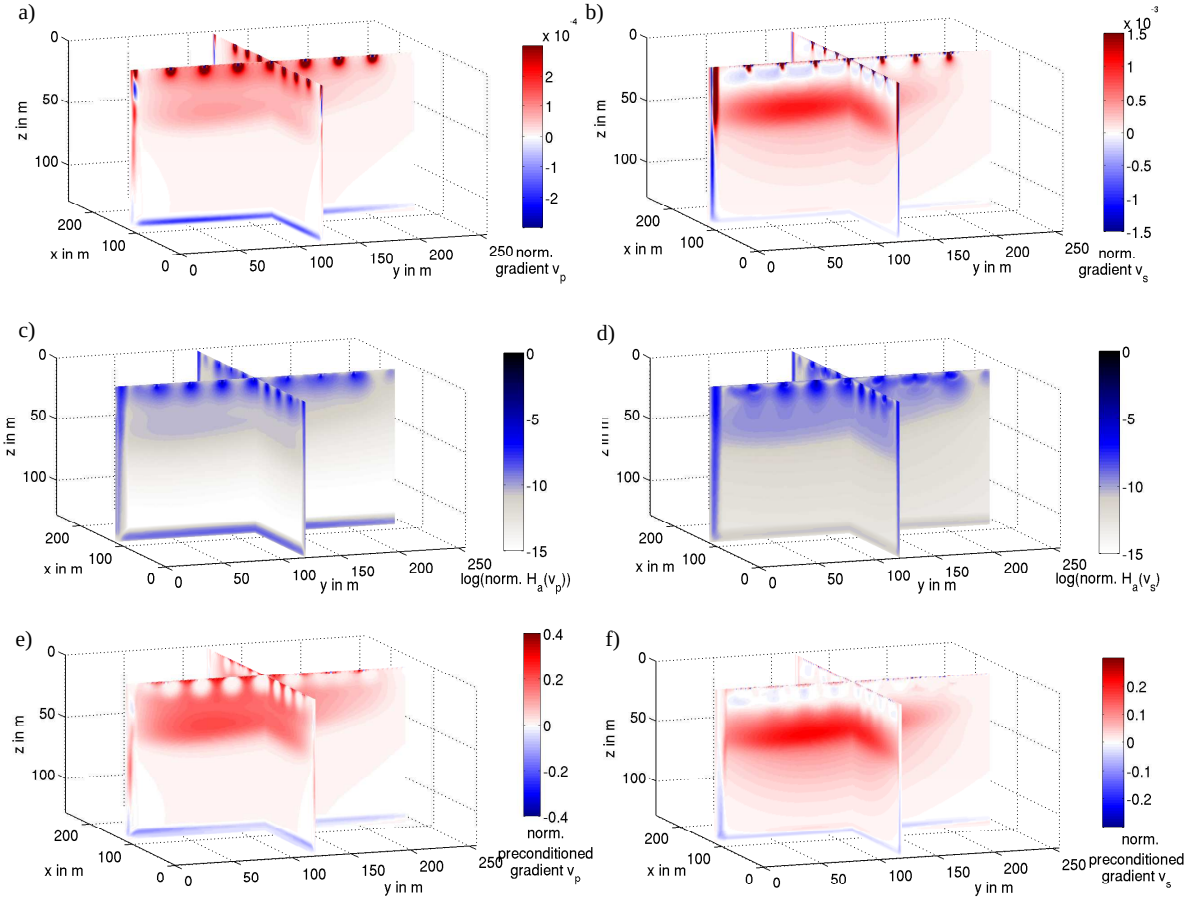

Figure 5.14: Effects of Hessian preconditioning on the gradients of v_p and v_s in the surface model for the first iteration (first frequency stage, 2-4 Hz): a) normalised gradient v_p, b) normalised gradient v_s, c) logarithm of normalised Hessian $\mathbf{H}_D(v_p)$, d) logarithm ($\log_{10}$) of normalised Hessian $\mathbf{H}_D(v_s)$, e) normalised preconditioned gradient v_p and f) normalised preconditioned gradient v_s.

gradient for this depth decay would thus affect only the most shallow part of the model. After applying the Hessian preconditioning with the Hessian operators shown in Figure 5.15 c) and d) the gradients looks very different (Figure 5.15 e) and f). Deeper structures in the gradients now dominate, whereas the high amplitudes in the shallow part vanish. Hereby the right choice of the water level used in equation 5.9 is important. If the water level is too small, artefacts in the gradient, especially in the deeper part of the model, are emphasised. By increasing the water level, the effects of Hessian preconditioning decrease and the preconditioned gradients approach the unpreconditioned gradients. Although the water level is necessary, its optimal estimation is still a main difficulty in Hessian preconditioning.

5.5.4 Inversion results

Inversion progress

The evolution of the misfit value is plotted in Figure 5.16. The values are normalised to the total misfit between starting data and observed data for the full frequency range up to 22 Hz. Using only frequencies up to 4 Hz, the amplitude values of the wavefields and the correspond-

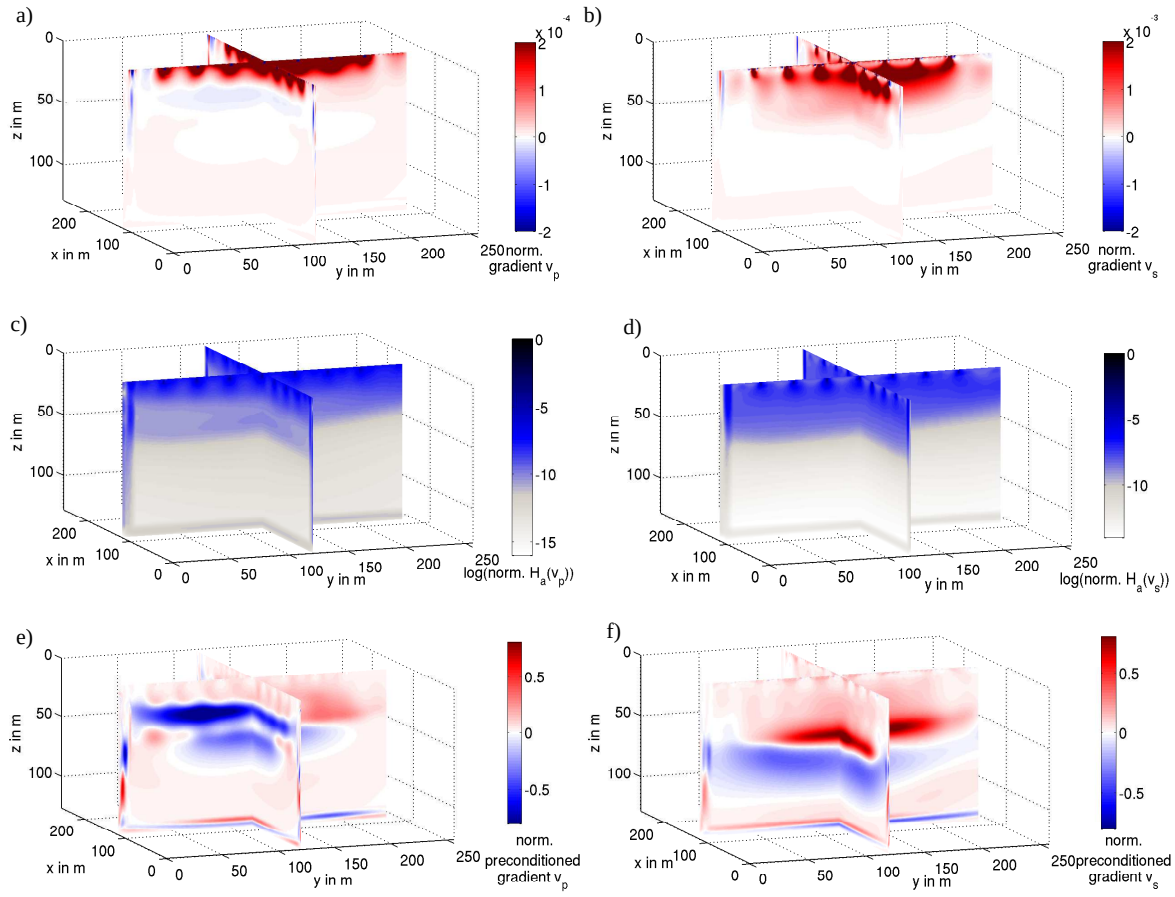

Figure 5.15: Effects of Hessian preconditioning on the gradients of v_p and v_s in the surface model for iteration 63 (third frequency stage, 6.5-9 Hz): a) normalised gradient v_p, b) normalised gradient v_s, c) logarithm of normalised Hessian $\mathbf{H}_D(v_p)$, d) logarithm ($\log_{10}$) of normalised Hessian $\mathbf{H}_D(v_s)$, e) normalised preconditioned gradient v_p and f) normalised preconditioned gradient v_s.

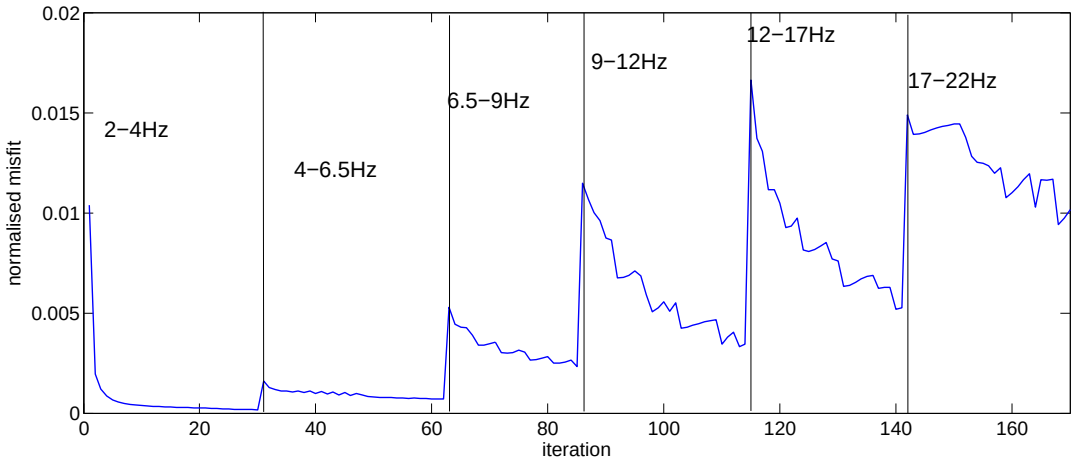

Figure 5.16: Misfit proceedings of the surface model inversion for the different frequency stages. Misfit values are normalised to the misfit between observed and starting data of the full frequency range up to 22 Hz.

ing misfit values are low for the first frequency stage. The inversion converges very rapidly and smoothly. Several characteristics are visible over the following frequency stages. The higher frequency stages show slower convergence than the first frequency stage. Additionally, the misfit curve gets increasingly rougher. This means, that at higher frequencies, the inversion does not always find a steplength to decrease the misfit, even though an overall convergence to lower misfit values is visible. The jumps in the misfit curve are especially high in the last frequency stage, which is why I decided to terminate the inversion after this stage. Looking at the final misfit values of each frequency stage, an overall increase is visible. Thus, the inversion does not succeed to explain all data added in each frequency stage. In spite of this, it converges in each stage and the final misfit value of 1% of the initial misfit for the full frequency range represents an good final data fit.
Overall, the high number of iterations, the roughness of the misfit curve, its inability to find appropriate steplengths in each iteration and the increasing misfit between the frequency stages are likely to be caused by the high nonlinearity of the surface acquisition geometry inversion which is especially caused by the dominating surface waves. This nonlinearity increases to higher frequencies. Still, the inversion successfully converged to low misfit values for frequencies up to 22 Hz.

The final data fit of the vertical component is presented exemplarily for one shot along a x- and a y-profile in Figure 5.17. Overall, the observed data is very well explained by the inversion result for both low amplitude body waves and high amplitude surface waves. Some data misfit is left over at far offset receivers.

Figure 5.18 shows exemplarily the development of the inverted S-wave velocity model over the frequency stages for one vertical slice in y-direction. The starting model is smooth, and the different layers are not clearly distinguishable. Already in the first frequency stage (b) the three upper layers of the model, including the low velocity zone (third layer) become visible. With further proceedings of the inversion the layer boundaries become more and more confined. This is also the case for the deepest interface, which develops from a broad, smooth structure at the beginning to a very defined zone at the end of the inversion.

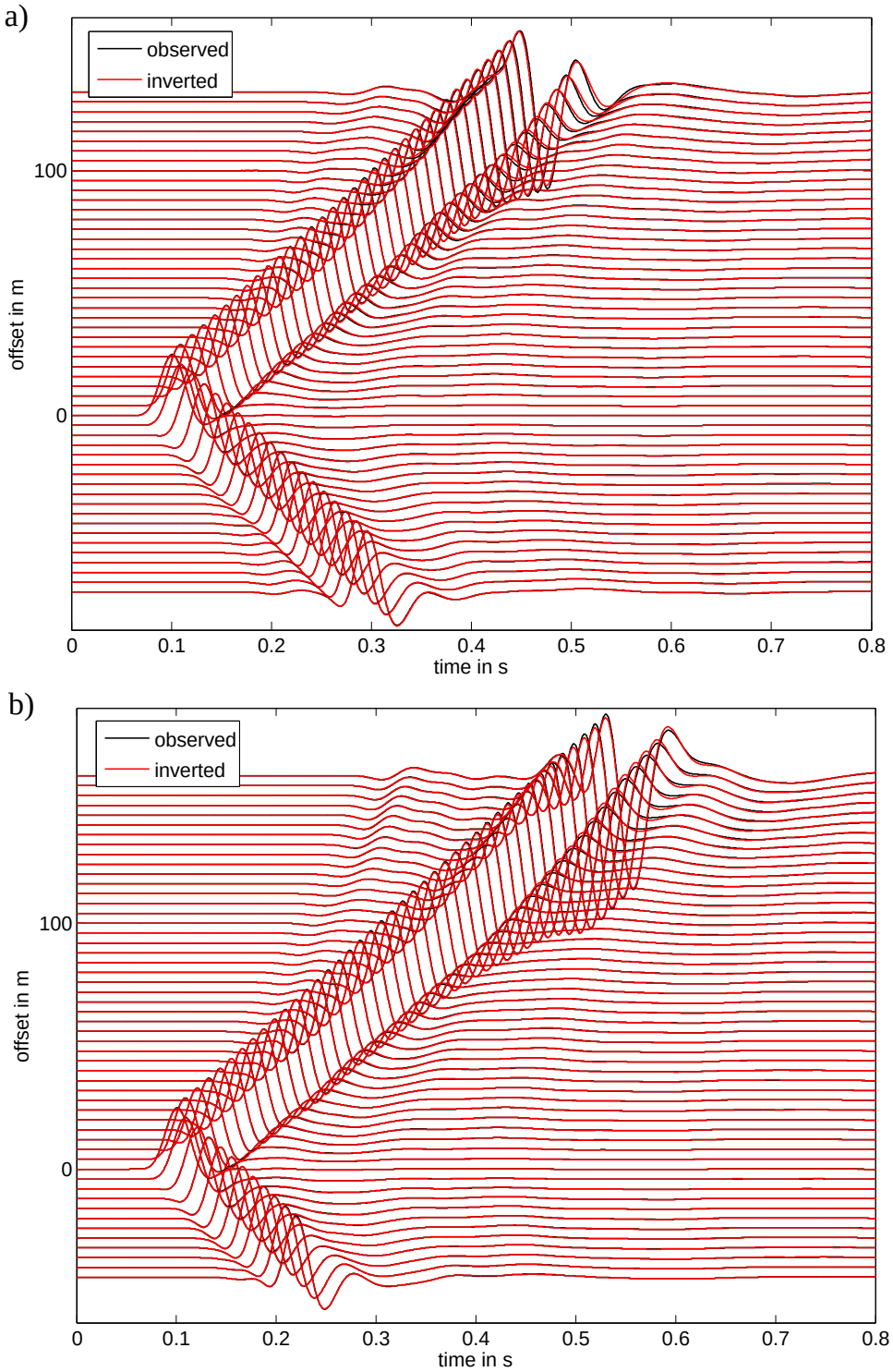

Figure 5.17: Comparison of observed and final inverted waveforms filtered below 22 Hz (trace normalised) for one exemplary shot at x=92 m and y=64 m for receivers along two perpendicular profiles: a) x-profile at y=64 m and b) y-profile at x=92 m.

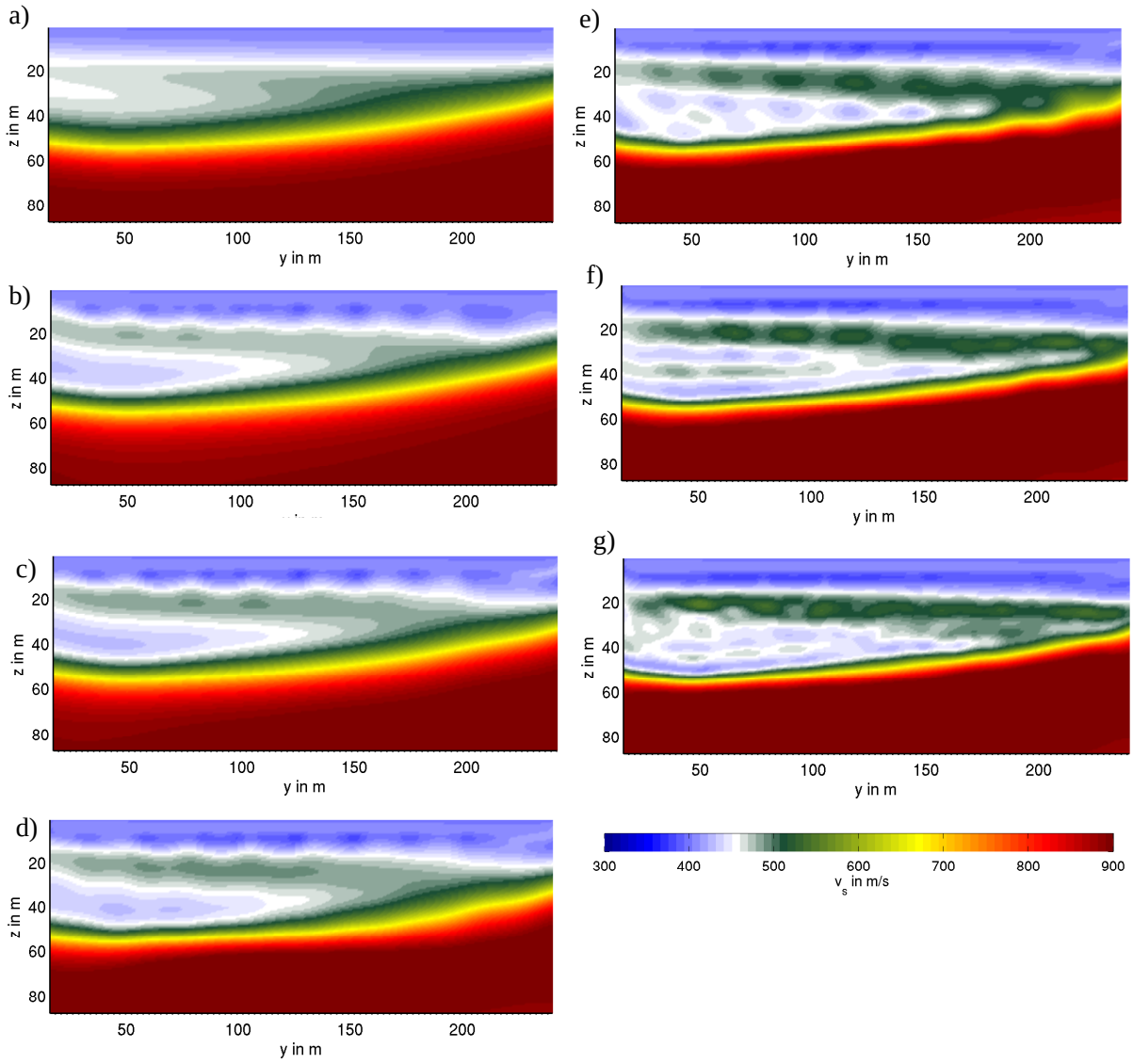

Figure 5.18: Progress of the v_s inversion result for a vertical slice of the surface model at $x = 128$ m over the different frequency stages: a) starting model, b) iteration 29 (up to 4 Hz), c) iteration 62 (up to 6.5 Hz), d) iteration 85 (up to 9 Hz), e) iteration 114 (up to 12 Hz), f) iteration 141 (up to 17 Hz) g) iteration 170 (up to 22 Hz, final inversion result).

Computational costs

Surface geometry FWIs are computationally more expensive compared to transmission geometry FWIs of models with comparable sizes for different reasons. First, the surface acquisition geometry is less optimal and requires a higher number of sources. Second, longer recording times are necessary to cover the full seismic wavefield with reflected waves and surface waves. Third, the inversion of surface seismic data is more complex and thus typically requires a higher number of iterations compared to transmission geometry data. Overall, the example shown here was computationally quite costly, which mainly resulted from the high number of 49 sources and the great number of 170 iterations. On 800 cores, each iteration, without the calculation of the Hessian, took about 1.5 hours. Hereby 118 forward simulations were performed. Additionally, 100 extra forward simulations were required for the Hessian calculations in the first iteration of each frequency stage. These iterations were more costly and took about 2.5 hours. However, compared to the total computational costs, the extra costs for the Hessian calculation are reasonable. This is of course mainly due to the approximations made for its calculation. In total, on 800 cores the inversion took about 260 hours. For future applications with higher number of shots the computational performance would profit from a shot parallelisation, which is employed additionally to the domain decomposition.

The final models

The final inverted models of the seismic velocities are presented for two perpendicular slices in the Figures 5.19 to 5.22. In Figure 5.19c) the inverted v_p result is plotted. Additionally, d) compares the real, initial and inverted models along two profiles. Their locations are indicated in the real model (a). The three upper sedimentary layers of the real model (a) are clearly reconstructed by the inversion. These layers are not present in the starting model (b). Especially the low velocity zone (grey layer) is well resolved. The high-velocity layer at the base of the basin (orange) was not resolved, however, the basin base is now much better confined. Within this frequency range it was not possible to resolve the interfaces of the real models as sharp boundaries. They show as smoothed interfaces. The amplitudes within the second layer and below the basin are overestimated.
A similar result can also be seen in the second vertical v_p slice directed in x-direction (Figure 5.19). Overall, the layers are clearly visible (except for the fourth, orange layer) and the basin base is much better confined compared to the starting model. Still, the layer boundaries are not clearly defined, but are imaged as smoothed structures in the inversion result.
A better resolution of the layers was achieved by the v_s inversion shown in the Figures 5.21 for the y-z-slice and 5.22 for the x-z-slice. The inversion results plotted in c) and d) show a nice reconstruction of the sedimentary layers, including the low velocity zone. The forth, yellow layer is not clearly resolved. Still, the upper boundary of this layer and the basin base are now much better confined compared to the starting model plotted in b). However, the velocities within the layers, especially in the second layer are not reconstructed homogeneously, but show some velocity variations. This might cause some of the problems observed in the inversion when going to higher frequencies.

To summarise, the conjugate gradient method with Hessian preconditioning was able to reconstruct the sedimentary layers of the 3D surface model successfully for both v_p and v_s. Hereby

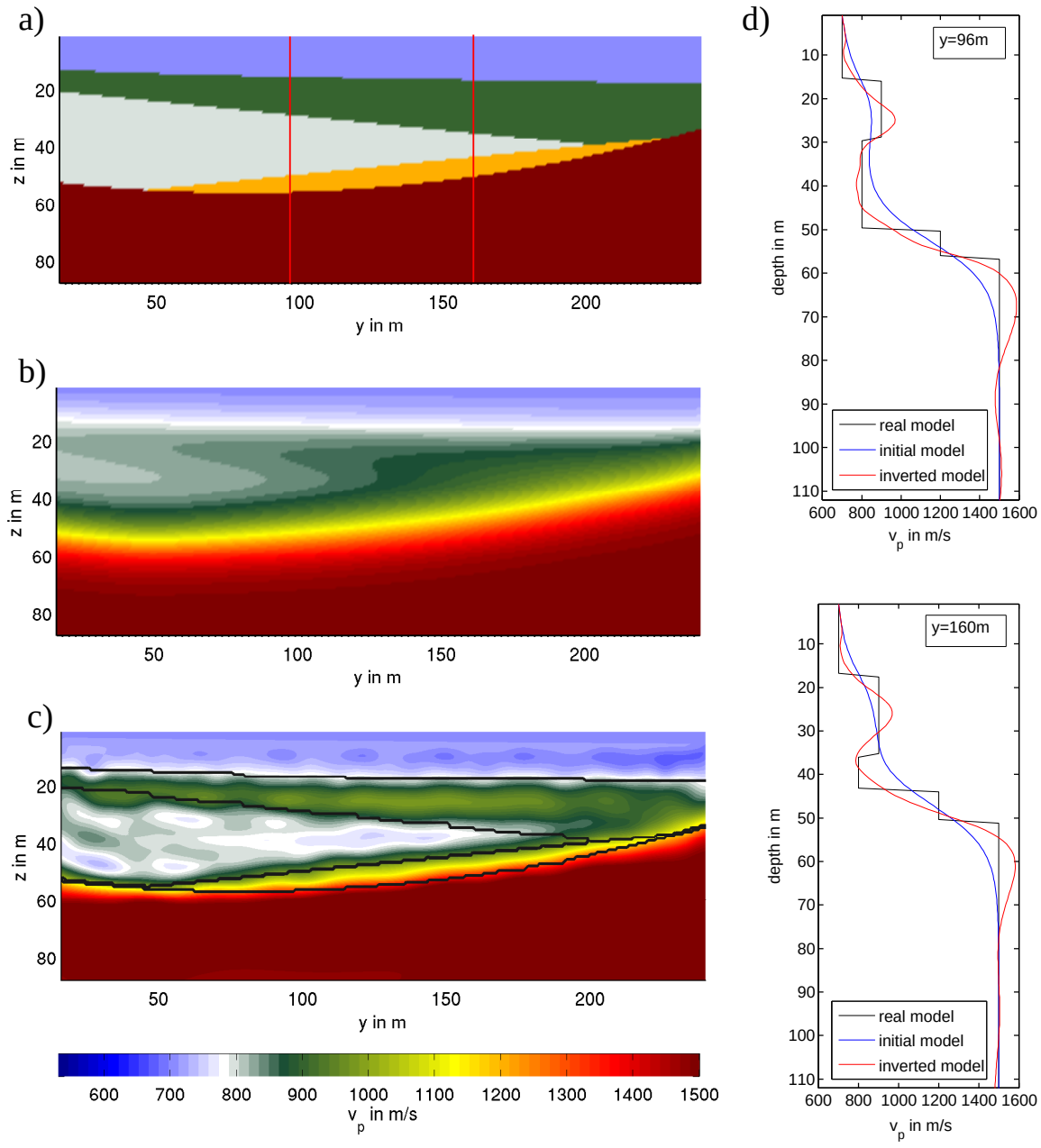

Figure 5.19: Vertical slice of the v_p surface model at $x = 128$ m: a) real model v_p, b) initial model v_p, c) inverted model v_p (170 iterations) with black lines indicating the layers of the real model, d) comparison of real, initial and starting models along vertical profiles highlighted by red lines in a).

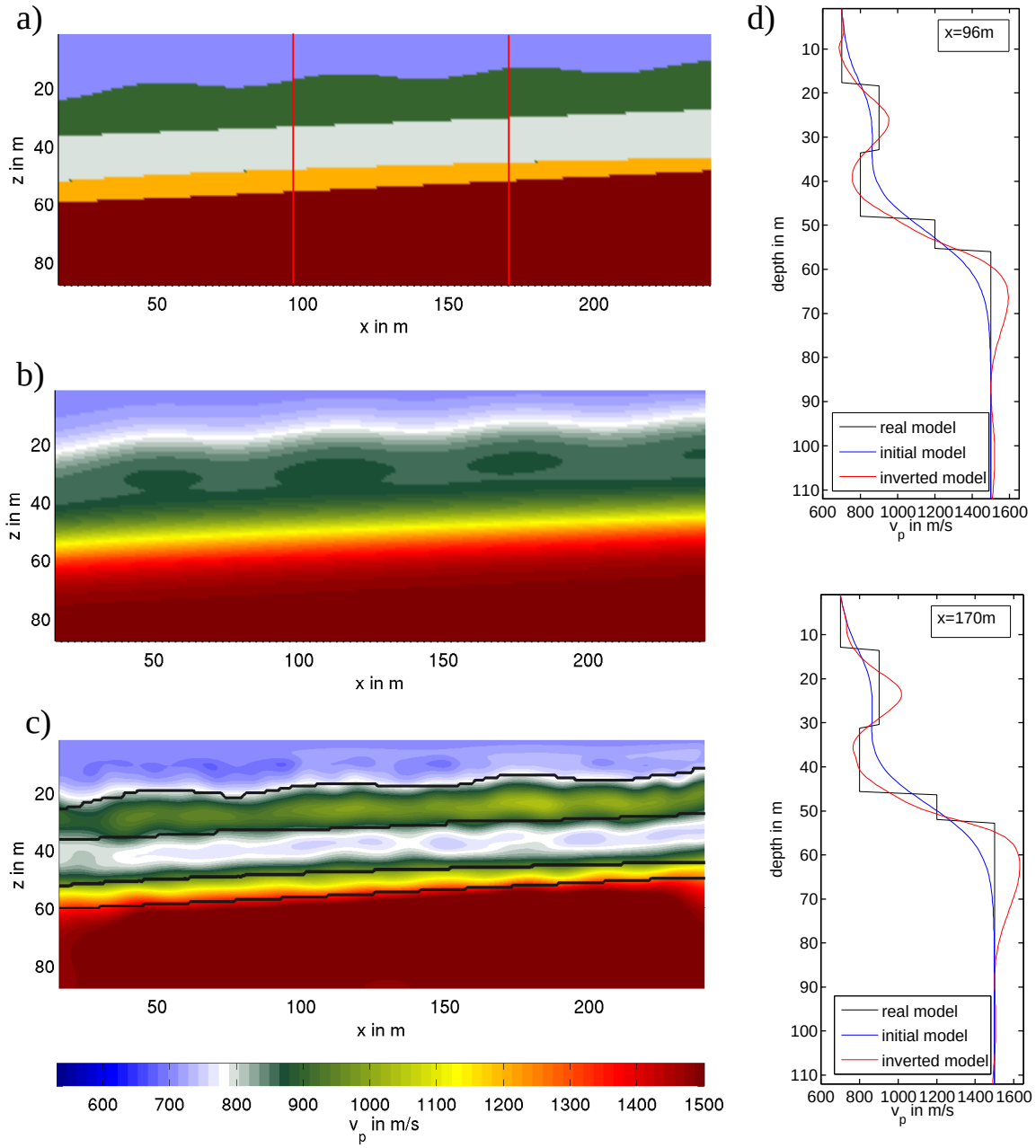

Figure 5.20: Vertical slice of the v_p surface model at $y = 128$ m: a) real model v_p, b) initial model v_p, c) inverted model v_p (170 iterations) with black lines indicating the layers of the real model, d) comparison of real, initial and starting models along vertical profiles highlighted by red lines in a).

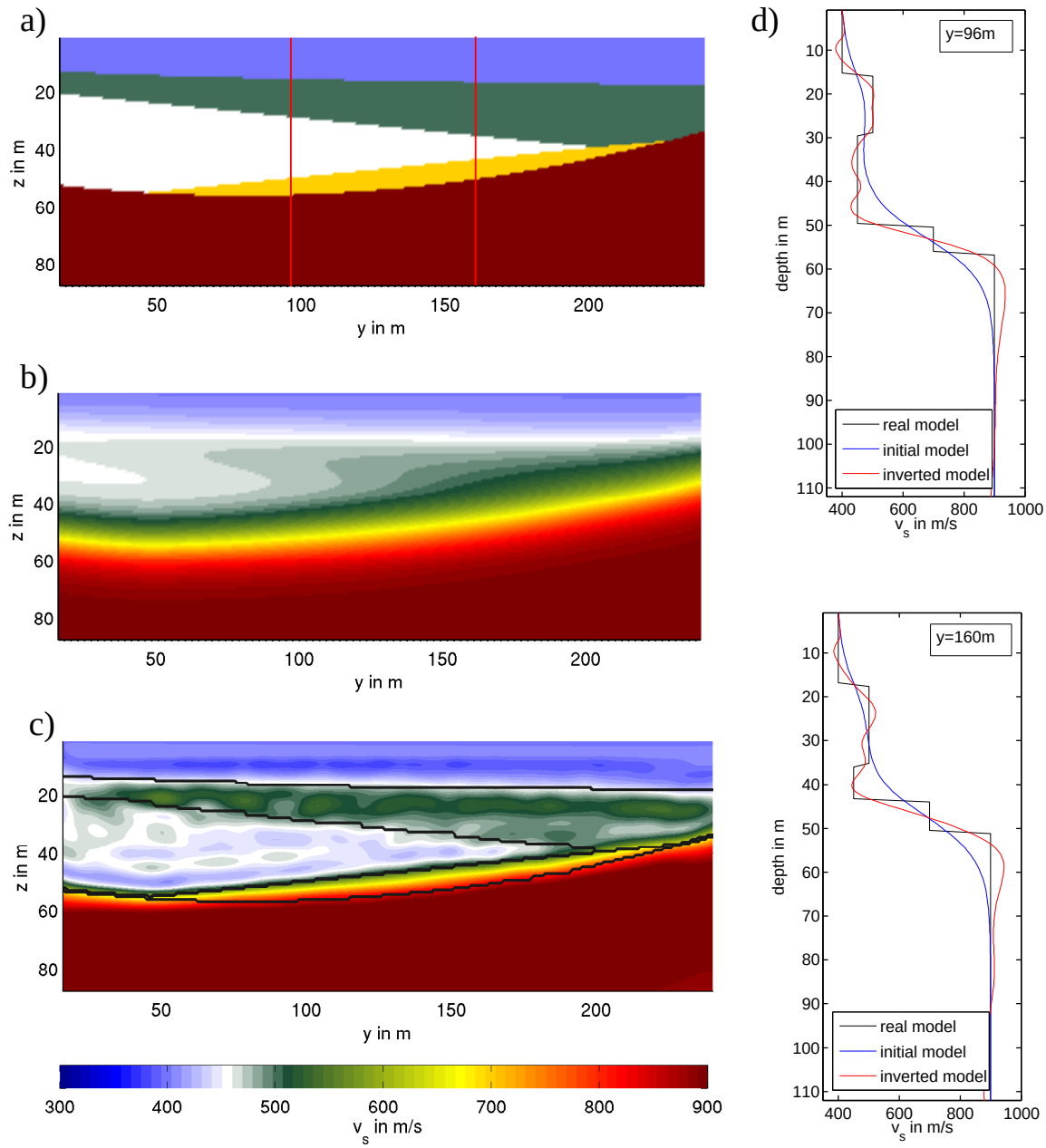

Figure 5.21: Vertical slice of the v_s surface model at $x = 128$ m: a) real model v_s, b) initial model v_s, c) inverted model v_s (170 iterations) with black lines indicating the layers of the real model, d) corommparison of real, initial and starting models along vertical profiles highlighted by red lines in a).

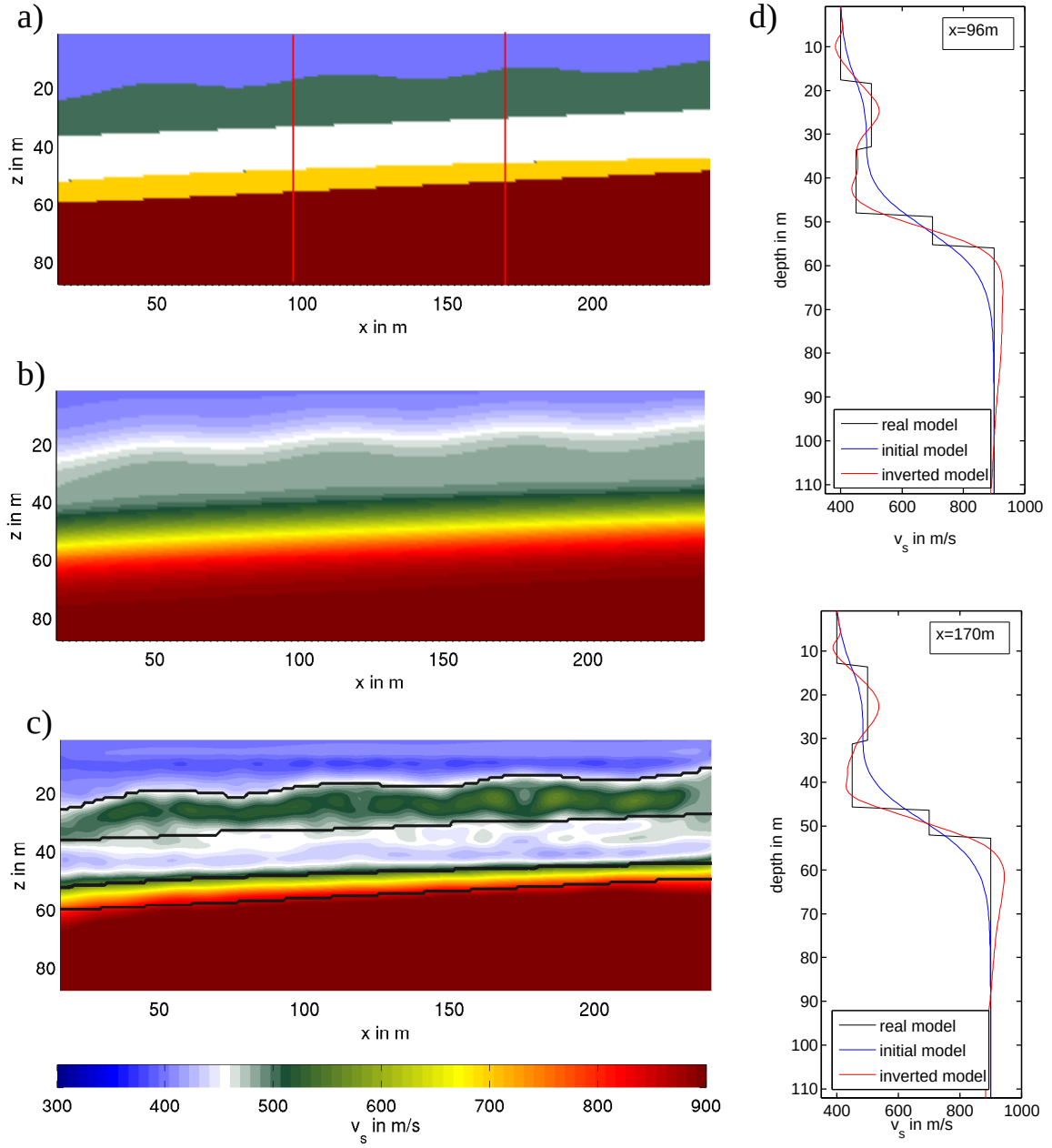

Figure 5.22: Vertical slice of the v_s surface model at $y = 128$ m: a) real model v_s, b) initial model v_s, c) inverted model v_s (170 iterations) with black lines indicating the layers of the real model, d) comparison of real, initial and starting models along vertical profiles highlighted by red lines in a).

the Hessian preconditioning allowed a reconstruction of the deeper part of the basin. Still, the sharp model boundaries could not be clearly imaged, but would require higher frequencies. Hereby, the shorter v_s wavelengths allow a much better resolution of the layer interfaces compared to the larger v_p wavelengths. The observed data is well explained by the final inverted models, even though convergence was relatively slow and a high number of iterations was necessary to achieve this data fit.

5.6 Conclusion

The inversion of complex models can require more sophisticated methods for preconditioning than local damping. In this chapter I showed the feasibility and application of Hessian preconditioning for 3D elastic FWI. Hereby, a diagonal Hessian approximation is applied to the gradient. The calculation of the Hessian matrix generally requires the calculation of the Green's receiver functions for each receiver and for forces directed in x-, y- and z direction, which can be very time consuming. Additionally, the storage of these wavefields (or the forward wavefield) is required, unless they are recalculated for each shot. I therefore use the following approximation to the full diagonal Hessian. On the one hand, only a subset of receivers is used. On the other hand, the simulations of the Green's receiver functions are only performed for δ-pulse forces directed in one spatial direction. Another aspect is, that the Hessian preconditioner is only calculated once for each frequency stage. These approximations allow the implementation of Hessian preconditioning at reasonable computational costs. For the storage costs, the implementation also benefits from the calculation of the diagonal Hessian in frequency domain, which requires the storage of wavefields only for discrete frequencies.
The Hessian preconditioning was validated in a simple transmission geometry experiment. By applying the Hessian preconditioning, geometric amplitude effects in the gradient were successfully removed, which mainly affected the area around sources and receivers. The inversion successfully reconstructed the different areas of a subdivided box. However, no higher convergence and improved inversion result was visible compared to the conjugate gradient inversion with local preconditioning. This is probably mainly due to the simplicity of this problem, which can already be well preconditioned by local damping.
A much more significant effect of Hessian preconditioning can be observed in surface geometry experiments. I inverted a model of sedimentary layers over a basin-like structure. The wavefields are dominated by surface waves, which indicates a high nonlinearity of the inverse problem. For a successful inversion it was necessary to start from very low frequencies of 2 Hz. Due to the acquisition geometry and the presence of the surface waves, significant gradient amplitudes are mainly limited to the very shallow subsurface. By applying Hessian preconditioning the amplitudes could be corrected to allow a model update in the middle and deeper part of the basin. This allowed a successful inversion of the 3D layered v_p and v_s model down to the basement of the basin. The different layers, including a low-velocity zone are clearly resolved. Even though convergence at higher frequency stages was relatively low, the final inverted data fit the observed data very well.

Chapter 6

Optimisation with the L-BFGS approach

6.1 Introduction

As the previous chapters showed, 3D elastic full-waveform inversion (FWI) is computationally still very costly. Nowadays, a Gauss-Newton or even full-Newton algorithm is generally too expensive for the inversion of realistically sized 3D problems. The conjugate gradient approach is advantageous due to its efficient calculation with the adjoint method, which features relatively low numbers of operations per iteration. Additionally, storage costs are reasonably low, especially when implemented for discrete frequencies in frequency domain. Still, the use of the Hessian operator can significantly improve the inversion and result in higher convergence. To optimise the conjugate-gradient approach it is possible to include information about the Hessian in the inversion. As a first approach, I discussed the application of an inverse diagonal Hessian approximation for preconditioning in the previous chapter. Geometric amplitude effects could be successfully corrected. Source and receiver artefacts were removed and an improved scaling of the gradient with depth was achieved. The additional use of the off-diagonal Hessian elements provides more benefits, as already addressed in section 5.2.1. The application of the inverse Hessian to the gradient acts as deconvolution of the gradient from limited-bandwidth effects and geometric amplitude effects (Pratt et al., 1998; Brossier et al., 2009).
Due to the impracticability to use the Newton methods for large-scaled problems, an additional class of optimisation methods was developed - the quasi-Newton methods. They infer information about the second derivative of the misfit function from changes in gradient and model for a certain range of iterations. Hence, no additional forward problems need to be solved for the calculation of second derivatives, even though additional storage capacity for gradients and models is required. The most popular quasi-Newton method for large-scaled problems is the L-BFGS (low memory BFGS) method (e.g. Nocedal, 1980). It is a variant of the BFGS method, which was named after its developers Broyden, Fletcher, Goldfarb and Shannon. Previous applications for 2D FWI by Brossier et al. (2009) and Brossier (2011) showed higher convergence rates and sharper images when using the L-BFGS approach compared to the conventional conjugate gradient method. Additionally, a correct inversion of multiple parameters without additional empirical weighting of the different parameter classes could be achieved. Another alternative approach is the truncated Newton method, which incorporates information about the Hessian by using second-order adjoint formula (Fichtner & Trampert, 2011; Métevier et al., 2012). These approximations of the Newton method can be an implemented efficiently in the FWI scheme.

In this chapter I will present the L-BFGS algorithm as an optimisation to the conjugate gradient approach in 3D elastic FWI. First, I will explain its theory and implementation. Second, the approach will be validated using the simple box example introduced in the previous chapter. Finally, I will present first results for the inversion of the surface geometry example.

6.2 Theory and Implementation

6.2.1 The BFGS-method

Instead of calculating the inverse Hessian $\mathbf{H}_k^{-1}$ directly, quasi-Newton methods approximate the (inverse) Hessian by using the change of the gradients over the iterations. Starting from an initial Hessian approximation $\mathbf{H}_0$ the approximation of the (inverse) Hessian is updated in each iteration to find a more accurate estimate. In the following, I will shortly introduce the concepts of the most common quasi-Newton method,- the BFGS method. Hereby I closely follow the theory given by Nocedal & Wright (1999).
The derivative of the Taylor expended misfit function E_k (equation 2.10) with respect to the model perturbation $\delta\mathbf{m}$ is given by

$$\nabla E_k(\mathbf{m}+\delta\mathbf{m}_k) = \nabla_m E_k(\mathbf{m}) + \mathbf{H}_k\delta\mathbf{m}_k. \tag{6.1}$$

The model update of the Newton method (see section 2.3) in iteration k is then calculated as

$$\delta\mathbf{m}_k = -\mathbf{H}_k^{-1}\nabla_m E_k. \tag{6.2}$$

If we consider the next iteration $(k+1)$ the following requirement can be defined:

$$\nabla E_k(\mathbf{m}) \stackrel{!}{=} \nabla E_{k+1}(\mathbf{m}-\delta\mathbf{m}_k). \tag{6.3}$$

Rewriting the right hand side with equation 6.1 gives

$$\begin{aligned} \nabla E_k(\mathbf{m}) &= \nabla E_{k+1}(\mathbf{m}) - \mathbf{H}_{k+1}\delta\mathbf{m}_k, \\ \nabla E_{k+1}(\mathbf{m}) - \nabla E_k(\mathbf{m}) &= \mathbf{H}_{k+1}(\mathbf{m}_{k+1}-\mathbf{m}_k), \\ \boldsymbol{\gamma}_k &= \mathbf{H}_{k+1}\mathbf{s}_k. \end{aligned} \tag{6.4}$$

This equation is known as the secant equation and poses a condition on $\mathbf{H}_{k+1}$ depending on the changes in the gradient $\boldsymbol{\gamma}_k = \nabla E_{k+1} - \nabla E_k$ and model changes $\mathbf{s}_k = \mathbf{m}_{k+1} - \mathbf{m}_k$. The secant equation can also be formulated for the inverse Hessian as

$$\mathbf{H}_{k+1}^{-1}\boldsymbol{\gamma}_k = \mathbf{s}_k. \tag{6.5}$$

$\mathbf{H}_{k+1}^{-1}$ is symmetric and positive definite. To fullfill equation 6.5 the curvature condition $\mathbf{s}_k^T\boldsymbol{\gamma}_k > 0$ has to be valid, which can be ensured by an adequate steplength.
The secant equation does not define $\mathbf{H}_{k+1}^{-1}$ uniquely. For a unique determination the BFGS approach searches for the positive definite, symmetric matrix $\mathbf{H}_{k+1}^{-1}$, which is closest to $\mathbf{H}_k^{-1}$

and fullfills the secant equation. Solving this minimisation problem for the weighted Frobenius norm (Nocedal & Wright, 1999) leads to the BFGS-formula

$$\mathbf{H}_{k+1}^{-1} = \mathbf{V}_k^T \mathbf{H}_k^{-1} \mathbf{V}_k + r_k \mathbf{s}_k \mathbf{s}_k^T \quad \text{with } r_k = \frac{1}{\boldsymbol{\gamma}_k^T \mathbf{s}_k} \quad \text{and } \mathbf{V}_k = \mathbf{I} - r_k \boldsymbol{\gamma}_k \mathbf{s}_k^T. \tag{6.6}$$

This equation updates $\mathbf{H}_k^{-1}$ using the changes in gradients and models of the recent iteration. An initial guess of $\mathbf{H}_0^{-1}$ is required. These approximations of the inverse Hessian matrix can then be applied to the gradients to find the model update (equation 6.2).
Of course, the estimation of $\mathbf{H}_{k+1}^{-1}$ only gives an approximation to the exact inverse Hessian, but still the BFGS method offers superlinear convergence and tends to be robust with good self-correcting properties (Nocedal & Wright, 1999).

6.2.2 The L-BFGS-approach

A main advantage of the BFGS approach is, that it avoids the explicit calculation of second derivatives. However, it is not suited for problems with a high number n of model parameters , due to the large number of BFGS-operations of $O(n^2)$ and the high storage costs for the huge, dense Hessian.
For these large problems, a variant of the BFGS approach was developed, known as the limited-memory BFGS (L-BFGS) method. Instead of calculating and storing the Hessian matrix of size $n \times n$ explicitly, the L-BFGS stores the change in gradients ($\boldsymbol{\gamma}$) and in models ($\mathbf{s}$) of the last m iterations. It thus employs curvature information of the recent m iterations to calculate an approximation of $\mathbf{H}_k^{-1}$. Applying equation 6.6 repeatedly for the previous m iterations results in the following formula (Nocedal & Wright, 1999):

$$\begin{aligned}\mathbf{H}_k^{-1} =& (\mathbf{V}_{k-1}^T \ldots \mathbf{V}_{k-m}^T) \mathbf{H}_{k0}^{-1} (\mathbf{V}_{k-m} \ldots \mathbf{V}_{k-1}) \\ &+ r_{k-m} (\mathbf{V}_{k-1}^T \ldots \mathbf{V}_{k-m+1}^T) \mathbf{s}_{k-m} \mathbf{s}_{k-m}^T (\mathbf{V}_{k-m+1} \ldots \mathbf{V}_{k-1}) \\ &+ r_{k-m+1} (\mathbf{V}_{k-1}^T \ldots \mathbf{V}_{k-m+2}^T) \mathbf{s}_{k-m+1} \mathbf{s}_{k-m+1}^T (\mathbf{V}_{k-m+2} \ldots \mathbf{V}_{k-1}) \\ &+ \ldots \\ &+ r_{k-1} \mathbf{s}_{k-1} \mathbf{s}_{k-1}^T.\end{aligned} \tag{6.7}$$

Hence, instead of updating $\mathbf{H}_k^{-1}$ in each iteration, it is newly calculated with equation 6.7 with information of the m previous iterations. -H_{k0}^{-1} is an initial inverse Hessian approximation, which is allowed to vary from iteration to iteration.
The L-BFGS approach can be implemented very efficiently. The Hessian matrix is hereby not calculated explicitly, but instead the product $\mathbf{H}_k^{-1} \nabla E_k$ is estimated with the following recursive algorithm given in Nocedal & Wright (1999):

$\mathbf{q} \leftarrow \nabla E_k$
for $(i = k-1, \ldots, k-m)$
$\quad \alpha_i \leftarrow r_i \mathbf{s}_i^T \mathbf{q}$
$\quad \mathbf{q} \leftarrow \mathbf{q} - \alpha_i \boldsymbol{\gamma}_i$
end for

$\mathbf{z} \leftarrow \mathbf{H}_{k0}^{-1}\mathbf{q}$
for $(i = k-m, \ldots k-1)$
$\quad \beta \leftarrow r_i \boldsymbol{\gamma}_i^T \mathbf{z}$
$\quad \mathbf{z} \leftarrow \mathbf{z} + \mathbf{s}_i(\alpha_i - \beta)$
end for
$\mathbf{H}_k^{-1}\nabla E_k = \mathbf{z}$

This algorithm is performed once in each iteration and consists of $(4mn+n)$ multiplications if a diagonal $\mathbf{H}_{k0}^{-1}$ is chosen. The L-BFGS algorithm requires additional storage of $(2mn+m)$ floating numbers for $\boldsymbol{\gamma}, \mathbf{s}$ and r. For reasonable numbers of m the L-BFGS method can therefore be performed at relatively low extra costs of runtime an storage. In general, values between 5 and 20 are used for m. In the first $(m-1)$ iterations the result corresponds to the BFGS method. Even though Nocedal & Wright (1999) mentioned, that convergence of the L-BFGS can be low compared to the BFGS approach, Brossier et al. (2009) and Brossier (2011) showed great improvements in convergence and resolution compared to the conjugate gradient method for FWI. This, together with the low computational extra costs make it very attractive for 3D FWI.

6.2.3 Application of L-BFGS for multi-parameter FWI

Our aim in 3D FWI is the successful, optimised, multi-parameter inversion of complex 3D structures. In this section, I present the L-BFGS approach not as the only preconditioner for FWI but as an additional applied technique to improve the inversion. Hereby I follow the approach discussed by Brossier (2011) and apply the L-BFGS approach using the preconditioned gradients of normalised parameters.

L-BFGS for preconditioned gradients

The general L-BFGS approach uses the changes in gradients and model parameters as discussed in section 6.2.2. For FWI, however, Brossier (2011) successfully applies the L-BFGS algorithm for multi-parameter FWI by using changes in preconditioned gradients. In a first step the diagonal Hessian approximation is applied to the gradients for preconditioning, in a second step, the model update is calculated using the L-BFGS approach. The preconditioning with the diagonal Hessian approximation was discussed in the previous chapter as a first optimisation to the conjugate gradient approach. The L-BFGS method can then be seen as a second optimisation.
Differently to the application of the diagonal Hessian for preconditioning, the approximation of the Hessian by L-BFGS also considers the off-diagonal elements. Additionally, the L-BFGS algorithm is performed in each iteration, and thus offers a Hessian approximation for the current iteration, whereas the diagonal Hessian preconditioner was approximated as constant for the whole frequency stage.
The general L-BFGS method requires a good guess for the initial Hessian, such as a diagonal Hessian approximation, to offer a good convergence. When applying the L-BFGS algorithm to gradients preconditioned by the diagonal inverse Hessian a simple guess for the initial Hessian is sufficient. I employ a scaled identity matrix as initial Hessian approximation as suggested by

Nocedal & Wright (1999)

$$\mathbf{H}_{k0}^{-1} = \frac{\mathbf{s}_{k-1}^T \boldsymbol{\gamma}_{k-1}}{\boldsymbol{\gamma}_{k-1} \boldsymbol{\gamma}_{k-1}^T} \mathbf{I}, \tag{6.8}$$

which attempts to estimate the correct size of the Hessian operator, so that a steplength close to one is valid. Still, a steplength calculation is useful and is included similar to the conventional conjugate gradient implementation. The difference is, that test steplengths go up to one and optimal steplengths close to one should be reached during inversion.

Parameter normalisation

In the conjugate gradient approach I normalise the conjugate gradients to their maximum values for each parameter class and scale them with an average model parameter. No information about the relative model update of the parameter classes compared to each other is provided by the inversion (section 3.5.1). The L-BFGS approach, by contrast, approximates the Hessian matrix, which can account for the scaling of the model update for the different parameter classes and offers an absolute model update. The L-BFGS approach should therefore lead to an improved multi-parameter FWI compared to the conjugate gradient method. Still, the multiparameter FWI inverts for parameters of different amplitudes and units. For this reason, Brossier (2011) suggested a normalisation of each parameter class to enable a dimensionless L-BFGS scheme. The normalised parameters are calculated as

$$\hat{v}_p(\mathbf{x}) = \frac{v_p(\mathbf{x})}{v_{p0}}, \quad \hat{v}_s(\mathbf{x}) = \frac{v_s(\mathbf{x})}{v_{s0}}, \quad \hat{\rho}(\mathbf{x}) = \frac{\rho(\mathbf{x})}{\rho_0}. \tag{6.9}$$

The representative parameters v_{p0}, v_{s0} and ρ_0 can be chosen as the average value of the initial models. This leads to the following normalisation of the gradients

$$\frac{\partial E}{\partial \hat{v}_p(\mathbf{x})} = \frac{\partial E}{\partial v_p(\mathbf{x})} v_{p0}, \qquad \frac{\partial E}{\partial \hat{v}_s(\mathbf{x})} = \frac{\partial E}{\partial v_s(\mathbf{x})} v_{s0}, \qquad \frac{\partial E}{\partial \hat{\rho}(\mathbf{x})} = \frac{\partial E}{\partial \rho(\mathbf{x})} \rho_0. \tag{6.10}$$

and the following normalisation for the diagonal Hessian approximation:

$$H_a^{\hat{v}_p} = H_a^{v_p} v_{p0}^2, \qquad H_a^{\hat{v}_p} = H_a^{v_p} v_{p0}^2, \qquad H_a^{\hat{\rho}} = H_a^{\rho} \rho_0^2. \tag{6.11}$$

This normalisation of gradients and Hessian matrices results in a normalised model update $\delta\hat{m}$ which needs to be denormalised before updating the model

$$\delta v_p = \delta \hat{v}_p v_{p0}, \qquad \delta v_s = \delta \hat{v}_s v_{s0}, \qquad \delta \rho = \delta \hat{\rho} \rho_0. \tag{6.12}$$

The normalisation finally allows a dimensionless L-BFGS algorithm joint for all parameter classes. The vector pairs saved in the iteration k, i.e., $\boldsymbol{\gamma}_k = ((\gamma_1)_k, ..., (\gamma_n)_k)$ and $\mathbf{s}_k = ((s_1)_k, ..., (s_n)_k)$ run over all model parameters, that is, all parameter classes and grid points. As a result, the L-BFGS gives an absolute model update for each normalised parameter without the need of heuristic weighting of each parameter class.

The workflow

The combination of the Hessian preconditioning and the L-BFGS scheme for normalised parameters is implemented as follows:

---

for each frequency stage

- calculate diagonal Hessian approximation ($H_a^{\hat{v}_p}, H_a^{\hat{v}_s}, H_a^{\hat{\rho}}$) for normalised parameters

in each iteration

- calculate gradients of normalised parameters
- apply diagonal Hessian approximation to gradient
- save change in preconditioned gradients and models, discard iterations smaller k-m
- L-BFGS algorithm to find normalised model update
- denormalise model update
- step length calculation: try steplength $\alpha = 1$ first
- model update

In the following I will show first applications of this approach for 3D elastic FWI.

6.3 Validation with the subdivided box model

6.3.1 Inversion setup

For a first validation of the technique I inverted the subdivided box model dataset with the L-BFGS approach. Its inversion with Hessian preconditioning was presented in section 5.4 and compared to the results of the conventional conjugate gradient method with local preconditioning. For the L-BFGS inversion I invert the same transmission geometry data set, using the same initial models. Inversion parameters, such as frequency stages and number of iterations, are similar. Furthermore, the diagonal Hessian approximations calculated for each frequency stage in the inversion with Hessian preconditioning (section 5.4) are reused for preconditioning in this example.
Differently to the previous inversions, however, the model update is now estimated by applying the L-BFGS algorithm, as explained in the previous section. Hereby the changes in gradients and model parameters over the previous five iterations are employed. The gradients are not normalised by their maximum values and scaled with an average model parameter. By contrast, this method should provide a maximum model update for the different model parameter classes with respect to each other. An optimal steplength is estimated by using test steplengths of 0.4 and 0.8. A steplength close to one is expected with this approach. The maximum steplength is limited to one.

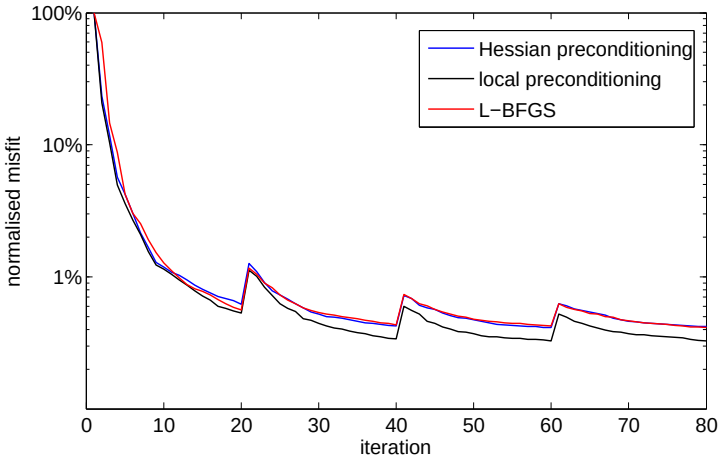

Figure 6.1: Misfit process for box model normalised to the initial misfit. Comparison of L-BFGS approach, Hessian preconditioning and local tapering.

6.3.2 Inversion process and comparison to previous results

The results of the L-BFGS approach presented in this section are compared to the conjugate gradient method with Hessian preconditioning and local preconditioning discussed in section 5.4. Figure 6.1 shows the misfit process for the three different approaches. Overall, they show a very similar behaviour. The local preconditioning shows a slightly better convergence to a value of 0.3% of the initial misfit, whereas the Hessian preconditioning and the L-BFGS approach look very similar and reach a value of 0.4% of the initial misfit. The same behaviour of convergence is very interesting regarding the fact, that the estimation of the absolute model update, i.e., normalisation and steplength estimation, is very different for the L-BFGS method.

Therefore, the L-BFGS approach could successfully minimise the misfit. In most iterations, a steplength of one was used for the update. Another inversion test without limitation of the maximum steplength to one showed, that the inversion processed very similar and optimal steplengths close to one were estimated.

The inverted models are not plotted here, because they look very similar to the inversion results shown in section 5.4.3. For a quantitative comparison I calculated the RMS deviations of the initial and starting velocity models with respect to the true velocity models (Table 6.1) for the three different approaches. The starting models have average parameter deviations of about 1% from the true model. They are only caused by variations within the box. The model error decreased to about 0.53% for v_p and 0.35% for v_s. Variations between the different methods are very small and no significant improvement can be observed for any of the three approaches.

	RMS(v_p, start)	RMS(v_s, start)	RMS(v_p, inv.)	RMS(v_s, inv.)
local tapering	1.09%	0.90%	0.53%	0.34%
Hessian preconditioning	1.09%	0.90%	0.54%	0.36%
L-BFGS	1.09%	0.90%	0.54%	0.35%

Table 6.1: RMS model deviation of initial and inverted models with respect to the true models. Comparison of local tapering, Hessian preconditioning and L-BFGS.

Overall, the L-BFGS approach was successfully applied for this simple example. Its results for the inversion of v_p and v_s are comparable to the conjugate gradient approach with local tapering

and with Hessian preconditioning. In spite of this, this optimisation did neither improve the results nor result in a better convergence. This is probably due to the limited complexity of the transmission geometry used in this inverse problem. In this case, Hessian preconditioning and L-BFGS optimisations are dispensable, which is likely to change for increasingly complex inversion problems.

6.4 3D layered surface model - preliminary results

Of course, the L-BFGS optimisation is mainly intended for its application to complex problems. This section will offer first results of the L-BFGS approach applied to the 3D layered surface model. This complex problem was introduced in the previous chapter in section 5.5 and inverted with the gradient approach applying Hessian preconditioning. In this section I will optimise this former inversion scheme and test the application of the L-BFGS approach to the preconditioned gradients for this problem.

6.4.1 Model and inversion setup

The same surface geometry data set as previously was inverted and the corresponding real models, the acquisition geometry as well as the starting models were already given in section 5.5.1. I used the same inversion setup as described in section 5.5.2 with frequencies and frequency stages given in Table 5.2, only the number of iterations per stage varies. Furthermore, the same preconditioning was applied using the diagonal Hessian approximations presented in section 5.5.3.
As already explained in this chapter, the L-BFGS algorithm estimates the absolute model update and no normalisation of the gradients to their maximum is required. Five previous iterations were taken into account in the L-BFGS algorithm. The test steplengths for the steplength calculation were 0.5 and 1.0 and a maximum steplength of one was set.

6.4.2 Inversion process

In total 78 iterations were performed over the six frequency stages. The process of the inversion can be visualised when looking at the misfit process plotted in Figure 6.2a). Overall, the misfit curve converges very nicely. In each frequency stage, about 15 iterations were performed, which corresponds to 24 hours of runtime on 800 cores. This is the maximum time per run on the supercomputer JUROPA (Jülich). Two of the frequency stages were problematic, and did not result in a significant misfit decrease. For frequencies from 6.5-9.0 Hz, the misfit decreased only in the first three iterations of this stage and started increasing afterwards. Because of the already good data fit in this stage, I decided to continue with the next frequency stage. The other frequency stage, which failed to converge sufficiently is the last stage from 17.0-22.0 Hz. Here only a small misfit decrease could be reached and the inversion was terminated after this stage.
A good convergence of the misfit generally also corresponds to a successful steplength estimation. The steplength curve is plotted in Figure 6.2b). Overall, it is visible, that a steplength close to one was reached very frequently. Especially in the first two frequency stages there are only few exceptions. However, in frequency stage three and six, very small steplengths were

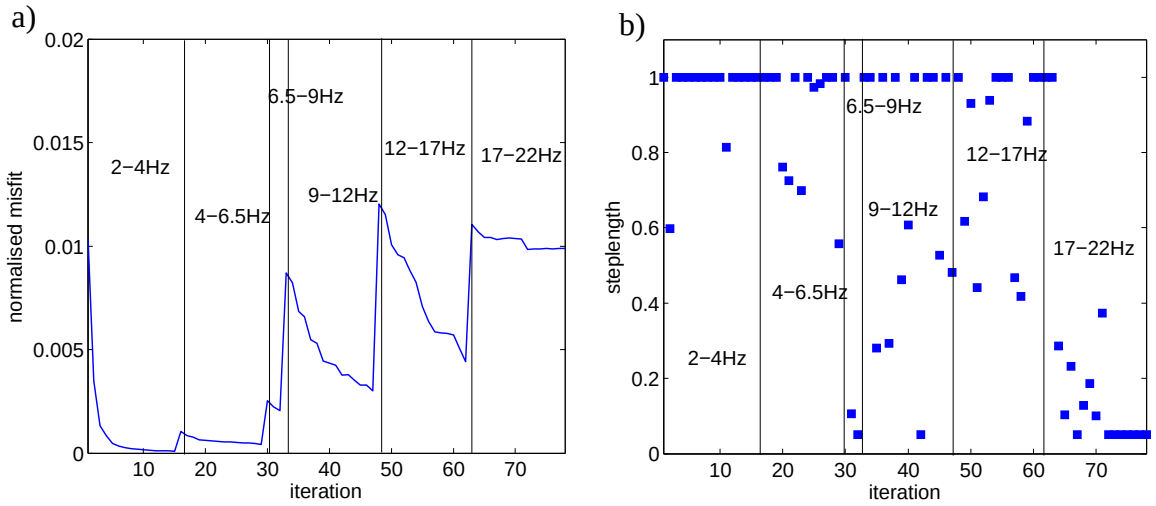

Figure 6.2: a) Misfit process of the surface model inversion for the different frequency stages. Misfit values are normalised to the misfit between observed and starting data of the full frequency range up to 22 Hz and b) steplengths employed for the different stages.

estimated, or the steplength calculation failed and the algorithm was forced to apply minimum steplengths of 0.05. Here again, the problems within these stages become visible. Future work will be necessary to solve the problems to enable a stable inversion within these frequency stages.
In spite of the difficulties in two of the stages, the L-BFGS algorithm succeeded to improve convergence significantly. The misfit process for the inversion applying only Hessian preconditioning was presented in Figure 5.16. A comparison of the misfit curves shows, that the L-BFGS approach

- requires less than half the iterations compared to the inversion without L-BFGS,
- provides a better data fit in all frequency stages except for the last and
- exhibits a much smoother misfit curve and thus stabilises the inversion.

Thus, overall the L-BFGS optimisation results in a much better convergence. Due to the difficulties of the L-BFGS inversion in the last frequency stage, the final misfit values of both inversions are very similar.

6.4.3 Results

The inverted models for the inversion with L-BFGS are shown on two perpendicular vertical slices in Figure 6.3 and 6.4 for v_p and v_s, respectively. The results without L-BFGS were plotted in the Figures 5.19 to 5.22. The latter also show the starting and real models.
In the v_p model the different sedimentary layers (indicated as black lines) of the true model are successfully recovered, except for the thin high-velocity layer at the basin base. The resolution of the interfaces is similar to the result without L-BFGS shown in the Figures 5.19 and 5.20. Still, the L-BFGS model looks very rough, with small heterogeneities introduced within the layers. It is possible, that these artefacts cause some problems in the last frequency stage. For

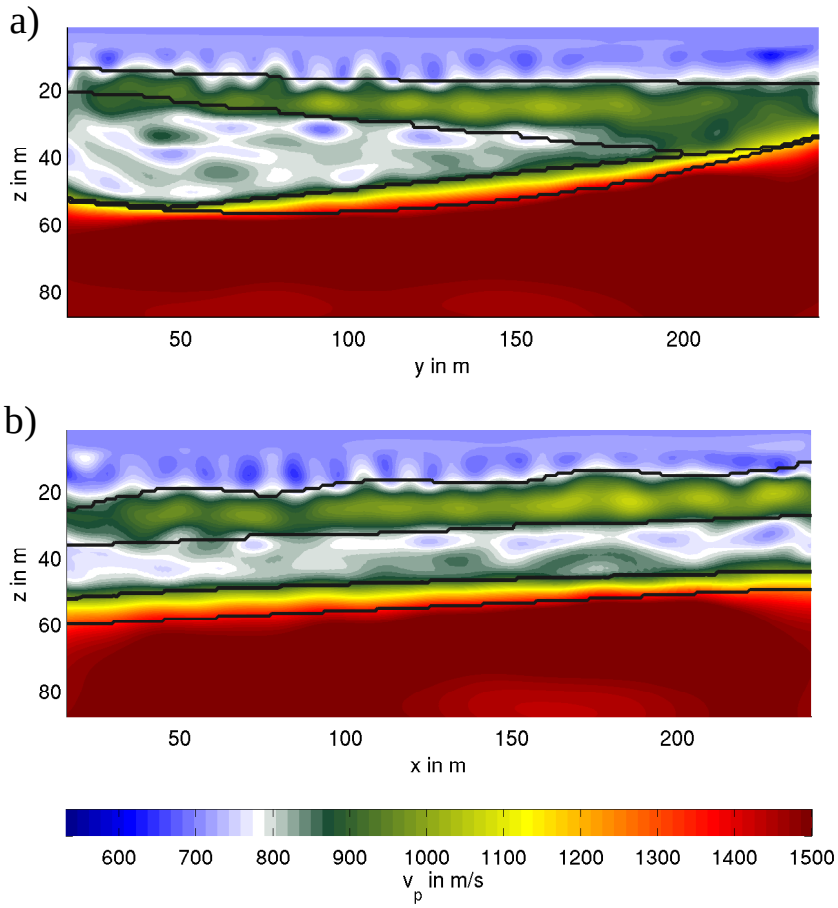

Figure 6.3: Inversion result of v_p for surface model inversion with L-BFGS after 78 iterations. Black lines indicate interfaces of true model: a) slice at $x = 128$ m and b) slice at $y = 128$ m.

the future, a smoothing of v_p gradients or models could be helpful.
The v_s result presented in Figure 6.4 shows a very good reconstruction of the different sedimentary layers. Interfaces between the layers are well resolved. The result is comparable to the result without L-BFGS (Figures 5.21 and 5.22).

6.4.4 Discussion

Overall, the first results of the layered surface model FWI with L-BFGS are very promising. Convergence could be improved significantly compared to the inversion without L-BFGS. Consequently, less than half the iterations were sufficient and lower misfit values could be reached. The final velocity models are comparable to the results without L-BFGS. Thus, no significantly improved and sharpened images could be gained during this first inversion.
Some difficulties occured during the inversion of two frequency stages, including the last. Problems in the last frequency stage were already found in the inversion without L-BFGS. Apparently, with increasing nonlinearity at higher frequencies, the inversion with and without L-BFGS exhibits problems. Further work will be necessary to solve these problems and to enable an overall stable inversion.

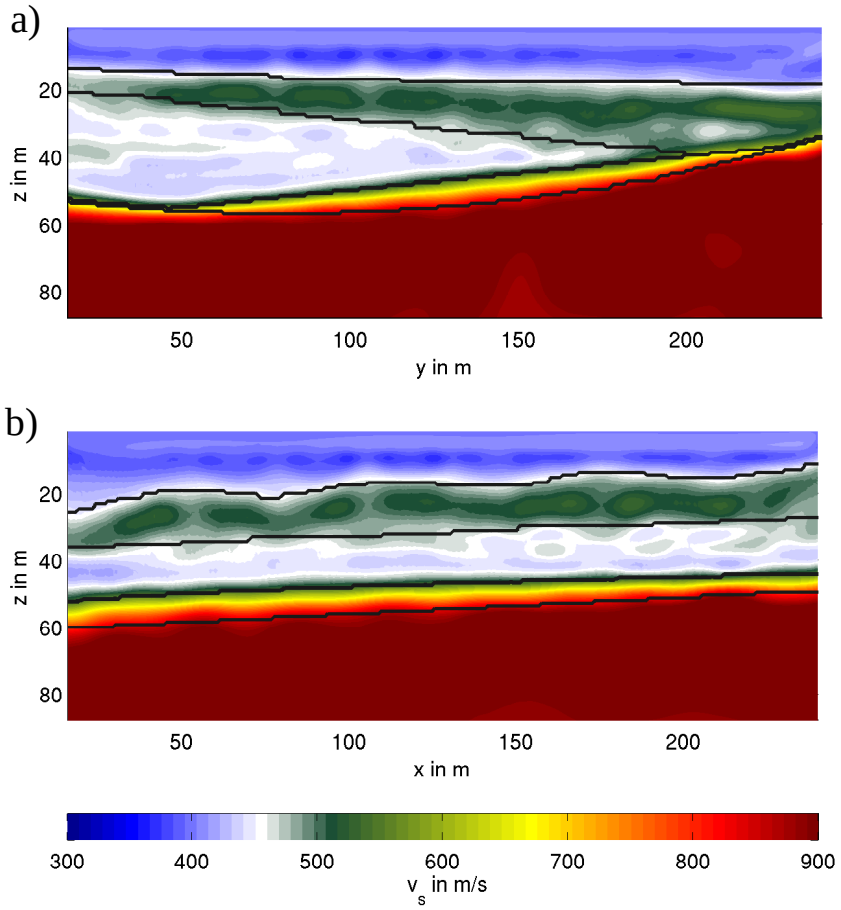

Figure 6.4: Inversion result of v_s for surface model inversion with L-BFGS after 78 iterations. Black lines indicate interfaces of true model: a) slice at $x = 128$ m and b) slice at $y = 128$ m.

6.5 Summary and outlook

In this chapter, the L-BFGS approach was presented as a second optimisation beside Hessian preconditioning for 3D FWI. By using changes in gradients and models over the few former iterations, the L-BFGS algorithm approximates the effects of the Hessian on the gradients. Absolute model updates do not require further scaling, and in the optimal case, a steplength of one can be used. For its application to multi-parameter FWI a parameter normalisation is employed, which results in a dimensionless L-BFGS scheme and enables the simultaneous inversion of parameter classes of different scales and units. The workflow presented here applies the L-BFGS approach to the gradients preconditioned with the approximate Hessian. Even though some additional storage for gradients and models needs to be provided, only few additional operations are necessary so that this optimisation can be implemented at cheap costs.
The scheme could be successfully validated for a simple box example in transmission geometry, even though no improvements regarding convergence and model reconstruction compared to the inversion without L-BFGS were observed in this simple example. As the inversion of the sedimentary layers in surface geometry showed, this is different for increasingly complex problems. First results of this example indicate, that the L-BFGS approach can considerably improve the convergence and enables a better data fit. Thus, even though some additional operations are required to calculate the L-BFGS update, the reduced number of iterations results in a significant reduction of runtime. Hereby the inverted models with and without L-BFGS were comparable.
Even though the overall performance was very promising some problems were encountered during this first application to the surface model, which require future testing. In two frequency stages of the inversion the inversion could not successfully decrease the misfit and failed to find an optimum steplength. Here it might be necessary to exclude some gradients from the L-BFGS scheme and only allow “successful’ gradients to contribute. Additionally, the number of iterations in this example was defined by the maximum runtime on JUROPA, Jülich. It is possible, that with an increasing number of iterations, the L-BFGS approach would result in improved images compared to the results without L-BFGS. Future testing could also include the variation of the number of iterations used for L-BFGS and different types of preconditioning applied to the gradients before L-BFGS.

Chapter 7

Summary and conclusion

Full-waveform inversion (FWI) aims to deduce highly resolved, multi-parameter images of the subsurface by minimising the misfit between observed and synthetic waveforms. Depending on the size of the problem and the number of model parameters, it can be computationally quite expensive. Only in recent years, with ongoing progress in computer technologies, investigations in seismic tomography concentrate increasingly on FWI. Nevertheless, most applications today are still restricted to the 2D or 3D acoustic or the 2D elastic wave equation. However, the inversion of data sets from complex 3D subsurfaces, which contain significant impact of shear waves, can require the consideration of 3D elastic wave propagation.
In this thesis I presented the theory and implementation of 3D elastic FWI and its application for the 3D reconstruction of elastic subsurface parameters. A key aspect of this work was the implementation of a computationally efficient FWI which enables the solution of the large-sized 3D elastic inverse problem. This was realised by employing the conjugate gradient approach calculated with the adjoint method and with the use of a time-frequency approach. FWI results showed, that this implementation is well capable to resolve differently sized 3D structures in crystalline rock environment in transmission geometry.
To optimise the performance of the gradient approach for the FWI of complex data sets, two different optimisations based on approximations of the Hessian were tested. In a first step I applied a diagonal Hessian approximation for preconditioning. In a second step, this approach was extended with the L-BFGS algorithm. The improved preconditioning enabled a successful inversion of a complex 3D layered surface model. Additionally, the L-BFGS method could considerably improve the convergence.
In the following, I describe the results gained in the different parts of this thesis in more detail.

Theory and implementation

The 3D FWI code 3D-AWAIT iteratively minimises the L_2-norm based misfit of displacements in conjugate gradient direction. Gradients are calculated with the adjoint method, which is derived for the self-adjoint second-order wave equation. For a computationally efficient 3D FWI, I employ a time-frequency approach, which simulates wavefields in time domain but performs the gradient calculation in frequency domain. Thus, gradients are calculated as a multiplication of forward and adjoint wavefield for discrete frequencies. The inversion in frequency domain results in an enormous storage reduction compared to time domain. Differently to the storage of the full forward wavefield in time domain, wavefields are saved in memory only for few fre-

quencies. A small drawback is the resulting loss in redundancy compared to time domain.
Considering the runtime, a very efficient modelling of 3D elastic wavefields can be achieved with the finite-difference code SOFI3D in time domain. It is based on a velocity-stress formulation. Its parallelisation, which is based on domain decomposition, can be maintained during the full inversion process, leading to efficient computations on large computer clusters. During the simulation of forward- and back-propagated wavefields discrete Fourier transforms are performed on the fly to extract particle velocities at monochromatic frequencies. They are stored in memory and used for subsequent gradient calculations. I employ the compressional and shear wave velocities as well as density as parametrisation.
To correct the gradient from high amplitude artefacts in the vicinity of sources and receivers a preconditioning is applied, which corresponds to a local circular taper around source and receiver positions in its simplest form. The search direction is then calculated in conjugate gradient direction. To derive the total model update, a steplength estimation is performed by means of a parabolic line search.
Important aspects to mitigate the high nonlinearity of FWI are the choice of starting models and the multi-scale inversion. The implementation of the multi-scale inversion is straightforward in frequency domain. Groups of discrete frequencies are inverted simultaneously with frequencies increasing from frequency stage to frequency stage. The use of groups of overlapping frequencies is favourable. Hereby the choice of starting frequency and starting model is strongly correlated and needs to ensure, that no cycle skipping occurs.

3D FWI of small-scale heterogeneities in transmission geometry

As a first application, the synthetic inversion of small-scale 3D structures of crystalline rocks, represented by a random medium, was presented for different transmission geometries. Additionally, results of a checkerboard resolution test were shown for the different acquisition geometries. In a favourable 3D acquisition geometry, where sources and receivers are located in opposite planes, the inversion successfully recovered the cubes of the checkerboard and the differently sized 3D structures of the random medium starting from homogeneous initial models. Still, most transmission geometry measurements use a 2D cross-well geometry of two boreholes. My study showed, that even with this simple geometry the 3D FWI is capable to resolve 3D structures up to 1-2 wavelengths adjacent to the plane between the boreholes. The resolution is lower compared to the 3D geometry inversion. Additionally, a 2D FWI of this data set was performed. It offers a similar resolution in the source-receiver plane but suffers from artefacts due to the 3D scattered waves. Thus, even in case of simple 2D geometries it can be useful to apply a 3D FWI instead of a 2D FWI to detect 3D structures adjacent to source-receiver plane.

Methodological optimisation of the gradient approach

By considering the Hessian operator, Newton methods offer an improved convergence and can result in better models compared to gradient methods. However, the calculation of the Hessian matrix is computationally too expensive for 3D FWI of realistic problems. Instead, I tested two different approximations of the Hessian in order to improve the performance of the gradient method: the use of a diagonal Hessian approximation for preconditioning and the L-BFGS algorithm.

The inverse of a diagonal Hessian approximation can be applied for preconditioning of the gradients as a physically funded approach to account for the geometric amplitude effects in the gradients. To allow its calculation at reasonable costs, this preconditioning operator is only estimated once per frequency stage. The elements of the Hessian are computed from the partial derivative wavefields in frequency domain. To estimate these, the Green's receiver functions are multiplied with the forward wavefields for each source-receiver combination and each frequency. The corresponding forward simulations from each receiver and the storage of these wavefields for discrete frequencies cause most of the additional computational costs required for this approach. In order to mitigate these costs, I only employ a subset of receivers and compute only one component of the partial derivative wavefields.

The L-BFGS method approximates the Hessian in a different way. Information about the second derivative of the misfit function is inferred from changes in gradients and models over the few previous iterations. Hereby the L-BFGS algorithm estimates the effects of the inverse Hessian on the gradient without explicitly forming the huge Hessian matrix. In my workflow, the L-BFGS algorithm is applied as a second optimisation additionally to the Hessian preconditioning. Thus, the preconditioned gradients are used for its computations. In this case a scaled identity matrix is sufficient as an initial guess of the Hessian for the L-BFGS scheme. To allow the inversion of different parameter classes, a parameter normalisation is implemented, which allows a dimensionless L-BFGS scheme of all parameter classes simultaneously. The L-BFGS method has the main advantage, that it provides absolute model updates of the parameter classes with respect to each other. In the optimal case, a steplength close to one is valid.

For validation a dataset of a simple box in transmission geometry was inverted and the three different approaches were compared: the conventional gradient method with local preconditioning, the gradient method with Hessian preconditioning and the extension with the L-BFGS approach. The results showed a very similar behaviour in convergence and model reconstruction. Thus, both optimisations allowed a successful inversion, but they did not improve convergence and model quality. The main reason for this is, that the effects of the Hessian for this transmission geometry example were mainly concentrated on the area around sources and receivers. However, artefacts in this area can also be successfully removed by local tapering.

The optimisations finally allowed the inversion of a surface geometry example. The model comprises four sedimentary layers, including a low-velocity layer over a basin-shaped homogeneous halfspace. The corresponding wavefields are dominated by high amplitude surface waves, which leads to a high complexity of the inverse problem. A sufficient illumination of the subsurface required 49 sources. The high number of sources is mainly responsible for the high computational costs of this example. I inverted for seismic velocities and used strongly smoothed versions of the real models as initial models.
For this example, the effects of Hessian preconditioning were much more profound. Due to the predominant surface wave and the surface acquisition geometry the unpreconditioned gradients mainly focus in shallow areas. The Hessian preconditioning allowed a model update in the deeper layers of the basin by removing geometric amplitude effects in the gradient. This finally resulted in a successful reconstruction of the different layers in both v_p and v_s. Hereby interfaces are resolved as smoothed structures. A higher resolution in v_s compared to v_p was

reached, which is likely to result from the smaller v_s-wavelengths.
Preliminary results of the inversion with L-BFGS optimisation were presented. They show a very promising performance, even though convergence problems in two of the six frequency stages were encountered. Still, the use of the L-BFGS algorithm caused a considerably improved convergence. Consequently, less than half of the iterations were required to gain a data fit and inverted model comparable to the result without L-BFGS.

To conclude, my thesis presents an efficient implementation of 3D elastic full-waveform inversion based on the conjugate gradient method. The synthetic applications prove the abilities of the inversion to reconstruct multi-parameter 3D structures of the subsurface. Very promising results for the inversion of complex models were achieved when applying the approximate diagonal Hessian for preconditioning combined with the L-BFGS approach. The combination of the time-frequency approach and these optimisations results in reasonable computational costs and is therefore very interesting for the 3D FWI of multi-parameter problems at larger scales.

Bibliography

Aki, K. & Richards, P., 1980. *Quantitative seismology theory and methods*, vol. 1, W.H. Freeman and Company.

Berenger, J., 1994. A perfectly matched layer for the absorption of electromagnetic waves, *Journal of Computational Physics*, **114**, 185–200.

Bohlen, T., 2002. Parallel 3-D viscoelastic finite difference seismic modeling, *Computers and Geoscience*, **28**, 887–889.

Brossier, R., 2011. Two-dimensional frequency-domain visco-elastic full waveform inversion: parallel algorithms, optimization and performance, *Computers and Geosciences*, **37**, 444–455.

Brossier, R., Operto, S., & Virieux, J., 2009. Seismic imaging of complex onshore structures by 2D elastic frequency-domain full-waveform inversion, *Geophysics*, **74**(6), WCC105–WCC118.

Bunks, C., Saleck, F., Zaleski, S., & Chavent, G., 1995. Multiscale seismic waveform inversion, *Geophysics*, **60**(5), 1457–1473.

Butzer, S., Kurzmann, A., & Bohlen, T., 2013. 3D elastic full-waveform inversion of small-scale heterogeneities in transmission geometry, *Geophysical Prospecting*, **61**(6), 1238–1251.

Byrd, R., Lu, P., Nocedal, J., & Zhu, C., 1995. A limited memory algorithm for bound constrained optimization, *SIAM Journal on Scientific and Statistical Computing*, **16**(5), 1190–1208.

Castellanos, C., Etienne, V., Hu, G., Operto, S., Brossier, R., & Virieux, J., 2011. Algorithmic and methodological developments towards full waveform inversion in 3D elastic media, in *Extended Abstract, SEG Annual Meeting*, San Antonio, 2793-2797.

Cerjan, C., Kosloff, D., Kosloff, R., & Reshef, M., 1985. A nonreflecting boundary condition for discrete acoustic and elastic wave equations, *Geophysics*, **50**(4), 705–708.

Collino, F. & Tsogka, C., 2001. Application of the perfectly matched absorbing layer model to the linear elastodynamic problem in anisotropic heterogeneous media, *Geophysics*, **66**(1), 294–307.

Courant, R., Friedrichs, K., & Lewy, H., 1967. On the partial differential equations of mathematical physics, *IBM Journal*, **11**(2), 215–234.

Epanomeritakis, I., Akcelik, V., Ghattas, O., & Bielak, J., 2008. A Newton-CG method for large-scale three-dimensional elastic full-waveform seimic inversion, *Inverse Problems*, **24**, 26pp.

Etienne, V., Hu, G., Operto, S., Virieux, J., Barkved, O., & J.H., K., 2012. Three-dimensional acoustic full waveform inversion - algorithm and application to Valhall, in *Extended Abstracts, 74th EAGE meeting*, Copenhagen, Denmark, P343.

Fichtner, A., 2011. *Full seismic waveform modelling and inversion*, Advances in Geophysical and Environmental mechanics and mathematics, Springer.

Fichtner, A. & Trampert, J., 2011. Hessian kernels of seismic data functionals based upon adjoint techniques, *Geophysical Journal International*, **185**, 775–798.

Fichtner, A., Kennett, B., Igel, H., & Bunge, H.-P., 2009. Full seismic waveform tomography for upper-mantle structure in the Australasian region using adjoint methods, *Geophysical Journal International*, **179**, 1703–1725.

Forbriger, T., 2001. Inversion flachseismischer Wellenfeldspektren, Dissertation, Universität Stuttgart.

Forgues, E. & Lambaré, G., 1997. Parametrization study for acoustic and elastic ray + Born inversion, *Journal of Seismic Exploration*, **6**, 253–277.

Gauthier, O., Virieux, J., & Tarantola, A., 1986. Two-dimensional nonlinear inversion of seismic waveforms: numerical results, *Geophysics*, **51**(7), 1387–1403.

Groos, L., Schäfer, M., Forbriger, T., & Bohlen, T., 2012. On the significance of viscoelasticity in a 2D full waveform inversion of shallow seismic surface waves, in *Extended Abstracts, 74th EAGE meeting*, Copenhagen, Denmark, C011.

Guasch, L., Warner, M., Nangoo, T., Morgan, J., Umpleby, A., Stekl, I., & Shah, N., 2012. Elastic 3D full-waveform inversion, in *Extended Abstract, SEG Annual Meeting*, Las Vegas, Nevada.

Kneib, G., 1995. The statistical nature of the upper continental crystalline crust derived from in situ seismic measurements, *Geophysical Journal International*, **122**, 594–616.

Köhn, D., 2011. *Time domain 2D elastic full waveform tomography*, Ph.D. thesis, Christian-Albrechts-Universität zu Kiel.

Köhn, D., De Nil, D., Kurzmann, A., Przebindowska, A., & Bohlen, T., 2012. On the influence of model parametrization in elastic full waveform tomography, *Geophysical Journal International*, **191**, 325–345.

Komatitsch, D. & Martin, R., 2007. An unsplit convolutional perfectly matched layer improved at grazing incidence for the seismic wave equation, *Geophysics*, **72**(5), SM155–SM167.

Kurzmann, A., 2012. *Applications of 2D and 3D full waveform tomography in acoustic and viscoacoustic complex media*, Ph.D. thesis, Karlsruhe Institute of Technology.

Kurzmann, A., Köhn, D., Przebindowska, A., Nguyen, N., & Bohlen, T., 2009. 2D acoustic full waveform tomography: performance and optimization, in *Extended Abstracts,71th EAGE meeting*, Amsterdam, Netherlands.

Lailly, P., 1983. The seismic inverse problem as a sequence of before stack migrations, in *Conference on Inverse scattering, Theory and application, Society for Industrial and Applied Mathematics*, Philadelphia, 206-220.

Lay, T. & Wallace, T., 1995. *Modern global seismology*, vol. 58 of **International Geophysics Series**, Academic Press.

Levander, A., 1988. Fourth-order finite-difference P-SV seismograms, *Geophysics*, **53**, 1425–1436.

Liu, Z., 1994. A velocity smoothing technique based on damped least squares, in *Project Review, May 10, 1994, Consortium Project on Seismic Inverse Methods for Complex Stuctures*.

Métevier, L., Brossier, R., Virieux, J., & Operto, S., 2012. The truncated Newton method for full waveform inversion, *Journal of Physics: Conference Series*, **386**.

Mora, M., 1987. Nonlinear two-dimensional elastic inversion of multioffset data, *Geophysics*, **52**(9), 1211–1228.

Mueller, T. B. & Shapiro, S. A., 2001. Seismic scattering attenuation estimates for the German KTB area derived from well-log statistics, *Geophysical Research Letters*, **28**(19), 3761–3764.

Nihei, K. T. & Li, X., 2007. Frequency response modelling of seismic waves using finite difference time domain with phase sensitive detection (tdpsd), *Geophysical Journal International*, **169**, 1069–1078.

Nocedal, J., 1980. Updating quasi-Newton matrices with limited storage, *Mathematics of computation*, **35**(151), 773–782.

Nocedal, J. & Wright, S. J., 1999. *Numerical optimisation*, Springer series in operations research, Springer-Verlag New York.

Operto, S., Virieux, J., Dessa, J.-X., & Pascal, G., 2006. Crustal imaging from multifold ocean bottom seismometer data by frequency domain full waveform tomography: Application to the eastern nankai trough, *Journal of Geophysical Research*, **111**(B09306).

Operto, S., Virieux, J., Amestoy, P., LExcellent, J.-Y., Giraud, L., & Ben Hadj Ali, H., 2007. 3D finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver: A feasibility study, *Geophysics*, **72**(5), SM195–SM211.

Pica, A., Diet, J., & Tarantola, A., 1990. Nonlinear inversion of seismic reflection data in a laterally invariant medium, *Geophysics*, **55**(3), 284–292.

Plessix, R.-E., 2009. Three-dimensional frequency-domain full-waveform inversion with an iterative solver, *Geophysics*, **74**(6), WCC149–WCC157.

Pratt, R., 1990. Inverse theory applied to multi-source cross-hole tomography II: Elastic wave-equation method, *Geophysical Prospecting*, **38**, 311–330.

Pratt, R. & Worthington, M., 1990. Inverse theory applied to multi-source cross-hole tomography I: Acoustic wave-equation method, *Geophysical Prospecting*, **38**, 287–310.

Pratt, R., Chin, C., & Hicks, G., 1998. Gauss-Newton and full Newton methods in frequency-space seismic waveform inversion, *Geophysical Journal International*, **133**, 341–362.

Pratt, R., Hou, F., Bauer, K., & Weber, M., 2005. Cross-well seismic waveform tomography for monitoring CO_2 injection: a case study fom the Ketzin Site, Germany, *in Scientific Results from the Mallik 2002 Gas Hydrate Production Research Well Program, Mackenzie Delta, Northwest Territories, Canada*, **585**, 14p.

Pratt, R. G., 1999. Seismic waveform inversion in the frequency domain, part 1: Theory and verification in a physical scale model, *Geophysics*, **64**(3), 888–901.

Sears, T., Singh, S., & Barton, P., 2008. Elastic full waveform inversion of multi-component OBC seismic data, *Geophysical Prospecting*, **56**, 843–862.

Sheen, D.-H., Tuncay, K., Baag, C.-E., & Ortoleva, P., 2006. Time domain Gauss-Newton seismic waveform inversion in elastic media, *Geophysical Journal International*, **167**, 1373–1384.

Shin, C., Jang, S., & Min, D.-J., 2001. Improved amplitude preservation for prestack depth migration by inverse scattering theory, *Geophysical Prospecting*, **49**, 592–606.

Shipp, R. & Singh, S., 2002. Two-dimensional full waveeld inversion of wide aperture marine seismic streamer data, *Geophysical Journal International*, **151**, 325–344.

Sirgue, L. & Pratt, R. G., 2004. Efficient waveform inversion and imaging: A strategy for selecting temporal frequencies, *Geophysics*, **69**(1), 231–248.

Sirgue, L., Etgen, J., & Albertin, U., 2008. 3D frequency domain waveform inversion using time domain finite difference methods, in *Extended Abstracts, 70th EAGE meeting*, Rome, Italy, F022.

Sirgue, L., Barkved, O., Dellinger, J., Etgen, J.and Albertin, U., & Kommedal, J., 2010a. Full waveform inversion: the next leap forward in imaging at Valhall, *First Break*, **28**, 65–70.

Sirgue, L., Etgen, T., Albertin, U., & Brandsberg-Dahl, S., 2010b. System and method for 3D frequency domain waveform inversion based on 3D time-domain forward modeling, in *United States Patent*, US 7,725,266 B2.

Symes, W. W., 2007. Reverse time migration with optimal checkpointing, *Geophysics*, **72**(5), SM213–SM221.

Tarantola, A., 1984. Linearized inversion of seismic reflection data, *Geophysical Prospecting*, **32**, 998–1015.

Tarantola, A., 1986. A strategy for nonlinear elastic inversion of seismic reflection data, *Geophysics*, **51**(10), 1893–1903.

Vigh, D., Kapoor, J., Moldoveanu, N., & Li, H., 2011. Breakthrough acquisition and technologies for subsalt imaging, *Geophysics*, **76**(5), WB41–WB51.

Virieux, J., 1986. P-SV wave propagation in heterogeneous media: velocity-stress finite difference method, *Geophysics*, **51**(4), 889–901.

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

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

Wu, R. & Aki, K., 1985. Scattering characteristics of elastic waves by an elastic heterogeneity, *Geophysics*, **50**(4), 582–595.

Zhang, F., Juhlin, C., Cosma, C., Tryggvason, A., & Pratt, R., 2012. Cross-well seismic waveform tomography for monitoring CO_2 injection: a case study fom the Ketzin Site, Germany, *Geophysical Journal International*, **189**, 629–646.

List of Figures

List of Tables

Appendix A

Theory Green's functions

The concept of Green's function for the seismic wave equation is well described by Aki & Richards (1980). The Green's function $G_{in}(\mathbf{x},t;\mathbf{x}',\tau)$ is the solution of the elastic wave equation (see equation 2.7) for a unit impuls in n-direction applied at the location $\mathbf{x}=\mathbf{x}'$ at time $t=\tau$:

$$\rho\frac{\partial^2 G_{in}}{\partial t^2}=\delta_{in}\delta(\mathbf{x}-\mathbf{x}')\delta(t-\tau)+\frac{\partial}{\partial x_j}(C_{ijkl}\frac{\partial}{\partial x_l}G_{kn}) \tag{A.1}$$

With the use of Green's functions, the solution for the n-th component of the displacement caused by a body force $\mathbf{f}$ or surface force T_{jk} can be expressed as

$$u_n(\mathbf{x},t)=\int_V dV\int_\tau d\tau G_{ni}(\mathbf{x},t;\mathbf{x}',\tau)f_i(\mathbf{x}',\tau)+\int_V dV\int_\tau d\tau\frac{\partial}{\partial x'_k}G_{ni}(\mathbf{x},t;\mathbf{x}',\tau)T_{ik}(\mathbf{x}',\tau) \tag{A.2}$$

The Green's functions fulfil spatial reciprocity

$$G_{in}(\mathbf{x},t;\mathbf{x}',\tau)=G_{ni}(\mathbf{x}',t;\mathbf{x},\tau) \tag{A.3}$$

and time reciprocity

$$G_{in}(\mathbf{x},t;\mathbf{x}',\tau)=G_{in}(\mathbf{x},-\tau;\mathbf{x}',-t). \tag{A.4}$$

Danksagung

Zunächst möchte ich meinem Hauptreferenten Thomas Bohlen danken, der diese Arbeit initiierte und mich während der Promotionszeit betreute. Danke für den Glauben an meine Arbeit, die Gespräche und für die Offenheit gegenüber meinen Ideen. Auch die Möglichkeit an verschiedenen Konferenzen teilzunehmen habe ich sehr genossen.

Vielen Dank auch an Hansruedi Maurer von der ETH Zürich für die Übernahme des Korreferates und für das Interesse an meiner Dissertation.

Diese Arbeit wurde innerhalb des Projektes TOAST (TOolbox for Applied Seismic Tomography) gefertigt und ich danke den TOAST-Partnern für die gute Atmosphäre und Zusammenarbeit bei Treffen und Skype-Konferenzen.

Ganz besonders möchte ich André danken, der immer bereit war, meine Probleme und Fragen zu diskutieren, Texte Korrektur zu lesen und mir viele gute Ratschläge zum Gelingen der Arbeit gegeben hat. Danke Martin, für die gute Zeit in unserem gemeinsamen Büro: die Diskussionen, Motivationen und die nötige Ablenkung. Danke auch an Lisa, die sich immer Zeit für meine theoretischen Fragen genommen hat. Claudia, Thomas F., Stefan, Anna, Ines, Anja, Sven und der restlichen Seismik-Gruppe des GPI gilt mein Dank für die tolle Atmosphäre und den starken Zusammenhalt der Arbeitsgruppe, zu denen ihr alle auf eure Weise beigetragen habt. Dass mir die Promotionszeit so viel Freude bereitet hat, liegt nicht zuletzt an euch.

Auf diesem Wege möchte ich auch meinen Eltern und Geschwistern für ihren Glauben in mich danken und dafür, dass sie immer hinter mir stehen.

Ein ganz liebes Dankeschön gilt meinem Mann Jochen, der immer für mich da ist und mich unterstützt und meinem Sohn Oliver, der mich seit der letzten Zeit der Promotion begleitet und mich auch in stressigen Zeiten oft zum Lachen bringt.

Acknowledgements

The study was carried out within the GEOTECHNOLOGIEN program, funded by the German Ministry of Education and Research (BMBF), Grant 03G0752. It was also kindly supported by the sponsors of the Wave Inversion Technology (WIT) Consortium, Germany.

The simulations were performed on the JUROPA cluster at the Jülich supercomputing center and on the Cray XE6 at HLRS, Stuttgart.