University of Alberta

Regularized wave equation migration for imaging and data reconstruction

by

Sam T. Kaplan

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Geophysics

Department of Physics

Fall 2010
Edmonton, Alberta

Examining Committee

Mauricio Sacchi, Physics, University of Alberta

Claire Currie, Physics, University of Alberta

Mathieu Dumberry, Physics, University of Alberta

Frank Marsiglio, Physics, University of Alberta

Dale Schuurmans, Computing Science, University of Alberta

Gary Margrave, Geoscience, University of Calgary

If I choose a single word to describe an approach to the complications of life (and of mathematics too), it would be "optimism." Eventually I realized that, if you allow that word in its mathematical sense too, this whole book is for optimists.

Gilbert Strang (2008)

Abstract

The reflection seismic experiment is a technique used in oil exploration. It uses an active source that propagates energy into the earth. The energy is reflected, and recorded at the earth's surface. The energy is expressed by a wave-field. Recordings of the wave-field are seismic data, and seismic data processing transforms this into estimates of the earth's physical properties. Migration is one component of seismic data processing. It uses a *linearized operator* to estimate the wave-field, and such that the earth's physical property of interest is the *scattering potential*. Migration finds a regularized form of the scattering potential, and of which the most simple is equivalent to using the adjoint of the linearized operator. In the reflection seismic experiment, receivers are placed on a surface such that typical reflection seismic data have four spatial dimensions, and one dimension in time. This means that there are two redundant spatial dimensions in the survey, and this redundancy is reflected in the parameterization of the scattering potential, being either *source-receiver* or *shot-profile*. In source-receiver migration, the parameterization of the scattering potential begs its sparse representation. We use regularized inversion to, in part, apply a sparse expectation to the scattering potential, and which is expressed by its adherence to the Cauchy probability density function. In shot-profile migration, the parameterization of the scattering potential has its redundancy in multiple active energy sources (i.e. the shot locations). With limited numbers of shots, a smallest model regularized inverse representation of the scattering potential gives a more resolved picture of the earth, as compared to the simpler adjoint representation. Moreover, the shot-profile parameterization allows for a data reconstruction algorithm so that limited data can be interpolated/extrapolated. The linearized operators are expensive, encouraging their parallel implementation. For the source-receiver parameterization of the

scattering potential this parallelization is non-trivial, and worth discussion. Seismic data is typically corrupted by various types of noise. Sparse coding can be used to suppress noise prior to migration. It is a method that stems from information theory and that we apply to noise suppression in seismic data.

Acknowledgements

First and foremost, I thank my supervisor, Dr. Mauricio Sacchi. His clarity of thought, and uncanny ability to reduce complicated concepts down to their essential parts plays an important role in my thesis work. I thank my teachers and friends, Dr. Tadeusz Ulrych, Dr. Kristopher Innanen and Dr. Arthur Weglein. Without their mentoring this endeavour would not have been possible. Over the course of my thesis work, I was provided with summer internships by ConocoPhillips, and during which I was encouraged to pursue my research. I am appreciative of their support, and also for help from Dr. Simon Shaw, Dr. Partha Routh, Dr. Richard Day and Dr. Phil Anno. In my Ph.D. studies, I was a member of the SAIG consortium. I thank my supervisor for forming this group, and the consortium sponsors for their financial support. I thank my fellow students for making life in the Physics department enjoyable. In particular, I would like to thank Mostafa Naghizadeh, Evan Bianco, Wolfgang Engler, Natalia Gomez-Perez, Wubshet Alemie and Amsalu Anagaw for many interesting and entertaining discussions. Wubshet Alemie was also responsible for identifying several typographical and consistency errors in the mathematics in this manuscript. In addition, I thank Dr. Claire Currie, Dr. Matthieu Dumberry, Dr. Frank Marsiglio, Dr. Dale Schuurmans, and Dr. Gary Margrave for taking time to critically read my thesis. Finally and importantly, I thank Rhonika and my family for their endless supply of patience, encouragement and support.

Contents

Appendices

List of Tables

List of Figures

List of symbols

The symbols are listed in the order that they appear, and we give the page number of their first appearance.

$\mathcal{L}$	Helmholtz operator	9
∇	Laplacian operator	9
δ	Dirac-delta function	9
$\mathbf{x}_s$	lateral seismic point source location	9
f	frequency distribution of the seismic source	9
$\mathbf{x}_g$	lateral geophone location	9
ψ_d	direct wave-field	9
ψ_s	scattered wave-field	9
G_0	Green's function	10
α	scattering potential	10
c_0	reference wave-speed	10
k_{gz}	receiver side vertical wave-number	12
k_{sz}	source side vertical wave-number	12
u_p	phase-shift wave-field propagator	13
c_1	reference wave-speed for the split-step approximation	16
u_s	split-step correction	17
$\mathcal{F}_{gs}$	four dimensional Fourier transform over lateral survey coordinates (shot and geophone)	17
G_z	WKBJ Green's function	21
ψ_s^W	WKBJ scattered wave-field	21
u_p^W	WKBJ phase-shift wave-field propagator	21

$\alpha^\dagger$	adjoint scattering potential	23
$\mathbf{x}_m$	common depth point position, also known as midpoint	28
$\mathbf{x}_h$	half-offset, also known as offset	28
$\mathcal{F}_{mh}$	four dimensional Fourier transform over midpoint and offset coordinates	29
$\mathbf{p}_h$	offset ray parameter	35
$\mathcal{G}$	wave-field modelling operator, midpoint and ray-parameter	35
$\mathcal{F}_h$	two dimensional Fourier transform over offset coordinates	35
τ_{gz}	vertical travel-time for a plane-wave, receiver side	55
τ_{sz}	vertical travel-time for a plane-wave, source side	55
$\sigma_{n(i)}$	standard deviation of the i^{th} noise component	58
σ_m	scale parameter for the Cauchy distribution	58
$\mathbf{W}_d$	data weighting matrix	59
$\widehat{\mathbf{W}}_d$	data weighting matrix scaled by the standard deviation of the noise	60
ϕ_c	model norm for the Cauchy model prior	62
$\phi_{c,g}$	model norm for the mixed Cauchy-Gaussian model prior	62
$\mathcal{F}_g$	two dimensional Fourier transform over lateral geophone coordinates	79
g	representation of the seismic source for the shot-profile parameterization	79
$u^1_{p(l)}$	phase shift operator for shot-profile migration and de-migration.	79
$u^1_{s(l)}$	split-step correction for shot-profile migration and de-migration	79
x^n_g	nominal geophone grid	102
x^o_g	observed geophone grid	102
Δx^n_g	grid spacing for a nominal geophone grid	102
Δx^o_g	grid spacing for an observed geophone grid	102
III_o	Shah (sampling) function for the observed grid	103
$\alpha^\dagger_p$	p^{th} copy of the adjoint scattering potential due to the sampling function	103
$I\hat{I}I_o$	numerical Shah (sampling) function for the observed grid	105
$\mathbf{W}_m$	model weighting matrix	112
$\mathcal{H}$	wave-field modelling operator for shot-profile migration, single shot	131

■ CHAPTER 1

Introduction

1.1 Motivation

Regularized inversion can be summarized by a single equation that expresses, in mathematics, the relations between cause and effect: Bayes theorem (Bayes, 1763). Bayes theorem,

$$p(m|d) = \frac{p(d|m)p(m)}{p(d)},$$

asks, quite eloquently, what is the chance that some cause m results in the effect d? If the chances are good, then $p(m|d)$ evaluates to a number close to one. If, on the other hand, chances are slim, then $p(m|d)$ will be close to zero. Putting Bayes theorem, as a whole, aside for the moment, it is an understandable desire to study $p(d|m)$, found in the numerator on the right-hand-side of Bayes theorem. It can be used to ask the important question: given some observed cause and effect, along with some hypothesis $p(d|m)$ that connects cause to effect, what is the chance that the hypothesis is true? Indeed, this is an integral component of the scientific method. We leave a discussion of $p(m)$ to Section 1.4. The function $p(d)$ is not integral to our discussion.

If we presume confidence in a scientific hypothesis expressed by $p(d|m)$, then we have an opportunity to modify our question in one of two ways. 1) Given some observed event that we assume to be a cause m, can we predict the chance of the effect d occurring? 2) Given some observed event that we assume to be an effect d, can we predict the chance that it was caused by m? It is the latter of these two questions that we are primarily concerned with in this thesis, and it is the answer to this question that is expressed in the evaluation of $p(m|d)$ in Bayes theorem. This question, along with its eventual answer, also provides a concise description of regularized inversion, and of which regularized migration (the primary subject of this thesis) is an example.

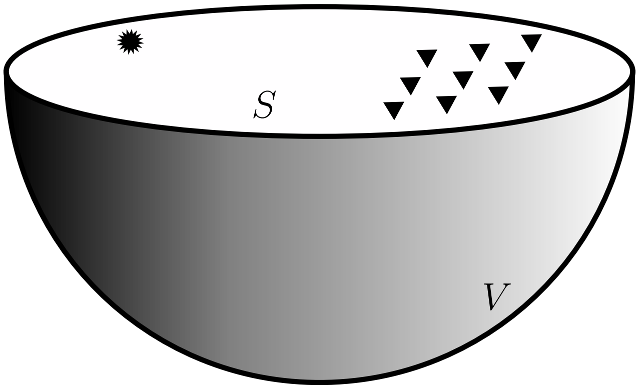

Figure 1.1: A schematic of the seismic experiment for a hemisphere V bounded by the surface S. A single active source (star), and a group of geophones (triangles) are positioned on the top surface of the hemisphere. In modelling the seismic wave-field, we will assume that the radius of the hemisphere goes to infinity.

1.2 The reflection seismic experiment

In this thesis work, m is some property of the earth, $p(d|m)$ contains an approximate description (hypothesis) of how seismic energy propagates through the earth given some corresponding seismic experiment, and d is a measure of this energy recorded at the earth's surface; then, to find m is regularized migration.

We now describe the reflection seismic experiment in less esoteric and more concrete terms. The reflection seismic experiment is an active exploration technique used in, for example, oil exploration. Examples of active sources of energy for the experiment are an air gun (marine experiment) or a Vibroseis truck (land experiment). The energy emanating from a source is described by a seismic wave-field, and is a function of space and time. The wave-field propagates into the earth, reflects, and is recorded at the earth's surface by a collection of hydrophones (or geophones) that are distributed either in the water (marine experiment), or across some swath of land (land experiment). The information recorded at the geophones are seismic data, and are a function of the location of the geophone and time. For the purpose of our experiment, we think of a small parcel of earth as a volume bounded by a surface. The locations of the source and the geophone are each determined by two coordinates, describing their positions on this surface. A cartoon schematic of a reflection seismic experiment is shown in Figure 1.1.

The physics of the experiment is described by a constitutive relation (Hooke's law) between the rank two tensors, stress τ_{ij} and strain ϵ_{kl},

$$\tau_{ij} = \sum_{k,l} c_{ijkl}\epsilon_{kl},$$

along with the momentum equation. The rank 4 tensor c_{ijkl} is called the stiffness tensor. In this thesis work, we use a simplified form of these equations (the scalar wave-equation) by assuming that the wave-field propagates through an isotropic and acoustic medium. There are many introductory text-books that describe these equations and their simplifications (for example, see Shearer (1999)). With the simplification at-hand, the influence of our parcel of earth on the seismic wave-field is described by a single parameter, the acoustic wave-speed. Mapping this experiment back to the language of Bayes theorem, m is some function of the earth's acoustic wave-speed, d is the seismic data recorded by the geophones, and $p(d|m)$ contains an approximation to the relevant physics using the scalar wave-equation. The goal of the experiment, then, is to find some value for acoustic wave-speed (or related quantity) that maximizes its chance of being true. This chance is, in turn and in part, influenced by the recorded seismic data, and the approximation to the physics given by the scalar wave-equation.

As mentioned, the reflection seismic experiment locates sources and geophones on some surface, and the location of each is described by two spatial dimensions. Therefore, in total, the data recorded by the experiment is described by four spatial dimensions, and one dimension in time. The reflection seismic experiment is made robust by collecting data from multiple source locations. Moreover, it may be that the location of the geophones will change with each new source location. It is important to note that traditional seismic data acquisition assumes that energy recorded from any given source does not interfere with energy recorded from any other source. A simple thought experiment should reveal that different source/receiver configurations will sample (illuminate) different portions of the earth. However, nearby configurations will have some intersection of illumination, and in those regions the experiment becomes more robust to noise due to the resulting redundancy of data. This redundancy is well known in the reflection seismic experiment, and is taken advantage of by various established processing methods. It will also have implications for various topics in this thesis, being taken advantage of with two different parameterizations of the equations of interest (Chapter 2 versus Chapter 5).

1.3 Seismic data processing

Traditional seismic data processing separates the physics of wave-field propagation into a sequence of steps. It is widely accepted that the complexity of seismic data makes it difficult to find a single all-encompassing processing method that turns seismic data recorded on some surface into a map of corresponding physical properties of the volume (our parcel of earth). We note that some authors do, indeed, challenge this assertion, and one such challenge is found in the subject of full-waveform inversion (e.g. Pratt et al., 1998, and references therein). This thesis does not challenge the assertion. As such, we can, perhaps arguably, divide seismic data processing into the following roughly ordered sequence of steps: 1) random noise suppression (e.g. Canales, 1984) and data regularization (e.g. Spitz, 1991); 2) direct-wave removal (e.g. Welford and Zhang, 2004); 3) wavelet estimation (e.g. Wiggins, 1978; Weglein and Secrest, 1990); 4) source- and receiver-side de-ghosting (e.g. Amundsen et al., 2005); 5) free-surface and internal-multiple elimination (e.g. Verschuur et al., 1992; Weglein et al., 1997); 6) macro estimation of the wave-speed (e.g. Stork, 1992); 7) refinement of this macro estimation (*migration*); 8) inversion for various earth properties (e.g. Ostrander, 1982). In this thesis we limit ourselves to, primarily, step 7, but also give some attention to steps 1 and 5 in Chapters 6 and 8. It should also be noted that some authors have suggested combining some subset of these steps. For example, Shaw et al. (2004) and Liu et al. (2006) effectively combine steps 6 and 7 using portions of the inverse scattering series.

The reason for the steps that precede the migration in step 7 should become evident as we work our way through the relevant physics in Chapter 2. In short, we use an approximation to the physics that assumes a single scatterer (linear) approximation to the wave-field. This requires the removal of all portions of the seismic wave-field that do not follow this single scattering approximation. Moreover, the linearization is about some macro estimate found in step 6. This can be thought about in analogy to a Taylor series that is truncated at its second term. We note that migration tends to be robust to the absence of steps 1 through 5, likely in part due to the already mentioned redundancy built into the reflection seismic experiment. However, this does not degrade their importance, as success in steps 1 through 5 will improve the migration result, and in some cases their absence may obscure regions of interest in our parcel of earth.

1.4 Regularized migration

In migration (and under the isotropic and acoustic approximation) the property that we solve for is a scalar field occupying the volume illustrated in Figure 1.1. This scalar field has been given different names by different authors. For example, it is routinely called the reflectivity, the migration image, the migrated data, and the scattering potential. In this thesis we will, where appropriate, switch between these names; but for the most part, we will prefer to use the term scattering potential. In part this is because it seems to align most closely with the underlying physics (it is also a choice that is due to the influence of my teachers).

To confuse matters, *migrating seismic data* (to produce *migrated seismic data*, i.e. the scattering potential) can refer to a large number of related algorithms. We do not try to give an overview here. But to avoid confusion, we make the note that in this thesis we work within the realm of pre-stack wave-equation depth migration. Other adjectives such as (but not limited to) *post-stack*, *time*, *reverse-time* and *Kirkhoff* apply to other algorithms. A nice introduction to the meaning of some of these adjectives is given in Sava and Hill (2009).

To say that migration solves for the scattering potential is, perhaps, glib. Rather, we should prefer to say that it approximates the scattering potential. Colloquially, the term migration arguably refers to algorithms that approximate the scattering potential by its adjoint (or in-other-words, the result of applying the adjoint of our physical operator). The terms *true-amplitude migration* (e.g. Deng and McMechan, 2008) and *least-squares migration* (a class of regularized migration, the primary subject of this thesis) have brought attention to the details of the approximation. As mentioned, in regularized migration, the problem can be cast in terms of Bayes theorem. It is generally accepted that migration is an under-determined problem, so that an infinity of scattering potentials will satisfy the observed data. In Bayes theorem, the prior probability density function $p(m)$ is used to add information to the problem such that one solution is chosen from the infinite pool. It adds information to make this choice by asking the question: what is the chance that m is a valid solution (scattering potential) describing our parcel of earth, independent of the experiment?

1.5 Thesis overview

In Chapter 2, we introduce the single scattering approximation to the wave-field using the Born approximation. The Born approximation is a linearized solution for the scalar wave-field (commonly referred to as *de-migration*). To linearize the solution, the Born approximation introduces a reference wave-speed such that the true wave-speed is a perturbation on the reference. This linearization leads to a new quantity, *the scattering potential*, that is our earth property of interest in the volume. The Born approximation is built by the interaction of the scattering potential with Green's functions that are, in turn, constructed from the reference wave-speed. We review an analytic form for the Green's function that allows for lateral variation in the reference wave-speed (the split-step approximation). We allow for vertical variation in the reference wave-speed, ultimately, using a Gazdag algorithm. However, we also investigate the use of WKBJ Green's functions to accommodate this vertical variation. Since the Born approximation is linear, its adjoint is readily derived, and we show this in Chapter 2. Further, we note, importantly, that the adjoint operator has a classic interpretation in geophysics, namely that of *migration.* The construction of the Born approximation and its adjoint is done with a source-receiver parameterization of the wave-field (in Chapter 5, we use a shot-profile parameterization). With the forward and adjoint operators for, respectively, wave-field modelling and migration at hand, regularized migration algorithms can be constructed.

In Chapter 3 we introduce regularized migration with a smallest model constraint applied to the scattering potential. This constraint results in least-squares migration which, in-turn, is the solution to a set of least-squares normal equations. The solution is a smallest model regularized approximation to the scattering potential. We use this chapter to introduce a concept that remains important throughout much of the remainder of the thesis. Namely that of the conjugate gradient method, and the general necessity of an iterative algorithm for solving the normal equations, and which is precipitated by the size of the operators. In Chapter 3, we use eigenvalues and Ritz values to analyze the efficiency of an iterative solution to the normal equations, and use a well known pre-conditioner to try and improve this efficiency.

Chapter 4 exploits the expected distribution of energy in the scattering potential under its source-receiver parameterization. Namely that we expect the energy to be distributed sparsely. This motivates our use of a Cauchy and a mixed Cauchy-Gaussian prior to improve resolution in the scattering potential.

Chapter 5 introduces shot-profile least-squares migration. The development in this chapter is similar to that of Chapter 2, but for the shot-profile parameterization of the scattering potential. In turn, this leads to an alternative implementation of the Born approximation. The shot-profile parameterization of the scattering potential is integral to Chapter 6.

In Chapter 6, we turn our attention to step 1 of the seismic data processing work-flow. Namely, we introduce a data reconstruction algorithm using shot-profile least-squares migration (introduced by Chapter 5). In oder to make minimal assumptions about the earth, we build the wave-field modelling and migration operators using a constant reference wave-speed. Our use of a constant reference wave-speed also results in more efficient migration and de-migration operators.

In Chapter 7 we show a distributed parallel implementation for source-receiver regularized migration. Wave-field modelling and migration are expensive algorithms, motivating their parallel implementation. We note that for the shot-profile parameterization of the scattering potential, a similar parallelization strategy is relatively trivial. While the shot-profile examples in this thesis are run using, in part, parallel computer code, the simplicity of the parallelization does not warrant discussion.

Chapter 8 applies to steps 1 and 5 of the seismic data processing work-flow. We use sparse-coding for random noise and multiple suppression. Chapter 8 can be read independently from the rest of the thesis.

1.6 Contributions

The contributions of this thesis can be summarized as follows. 1) We provide a full algebraic derivation of wave-field modelling and migration operators for regularized migration under the source-receiver parameterization. The operators, in and in themselves, are not new, but were previously given by derivations relying on operator notation. Other algebraic derivations of similar operators exist, but seems to stop at the WKBJ Green's function, not allowing for laterally varying reference wave-speeds. 2) We provide an analysis of least-squares migration, and pre-conditioned least-squares migration using Ritz values. 3) We introduce a novel regularized migration algorithm by applying a Cauchy prior. 4) We give a full algebraic derivation of wave-field modelling and migration operators for regularized migration under the shot-profile parameterization of the scattering potential, and use them in least-squares migration for two different parameterizations of the model. Previously, authors have produced similar operators; but again, derived through operator notation. This is the first time these operators have been applied to least-squares migration. 5) We use shot-profile least-squares migration for a novel data reconstruction algorithm. 6) We introduce a novel parallelization scheme for least-squares migration under the source-receiver parameterization of the scattering potential. 7) We apply sparse-coding (an established method from information theory) to noise removal in seismic data.

1.7 A remark on notational conventions

In this thesis, we will often use the same symbol to denote different variables, but that represent the same physical concept. For example, the symbol $\psi(\mathbf{x}, z|\mathbf{x}_s, z_s; t)$ is the seismic wave-field at spatial location $(\mathbf{x}, z)$ and time t, due to a source excited at position $(\mathbf{x}_s, z_s)$, and time $t = 0$; whereas, the same symbol $\psi(\mathbf{k}_x, z|\mathbf{x}_s, z_s; \omega)$ is its three dimensional Fourier transform over $\mathbf{x}$ and t. Hence, the difference between the symbols is given by the context of their arguments. As a second example, $c_0(\mathbf{x}, z)$ is a three-dimensional reference wave-speed, whereas $c_0(z)$ is a one-dimensional reference wave-speed, invariant to lateral dimensions $\mathbf{x}$. We use lower-case bold symbols to denote vectors, upper-case bold symbols to denote matrices, and non-bold symbols to denote scalar values.

■ CHAPTER 2

Wave-field modelling and migration within the Born approximation

2.1 Introduction

Wave-equation migration uses a linear solution to the wave-equation to find a representation of earth properties (e.g. Cohen and Bleistein, 1979; Stolt and Benson, 1986). In this thesis, we consider the acoustic and non-dispersive (i.e. scalar) wave-equation so that the earth property is acoustic wave-speed. The linear solution to the wave-equation (i.e. the scalar wave-field) derives from the first two terms of the Born (forward scattering) series. In turn, the terms in the Born series consist of Green's functions and a scattering potential. The former describing the propagation of the wave-field through some reference medium, and the later describing where the wave-field is scattered due to divergence of the acoustic wave-speed from the reference medium.

The linear solution for the wave-field provides a linear operator, mapping scattering potential to wave-field. In the language of inverse theory, the scattering potential is the model, the measured wave-field is data, and our linearized solution provides a forward operator mapping model to data. Within this framework, then, there is an adjoint operator (conjugate transpose of the forward operator) providing a mapping from data space to model space. The adjoint operator is, loosely speaking, *migration*, and the forward operator is often called *de-migration.* Migration provides a reasonable representation of the scattering potential; a fact most-often explained in the literature by the *downward continuation of the wave-field* (e.g. Schultz and Claerbout, 1978). In this chapter, we analyze these forward (data modelling) and adjoint (migration) operators, all-the-while remembering that these operators are to be used, for this thesis, within the context of regularized inversion.

We start with a description of the forward operator (wave-field modelling) that is derived from the forward scattering series, and is the Born approximation shown in Section 2.2. The solution makes use of a constant velocity Green's function which we show in Section 2.3. The Green's function is used in conjunction with the Born approximation to find the a-fore-mentioned forward operators. First for an earth where wave-speed varies in depth only using Gazdag operators (Section 2.4.1), and then for an earth where wave-speed is allowed to vary in all dimensions using split-step operators (Section 2.4.2). The corresponding adjoint operators are described in Section 2.6. Finally, in Section 2.7, we discuss the dimensions of the model and data spaces, in particular stating the relation between shot-geophone and midpoint-offset coordinates. In Chapter 3 we discuss the midpoint and ray-parameter geometry for the scattering potential.

In this chapter, we develop operators, specifically, for the source-receiver pre-stack geometry. This means that the scattering potential is parameterized, in its lateral coordinates, by the acquisition geometry of the seismic survey. This will have important consequences in Chapters 3 and 4. In contrast, in Chapter 5, we develop wave-field modelling and migration operators for the shot-profile pre-stack geometry. In the shot-profile geometry, the lateral dimensions of the scattering potential are the lateral dimensions of the earth model. This parameterization is interesting in and in itself, and also leads to a new data reconstruction method (Chapter 6).

2.2 The Born approximation

To build a forward operator for regularized migration, we begin with the perturbed Helmholtz equation governing the propagation of a wave-field through an acoustic medium parametrized by wave-speed $c(\mathbf{x}, z)$, with $\mathbf{x} = (x, y)$. Namely,

$$\mathcal{L}(\psi, c) = \left(\nabla^2 + \frac{\omega^2}{c^2(\mathbf{x}, z)}\right) \psi(\mathbf{x}, z|\mathbf{x}_s, z_s; \omega) = f(\omega)\delta(z - z_s)\delta(\mathbf{x} - \mathbf{x}_s). \tag{2.1}$$

The forcing term $f(\omega)\delta(z - z_s)\delta(\mathbf{x} - \mathbf{x}_s)$ represents a point source in space $(\mathbf{x}_s, z_s)$ with $\mathbf{x}_s = (x_s, y_s)$ and frequency distribution $f(\omega)$. For regularized migration, we want a linear solution for the wave-field ψ as measured by geophones located at some measurement surface $(\mathbf{x}_g, z_g)$. For this purpose, we use scattering (perturbation) theory to find the particular solution to equation 2.1, arriving at the Born approximation (e.g. Clayton and Stolt, 1981),

$$\psi(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega) = \psi_d(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega) + \psi_s(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega), \tag{2.2}$$

where,

$$\psi_d(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega) = f(\omega)G_0(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega), \tag{2.3}$$

and,

$$\begin{aligned}&\psi_s(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s;\omega)\\ &\quad\approx f(\omega)\iint_{-\infty}^{\infty} G_0(\mathbf{x}_g, z_g|\mathbf{x}', z';\omega)(\omega/c_0(\mathbf{x}', z'))^2\alpha(\mathbf{x}', z')G_0(\mathbf{x}', z'|\mathbf{x}_s, z_s;\omega)d\mathbf{x}'dz'.\end{aligned} \tag{2.4}$$

In equations 2.2-2.4, the details of which are given in Appendix A.1, G_0 is a Green's function satisfying $\mathcal{L}(G_0, c_0) = \delta(z-z_s)\delta(\mathbf{x}-\mathbf{x}_s)$, α is a dimensionless quantity called the scattering potential (e.g. Weglein et al., 2003),

$$\alpha(\mathbf{x}, z) = 1 - \frac{c_0^2(\mathbf{x}, z)}{c^2(\mathbf{x}, z)}, \tag{2.5}$$

and c_0 is called the reference wave-speed. ψ_s is the scattered wave-field, and ψ_d is the direct wave-field. For the purpose of regularized migration, equation 2.4 is our linear forward operator mapping α (the model) to ψ_s (the data).

We can ascribe a physical interpretation to equations 2.2-2.4, and the scattering potential α in equation 2.5. In particular, the scattered wave-field ψ_s in equation 2.4 is modelled by propagating energy from the point source (forcing term in equation 2.1) through the reference medium c_0 to some scattering point $(\mathbf{x}', z')$ by way of the Green's function $G_0(\mathbf{x}', z'|\mathbf{x}_s, z_s;\omega)$. Then, the energy propagates from the scattering point to the geophones at the measurement surface by the Green's function $G_0(\mathbf{x}_g, z_g|\mathbf{x}', z';\omega)$. For realizations of $(\mathbf{x}', z')$ in the integral of equation 2.4, the amount of scattered energy is governed by the scattering potential α which, in turn, is a function of the ratio of reference to true wave-speed (equation 2.5). In addition to the scattered wave-field ψ_s there is energy that travels directly from source to geophone without interacting with the scattering potential, and that are described by the direct wave-field, ψ_d in equation 2.3. We assume that the contribution of non-scattered energy in equation 2.2 can be removed from the data, making equations 2.4 and 2.5 a sufficient, albeit linear, description of the physics. Having said that, the direct wave-field will be used in Section 2.4 to propagate energy toward or away from non-zero scattering points. Our interpretation of equations 2.4 and 2.5 will prove useful in understanding the subsequent development of forward and adjoint operators which, in turn, are employed, in later chapters, by regularized migration algorithms.

It is important to emphasize that the solution for the wave-field in equation 2.4 is only valid to first order in the scattering potential α. In fact, the Born approximation of the wave-field is a truncation of an infinite series solution for the wave-field called the forward scattering series (e.g. Weglein et al., 2003). The accuracy of this truncated solution depends on the relation between the true wave-speed c and the reference wave-speed c_0. In general, we want c_0 to be some smoothed version of c. This ensures the validity of the Green's functions in propagating the wave-field to and from potential scattering points. The subject of constructing c_0 belongs to seismic exploration methods in velocity analysis and model building which fall outside the scope of this thesis work. An alternative is to go beyond a linear representation of the wave-field, introducing a nonlinear forward operator (Innanen and Weglein, 2005) into the regularized migration. These nonlinear forward operators can be expressed by series solutions that tend to be computationally expensive. This is certainly an interesting topic, but is beyond the scope of this thesis. Additionally, we note that the Born approximation models a subset of wave-field phenomena. In the language of the reflection seismic experiment, it is limited to a first order description of primary events, whereas events such as free-surface multiples require multiple scattering or, in other words, non-linear terms in the forward scattering series (e.g. Weglein et al., 2003). For example, a first order approximation of a free-surface multiple event requires a second order scattering term (quadratic in α), and a first order approximation of an internal multiple event requires a third order scattering term (cubic in α).

2.3 Green's functions

Construction of a linear forward operator for regularized migration requires the evaluation of equation 2.4 which, in turn, requires some form for the Green's function G_0. When the reference wave-speed is constant there exists an exact analytic solution for G_0 (e.g. DeSanto, 1992). When the reference wave-speed varies in depth z, but is constant in lateral dimensions $\mathbf{x}$, then the WKBJ Green's function is used (e.g. Ghatak et al., 1991). But, when the reference wave-speed varies in both depth and lateral dimensions, a Green's function that is both physically comprehensive and computationally efficient is less obvious, and a multitude of attempts to produce suitable algorithms for this latter case have been proposed (e.g. Gazdag and Sguazzero, 1984; Wenzel, 1991; Stoffa et al., 1990; Kessinger, 1992; Popovici, 1996). In this thesis, we will use the split-step algorithm (Stoffa et al., 1990) to evaluate equation 2.4, which requires a constant velocity Green's function.

The three dimensional constant velocity Green's function satisfies,

$$\mathcal{L}(G_0, c_0) = \delta(\mathbf{x} - \mathbf{x}_s)\delta(z - z_s),$$

where $\mathcal{L}$ is defined by equation 2.1, and c_0 is constant for all space. The derivation results in both causal and anti-causal Green's functions. The latter is often used as a starting point for migration algorithms. However, in this thesis work, we are interested in both data modelling and migration. This makes it useful to proceed using the causal Green's function.

In particular, we find (see Appendix A.2),

$$G_0(\mathbf{k}_{gx}, z_g|\mathbf{x}', z'; \omega) = -\frac{1}{i4k_{gz}} e^{-i\mathbf{k}_{gx}\cdot\mathbf{x}'} e^{ik_{gz}|z'-z_g|} \tag{2.6}$$

$$G_0(\mathbf{x}', z'|\mathbf{k}_{sx}, z_s; \omega) = -\frac{1}{i4k_{sz}} e^{i\mathbf{k}_{sx}\cdot\mathbf{x}'} e^{ik_{sz}|z'-z_s|}, \tag{2.7}$$

where,

$$k_{gz} = sgn(\omega)\sqrt{(\omega/c_0)^2 - \mathbf{k}_{gx}\cdot\mathbf{k}_{gx}} \qquad k_{sz} = sgn(\omega)\sqrt{(\omega/c_0)^2 - \mathbf{k}_{sx}\cdot\mathbf{k}_{sx}}. \tag{2.8}$$

Equation 2.8 are dispersion relations relating frequency ω and horizontal wave-numbers at source $\mathbf{k}_{sx}$ and receiver $\mathbf{k}_{gx}$ locations to their respective vertical wave-numbers k_{sz} and k_{gz}.

Equations 2.6-2.8 describe Green's functions that can be used in the Born solution of the scattered wave-field ψ_s (equation 2.4) when the reference velocity c_0 is constant. That is, when the reference velocity is constant, they can be used with the Born approximation to construct a forward operator, mapping scattering potential α to the measured scattered wave-field ψ_s.

2.4 Forward operators for variable reference velocities

To derive a forward operator that allows for the reference medium c_0 to vary in space (depth z and lateral $\mathbf{x}$ directions), we proceed in two stages. The first stage finds a data modelling equivalent of Gazdag migration, and the second stage arrives at a data modelling equivalent of split-step migration. In the first stage, we take a reference velocity that varies in depth only so that $c_0(\mathbf{x}, z) = c_0(z)$. Given such a medium, it would seem that the WKBJ Green's function should be used. In such a regime it is valid for high frequencies in ψ and smoothness in $c_0(z)$, or some combination thereof. However, in the second stage we introduce lateral variation into the reference velocity by breaking this assumption of depth only variability in c_0. In this case, it turns out to be better to approximate $c_0(z)$ with a piece-wise constant function so that within each domain of constant velocity, the Green's function described in the previous section are applicable. In the case where c_0 varies in depth only, the Green's functions from the piece-wise representation of $c_0(z)$ and the WKBJ Green's function from the continuous representation of $c_0(z)$ produce loosely equivalent forward operators. But, when we introduce lateral variation into the reference velocity through split-step modelling, the latter choice proves to be more accurate.

2.4.1 Gazdag modelling

Figure 2.1 illustrates the Gazdag modelling algorithm for the forward operator of regularized migration. We call it such since it is, roughly speaking, based on the migration operator described by Gazdag (1978). In Gazdag modelling the reference velocity is one dimensional, being a function of depth only. Then, this one dimensional reference velocity $c_0(z)$ is partitioned into n_z domains (layers),

$$D_l = \{z \in \mathcal{R} | 0 \leq z_{l-1} \leq z < z_l\} \quad , \quad l = 1 \ldots n_z, \tag{2.9}$$

and within each layer there is a corresponding constant reference wave-speed $c_{0(l)}$ approximating $c_0(z)$ for $z_{l-1} \leq z < z_l$, and a constant velocity Green's function $G_{0(l)}$. Each layer contributes to the total scattered wave-field ψ_s such that the contribution from the l^{th} layer is denoted $\psi_{s(l)}$. Then, by super-position it must be that,

$$\psi_s = \psi_{s(1)} + \psi_{s(2)} + \cdots + \psi_{s(n_z)}. \tag{2.10}$$

We proceed to describe $\psi_{s(1)}$ and $\psi_{s(2)}$, contributions to the scattered wave-field from the scattering potential housed within, respectively, D_1 and D_2. We use these to infer a general solution for $\psi_{s(l)}$, the contribution from D_l.

In the first layer D_1, we apply the scattered wave-field component of the Born approximation in equation 2.4 so that upon substitution of $G_{0(1)}$, we find with $z_g = z_s = z_0$,

$$\psi_{s(1)}(\mathbf{k}_{gx}, z_0 | \mathbf{k}_{sx}, z_0; \omega) = f(\omega) \int_{z_0}^{z_1} u_{p(1)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z', \omega) \frac{\omega^2}{c_{0(1)}^2} \alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z') dz', \tag{2.11}$$

where $\mathbf{k}_{sx}$ and $\mathbf{k}_{gx}$ are the Fourier conjugate variables of $\mathbf{x}_s$ and $\mathbf{x}_g$, respectively. The function $u_{p(1)}$ is,

$$u_{p(l)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z', \omega) = -\frac{e^{i(k_{gz(l)} + k_{sz(l)})(z' - z_{l-1})}}{16 k_{gz(l)} k_{sz(l)}}, \tag{2.12}$$

for $l = 1$, and where $k_{gz(1)}$ and $k_{sz(1)}$ are given by the dispersion relations,

$$k_{gz(l)} = sgn(\omega) \sqrt{(\omega / c_{0(l)})^2 - \mathbf{k}_{gx} \cdot \mathbf{k}_{gx}} \tag{2.13}$$

$$k_{sz(l)} = sgn(\omega) \sqrt{(\omega / c_{0(l)})^2 - \mathbf{k}_{sx} \cdot \mathbf{k}_{sx}}, \tag{2.14}$$

for $l = 1$. Equation 2.12 is constructed from $G_{0(l)}$, governing the propagation of energy in D_l. The detailed relation between $G_{0(l)}$ and $u_{p(l)}$ are presented in Appendix A.3. In short, and as described previously, a Green's function propagates the energy from the point source

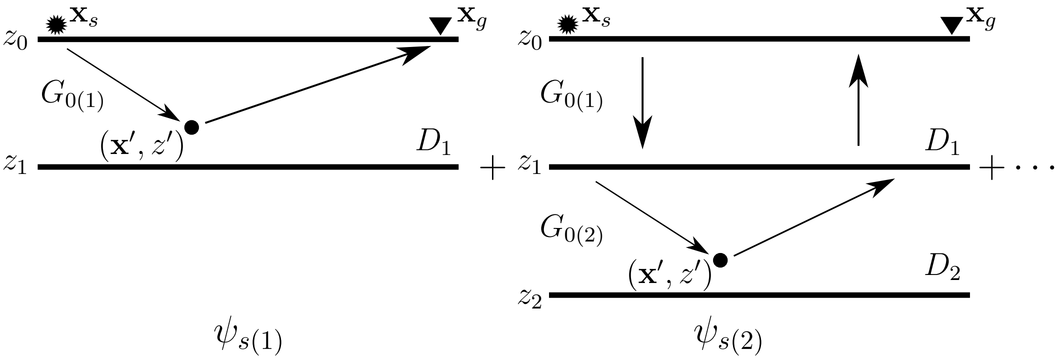

Figure 2.1: We give a schematic description of the first two terms in Gazdag wave-field modelling, $\psi_{s(1)}$ and $\psi_{s(2)}$. In general $\psi_{s(l)}$ is built from the scattering potential in D_l.

to all potential scattering points, and a second Green's function propagates energy from the scattering points back to the measurement surface. This explanation is also illustrated in the first term of Figure 2.1.

The construction of $\psi_{s(2)}$ is illustrated in the second term of Figure 2.1, and depicts the following three steps used in its computation.

- First, we consider the wave-field in D_1. We compute the direct wave-field (see equation 2.3) at z_1 due to the source at z_0. This represents the direct propagation, using $G_{0(1)}$, of the wave-field from the point source at z_0 to the bottom of D_1 (top of D_2). We denote this wave-field as $\psi_{d(1)}(\mathbf{x}', z_1|\mathbf{x}_s, z_0; \omega)$.

- Second, we consider the wave-field in D_2. We compute the scattered wave-field at z_1 due to the scattering potential within D_2, and the boundary condition at the top of D_2 given by $\psi_{d(1)}(\mathbf{x}', z_1|\mathbf{x}_s, z_0; \omega)$. Here, the energy is propagated according to $G_{0(2)}$, and we will denote the resulting wave-field as $\psi_{s(2,1)}(\mathbf{x}', z_1|\mathbf{x}'', z_0; \omega)$.

- Third, we again consider the wave-field in D_1. This time, we compute the direct wave-field at z_0 due to the boundary condition at the bottom of D_1 given by $\psi_{s(2,1)}(\mathbf{x}', z_1|\mathbf{x}'', z_0; \omega)$. Here, the energy is propagated according to $G_{0(1)}$, and the resulting wave-field is $\psi_{s(2)}(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega)$.

In particular, the sum total of the above three steps gives the following contribution to the scattered wave-field from the second layer (see Appendix A.3),

$$\begin{aligned}\psi_{s(2)}&(\mathbf{k}_{gx}, z_0|\mathbf{k}_{sx}, z_0; \omega)\\ &= f(\omega)u_{p(1)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z_1, \omega)\int_{z_1}^{z_2} u_{p(2)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z', \omega)\frac{\omega^2}{c_{0(2)}^2}\alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z')dz',\end{aligned} \tag{2.15}$$

where $u_{p(2)}$ is given by equation 2.12 for $l = 2$. Generalizing to the l^{th} layer (D_l), we find,

$$\begin{aligned}&\psi_{s(l)}(\mathbf{k}_{gx}, z_0|\mathbf{k}_{sx}, z_0; \omega)\\&\quad= f(\omega)u_{p(1)}u_{p(2)}\cdots u_{p(l-1)}\int_{z_{l-1}}^{z_l} u_{p(l)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z', \omega)\frac{\omega^2}{c_{0(l)}^2}\alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z')dz',\end{aligned} \tag{2.16}$$

where, again, $u_{p(l)}$ is given by equation 2.12. Note that in equation 2.16, the functions $u_{p(l)}$ falling outside the integral are evaluated for $z' = z_l$. In equations 2.15 and 2.16, $\mathbf{k}_{gx}$ and $\mathbf{k}_{sx}$ are, respectively, the Fourier conjugate variables of $\mathbf{x}_g$ and $\mathbf{x}_s$. Fourier kernels, identified in the constant velocity Green's function, make $\mathbf{k}_{gx} - \mathbf{k}_{sx}$ the Fourier conjugate variable of $\mathbf{x}'$. This makes an important connection between the scattering potential α and the lateral wave-numbers at the source $\mathbf{k}_{sx}$ and receiver $\mathbf{k}_{gx}$, effectively connecting the model to the source and receiver positions of the experiment. This is the source-receiver parameterization of the scattering potential, and is a direct consequence of the representation of the Green's functions in equations 2.6 and 2.7. We discuss this point further in Section 2.7, and in Chapter 5 we discuss an alternative shot-profile parameterization of the scattering potential.

Together, equations 2.10, 2.12 and 2.16, along with the dispersion relations in equations 2.13 and 2.14, are Gazdag data modelling for the source-receiver parameterization of the scattering potential. They are one example of a linear forward (wave-field modelling) operator for regularized migration. The equations are constructed in lateral wave-number rather than lateral space for the sake of efficiency. This is illustrated in Appendix A.3 where multiple integrations in space are replaced by Fourier transforms. In the next section, we will introduce lateral variation in the reference velocity model using the data modelling analogy to split-step migration, which maintains some of the efficiency gained through our use of Fourier transforms.

2.4.2 Split-step modelling

To allow for lateral variation in the reference velocity (i.e. a three dimensional reference velocity model $c_0(\mathbf{x}, z)$), we employ the data modelling analog of split-step migration. The method is a straight-forward extension of Gazdag modelling, but has seemingly significant physical limitations. Namely that the resulting forward operator is only valid for near vertical travelling plane wave components, and that the lateral variation in velocity is small. Despite these limitations, the algorithm performs remarkably well (a fact that likely deserves further investigation).

The well-know split-step algorithm (Feit and J. A. Fleck (1978), and Thomson and Chapman (1983), and presented for migration in Stoffa et al. (1990)) is derived by manipulation of the dispersion relations in equations 2.13 and 2.14. In particular, we let $k_{gz(l)}$ and $k_{sz(l)}$

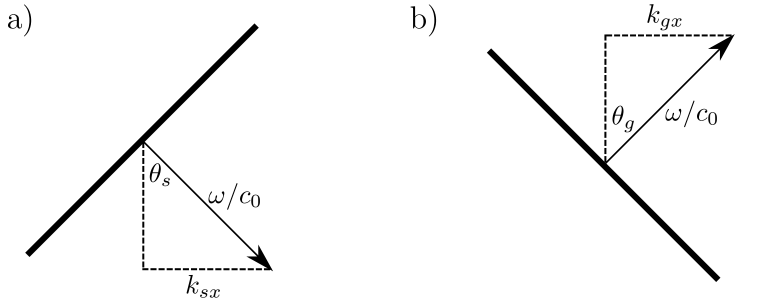

Figure 2.2: We illustrate plane-wave components of the wave-field for a) a plane-wave propagating from the source toward a scattering point, and b) for a plane-wave propagating from the scattering point toward the measurement surface. The split-step approximation is most readily applicable when θ_s and θ_g are small.

be functions of slowness $c_{0(l)}^{-1}$ which, in turn, is allowed to vary in the lateral dimensions $\mathbf{x}$. Then, we define a constant $c_{1(l)}^{-1}$, usually taken to be the average of $c_{0(l)}^{-1}(\mathbf{x})$, and take the Taylor expansions of $k_{gz(l)}$ and $k_{sz(l)}$ about $c_{1(l)}^{-1}$. Here, we consider $k_{gz(l)}$ which truncating at the first term of its Taylor expansion gives,

$$\begin{aligned} k_{gz(l)}(c_{0(l)}^{-1}) &\approx k_{gz(l)}(c_{1(l)}^{-1}) + \omega\left[1 - \left|c_{0(l)}(\mathbf{x}_g)\mathbf{k}_{gx}/\omega\right|^2\right]^{-1/2}\left(c_{0(l)}^{-1}(\mathbf{x}_g) - c_{1(l)}^{-1}\right) \\ &\approx k_{gz(l)}(c_{1(l)}^{-1}) + \omega\left(c_{0(l)}^{-1}(\mathbf{x}_g) - c_{1(l)}^{-1}\right), \end{aligned} \tag{2.17}$$

where $|\mathbf{x}|$ denotes the norm $\sqrt{\mathbf{x}^T\mathbf{x}}$, and where we have set $c_{0(l)}(\mathbf{x}_g)\mathbf{k}_{gx}/\omega$ to $\mathbf{0}$. This is reasonable when either ω is large or k_{gx} is small. The physical significance of this is two-fold. First, it breaks the assumption which allowed us to use the constant velocity Green's function.[1] Second, it means that the angle $\theta_{g(l)}$ made by the intersection of the normal to the plane wave component of the wave-field parametrized by this term and the vertical axis, given by,

$$\sin(\theta_{g(l)}) = c_{0(l)}(\mathbf{x}_g)k_{gx}/\omega, \tag{2.18}$$

must be small (see Figure 2.2). The same assumption is made for the angle formed by the k_{gy} component of the plane-wave. A similar derivation and analysis can be made for the source-side plane-waves, and $k_{sz(l)}$.

The computational significance of equation 2.17 is an algorithm that can employ the fast Fourier transform. The dispersion relation has been split into two parts in equation 2.17: a part that depends only on horizontal wave-numbers $\mathbf{k}_{gx}$, and a part that depends only on lateral space $\mathbf{x}_g$. In particular, this split allows us to re-write the l^{th} term in Gazdag

[1] Incidentally, this is likely the reason why it is not beneficial to employ WKBJ Green's functions (see Section 2.5).

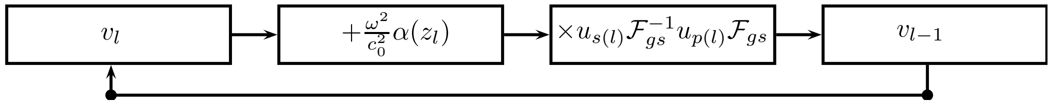
v_l
$+\frac{\omega^2}{c_0^2}\alpha(z_l)$
$\times u_{s(l)}\mathcal{F}^{-1}_{gs}u_{p(l)}\mathcal{F}_{gs}$
v_{l-1}

Figure 2.3: We illustrate the split-step modelling operator in equation 2.23. The algorithm begins at depth ($l = n_z$), and terminates the recursion at the measurement surface, $l = 0$.

modelling (equation 2.16) with its split-step correction so that if we let,

$$\alpha(k_{gx}, k_{sx}, z') = \alpha(k_{gx} - k_{sx}, z'), \tag{2.19}$$

then,

$$\psi_{s(l)}(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s;\omega) = \left(\frac{1}{2\pi}\right)^{4l} f(\omega)(u_{s(1)}\mathcal{F}^*_{gs}u_{p(1)}\mathcal{F}_{gs})\cdots(u_{s(l-1)}\mathcal{F}^*_{gs}u_{p(l-1)}\mathcal{F}_{gs}) \times \int_{z_{l-1}}^{z_l} (u_{s(l)}(\mathbf{x}_g,\mathbf{x}_s,z',\omega)\mathcal{F}^*_{gs}u_{p(l)}(\mathbf{k}_{gx},\mathbf{k}_{sx},z',\omega)\mathcal{F}_{gs})\frac{\omega^2}{c^2_{1(l)}}\alpha(\mathbf{x}_g,\mathbf{x}_s,z')dz', \tag{2.20}$$

where $\mathcal{F}_{gs}$ denotes the four dimensional Fourier transform over $\mathbf{x}_s$ and $\mathbf{x}_g$, $\mathcal{F}^*_{gs}$ is its adjoint, so that the corresponding inverse Fourier transform is $(2\pi)^{-4}\mathcal{F}^*_{gs}$. In equation 2.20, $u_{s(l)}$ is,

$$u_{s(l)}(\mathbf{x}_g, \mathbf{x}_s, z', \omega) = e^{i\omega(c^{-1}_{0(l)}(\mathbf{x}_g)+c^{-1}_{0(l)}(\mathbf{x}_s)-2c^{-1}_{1(l)})(z'-z_l)}. \tag{2.21}$$

We call $u_{p(l)}$ the *phase-shift* with $u_{s(l)}$ being its *split-step correction.* As before, when $u_{p(l)}$ and $u_{s(l)}$ fall outside the depth integral, they are evaluated for $z' = z_l$. Together, equations 2.10, 2.20, and 2.21 are split-step wave-field modelling, and within the context of regularized migration constitute a linear forward operator mapping scattering potential α to the measured wave-field ψ_s. Importantly, it is an algorithm that allows for a reference velocity to vary in all dimensions. The consequence of equation 2.19 is discussed in Section 2.7, and is the source-receiver parameterization of the scattering potential.

2.4.3 Numerical implementation of split-step modelling

An efficient numerical implementation of split-step modelling is found by recognizing a recursion in its operator. We substitute equation 2.20 into equation 2.10, re-arrange terms, and approximating the depth integrals so that,

$$\begin{aligned}\psi_s(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega) = \Delta z \left(\frac{1}{2\pi}\right)^{4l} f(\omega) u_{s(1)} \mathcal{F}^*_{gs} u_{p(1)} \mathcal{F}_{gs} \left(\frac{\omega^2}{c^2_{1(1)}} \alpha(\mathbf{x}_g, \mathbf{x}_s, z_1) + u_{s(2)} \mathcal{F}^*_{gs} u_{p(2)}\right. \\ \times \mathcal{F}_{gs} \left(\frac{\omega^2}{c^2_{1(2)}} \alpha(\mathbf{x}_g, \mathbf{x}_s, z_2) + \cdots + u_{s(n_z)} \mathcal{F}^*_{gs} u_{p(n_z)} \mathcal{F}_{gs} \frac{\omega^2}{c^2_{1(n_z)}} \alpha(\mathbf{x}_g, \mathbf{x}_s, z_{n_z})\right) \cdots \Bigg),\end{aligned} \tag{2.22}$$

where we have assumed that $z_l - z_{l-1} = \Delta z$, $l = 1 \dots n_z$. In equation 2.22, we recognize a recursion, so that for $l = n_z \dots 2$,

$$\begin{aligned}\psi_s(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega) &= \Delta z \left(\frac{1}{2\pi}\right)^{4l} f(\omega) v_1 \\ v_{l-1} &= u_{s(l)} \mathcal{F}^*_{gs} u_{p(l)} \mathcal{F}_{gs} \left(v_l + \frac{\omega^2}{c^2_{1(l)}} \alpha(\mathbf{x}_g, \mathbf{x}_s, z_l)\right),\end{aligned} \tag{2.23}$$

and where $v_{n_z} = 0$. Thus the wave-field modelling operator iterates over the depth dimension, starting at depth and terminating at the measurement surface. The algorithm is illustrated by the flow diagram in Figure 2.3.

As an example, we consider the often studied Marmousi model (e.g. Versteeg, 1993). The Marmousi model is two dimensional (x, z) so that the lateral dimensions $\mathbf{x}_g = (x_g, y_g)$ simplify to x_g. Similar simplifications are made to the source coordinates $\mathbf{x}_s$, and the corresponding Fourier conjugate dimensions. In Figure 2.4a, we plot the Marmousi velocity model $c(x, z)$. The units of the colour-scale are metres per second. In Figures 2.4b-d, we plot near offset data sections obtained using various wave-field modelling algorithms. That is, we plot approximations to $\psi_s(x_g, 0|x_s, 0; t)$ for all (x_g, x_s) satisfying $x_g - x_s = 400m$. In all four panels, the horizontal axis is the common depth point (CDP) position given by $x_m = (x_s + x_g)/2$ (see Figure 2.5 for an illustration). In Figure 2.4a the vertical axis is depth z, and in Figures 2.4b-d, the vertical axis is two-way travel time t. In Figure 2.4b, ψ_s is approximated using finite difference modelling as described by, for example, Versteeg (1994). In Figure 2.4c, we use Gazdag wave-field modelling, and in Figure 2.4d, we use split-step wave-field modelling, . It should be evident that the split-step modelling algorithm does a reasonable job in replicating the finite difference data, especially when compared to the Gazdag wave-field modelling result in Figure 2.4c. For example, notice the kinematics of the event indicated by the arrow in Figures 2.4b-d.

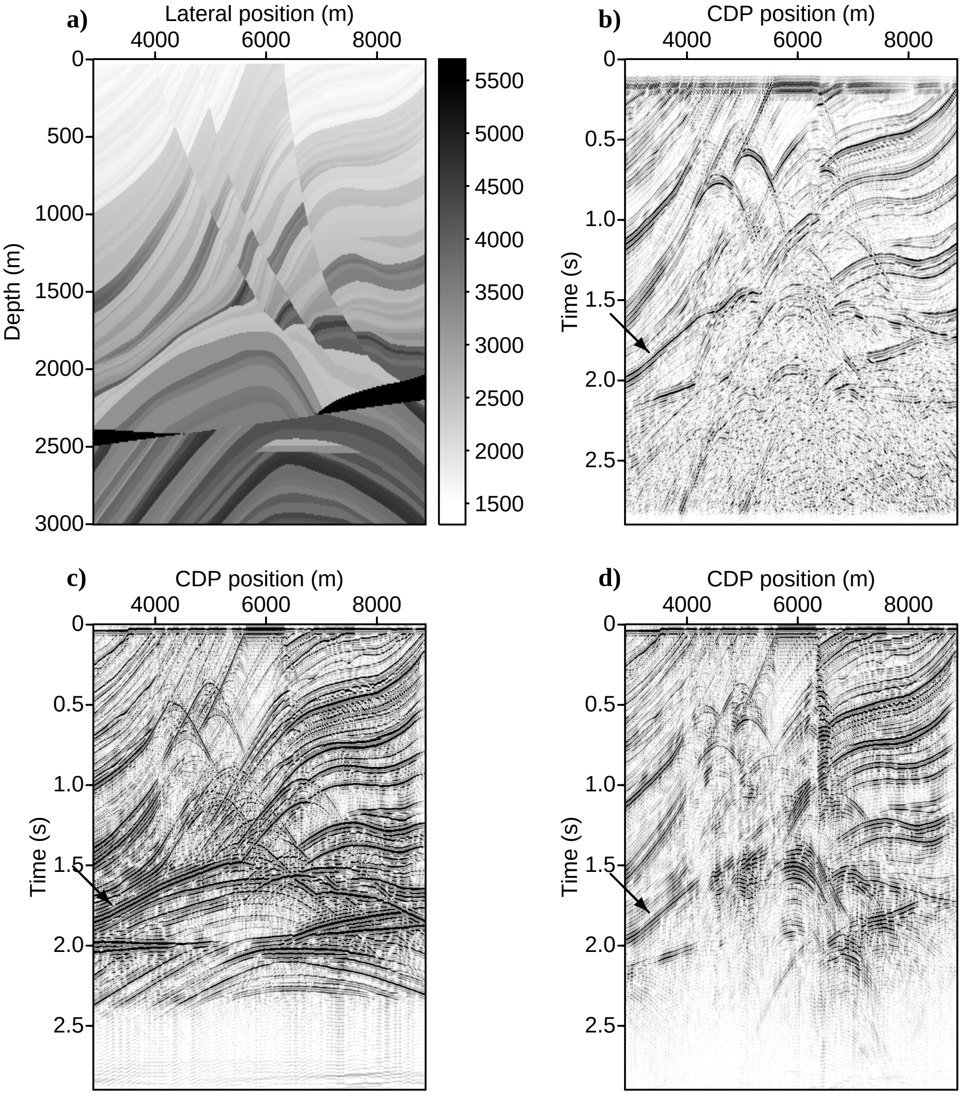

Figure 2.4: a) The Marmousi velocity model, and b)-d) common offset data sections of the wave-field. The data are computed using b) finite difference, c) Gazdag modelling, and d) split-step modelling. The units of the colour-bar in a) are metres per second. Arrows in b)-d) indicate a seismic event whose kinematics are comparably modelled by finite difference and split-step modelling. In contrast, and as expected, this same event is poorly modelled by Gazdag modelling.

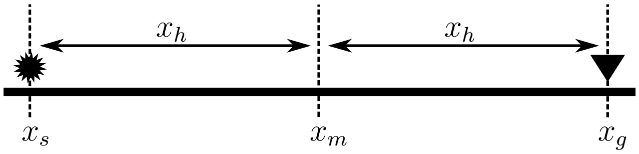

Figure 2.5: We illustrate the geometry used in data acquisition, showing the definition of half-offset x_h, and common depth point (CDP) location x_m in terms of source x_s and geophone x_g positions.

2.5 Wave-field modelling with WKBJ Green's functions

In Section 2.4.1, Gazdag wave-field modelling was derived from a piece-wise constant reference velocity model $c_{0(l)}$ where within each domain of constant velocity, a constant velocity Green's function is used to describe the propagation of the wave-field. Then, in Section 2.4.2 the assumption of constant velocity Green's functions is violated by introducing lateral variation into the reference velocity using the split-step approximation. Here, we derive a similar algorithm, but allow $c_0(z)$ to be continuous through use of a WKBJ Green's function. We show that it differs from the split-step algorithm in Section 2.4.1, and by example, that this difference makes it less accurate.

2.5.1 WKBJ Green's functions

The WKBJ Green's function satisfies $\mathcal{L}(G_z, c_0(z)) = \delta(\mathbf{x} - \mathbf{x}_s)\delta(z - z_s)$ where $\mathcal{L}$ is defined by equation 2.1, and $c_0(z)$ is constant in the lateral dimensions $\mathbf{x}$. As in Section 2.3, the derivation results in both causal and anti-causal Green's functions. Unlike Section 2.3, the WKBJ Green's function is approximate. The validity of its approximation depends on two factors. First, in the band-width of frequencies used in the operator; and second, in the smoothness of the reference velocity $c_0(z)$. The first factor is shown in Appendix A.4 where the derivation of the WKBJ Green's function includes the truncation of a power series in inverse-frequency ω^{-1}. The error in the approximation, due to the truncation, decreases as ω increases. Therefore, the WKBJ Green's function can be thought of as a high frequency approximation. The second factor is not shown in Appendix A.4, but can be shown by analysis of the coefficients of the power series (e.g. Ghatak et al., 1991).

In particular, the WKBJ Green's function is found by truncating the power series at the

ω^{-2} term, so that (see Appendix A.4),

$$
\begin{aligned}
G_z(\mathbf{k}_{gx}, z_g|\mathbf{x}', z'; \omega) &= -\frac{1}{i4\sqrt{k_{gz}(z_g)k_{gz}(z')}} e^{-i\mathbf{k}_{gx}\cdot\mathbf{x}'} e^{i\int_{z_g}^{z'} k_{gz}(z')dz'} & (2.24)\\
G_z(\mathbf{x}', z'|\mathbf{k}_{sx}, z_s; \omega) &= -\frac{1}{i4\sqrt{k_{sz}(z')k_{sz}(z_s)}} e^{i\mathbf{k}_{sx}\cdot\mathbf{x}'} e^{i\int_{z_s}^{z'} k_{sz}(z')dz'}, & (2.25)
\end{aligned}
$$

where,

$$
\begin{aligned}
k_{gz}(z) &= sgn(\omega)\sqrt{(\omega/c_0(z))^2 - \mathbf{k}_{gx}\cdot\mathbf{k}_{gx}} & (2.26)\\
k_{sz}(z) &= sgn(\omega)\sqrt{(\omega/c_0(z))^2 - \mathbf{k}_{sx}\cdot\mathbf{k}_{sx}}. & (2.27)
\end{aligned}
$$

Equations 2.24-2.27 describe WKBJ Green's functions that can be used in the Born solution of the scattered wave-field ψ_s (equation 2.4) when the reference velocity $c_0(z)$ varies in depth only.

2.5.2 WKBJ and split-step modelling

Applying the WKBJ Green's functions in equations 2.24 and 2.25 to the Born approximation in equation 2.4, we find (after recognizing Fourier transforms over the lateral directions $\mathbf{x}$),

$$\psi_s^W(\mathbf{k}_{gx}, z_0|\mathbf{k}_{sx}, z_0; \omega) = f(\omega)\int_{-\infty}^{\infty} u_p^W(\mathbf{k}_{gx}, \mathbf{k}_{sx}, \omega, z_0, z')\frac{\omega^2}{c_0^2(z')}\alpha(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z')dz', \quad (2.28)$$

where,

$$u_p^W(\mathbf{k}_{gx}, \mathbf{k}_{sx}, \omega, z_0, z') = -\frac{e^{i\int_{z_0}^{z'}(k_{gz}(z'')+k_{sz}(z''))dz''}}{16\sqrt{k_{sz}(z_s)k_{sz}(z')k_{gz}(z_g)k_{gz}(z')}}. \quad (2.29)$$

In equations 2.28 and 2.29 we use the super-script W to denote the WKBJ solution, and as before, we have let $z_g = z_s = z_0$.

In this section, we will show the similarity between WKBJ wave-field modelling in equations 2.28 and 2.29, and Gazdag wave-field modelling (equations 2.10 and 2.16). Then, we introduce the split-step approximation into the WKBJ wave-field modelling equation, calling the resulting modelling algorithm WKBJ+SS. In doing so it becomes evident that split-step wave-field modelling and WKBJ+SS wave-field modelling are not equivalent. This is illustrated by partitioning the depth axis into two domains (layers), $D_1 = \{z_0 \leq z < z_1\}$ and $D_2 = \{z_1 \leq z < z_2\}$, and calculating the resulting wave-fields using the two methods. Finally, we show, by example, that split-step wave-field modelling is more accurate than WKBJ+SS wave-field modelling. We conjecture that the reason for this is due to a nonlinearity introduced into the error terms of each of the respective modelling algorithms.

We start by partitioning the integral in equation 2.28 using D_1 and D_2 such that,

$$\begin{aligned}\psi_s^W(\mathbf{k}_{gx}, z_g|\mathbf{k}_{sx}, z_s; \omega) \approx f(\omega)\Big[\bar{u}^W_{p(1)} \int_{z_0}^{z_1} \frac{\omega^2}{c^2_{0(1)}}\alpha(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z')dz' \\ + \bar{u}^W_{p(2)}u^W_{p(1)} \int_{z_1}^{z_2} \frac{\omega^2}{c^2_{0(2)}}\alpha(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z')dz'\Big],\end{aligned} \tag{2.30}$$

where,

$$u^W_{p(l)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}) = e^{i(k_{gz}(z_l)+k_{sz}(z_l))(z_l-z_{l-1})}, \tag{2.31}$$

and,

$$\bar{u}^W_{p(l)} = -\frac{u^W_{p(l)}}{16\sqrt{k_{sz}(z_s)k_{sz}(z_l)k_{gz}(z_g)k_{gz}(z_l)}}, \tag{2.32}$$

for $l = 1, 2$, and we have approximated the integral in the exponential functions by Riemann sums. Equations 2.30-2.32 show that, modulus an amplitude factor, $\psi^W_s \approx \psi_s$. That is, the WKBJ modelled wave-field is roughly equivalent to the Gazdag modelled wave-field. In particular, the interested reader will make a comparison with equations 2.10 and 2.16 for $l = 1, 2$.

Next, we introduce the split-step approximation into equation 2.28 to arrive at the WKBJ+SS wave-field,

$$\psi_s^W(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega) \approx \left(\frac{1}{2\pi}\right)^4 f(\omega) \int_{-\infty}^{\infty} u^W_s \mathcal{F}^*_{gs} u^W_p \mathcal{F}_{gs} \frac{\omega^2}{c^2_0(z')}\alpha(\mathbf{x}_g, \mathbf{x}_s, z')dz', \tag{2.33}$$

where, as before, $\mathcal{F}_{gs}$ is the four-dimensional Fourier transform over $\mathbf{x}_g$ and $\mathbf{x}_s$. The split-step correction u^W_s is,

$$u^W_s(\mathbf{x}_g, \mathbf{x}_s, \omega, z_0, z') = e^{i\omega\int_{z_0}^{z'}(c_0^{-1}(\mathbf{x}_g,z'')+c_0^{-1}(\mathbf{x}_s,z'')-2c_1^{-1}(z''))dz''}, \tag{2.34}$$

where $c_1^{-1}(z'')$ is taken to be the average of $c_0^{-1}(\mathbf{x}, z'')$, over the lateral $\mathbf{x}$ dimensions. Again, we partition the integral in equation 2.33 using D_1 and D_2 so that,

$$\begin{aligned}\psi_s^W(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega) \approx \left(\frac{1}{2\pi}\right)^4 f(\omega)\Bigg[u^W_{s(1)}\mathcal{F}^*_{gs}\bar{u}^W_{p(1)}\mathcal{F}_{gs} \int_{z_0}^{z_1} \frac{\omega^2}{c^2_{0(1)}}\alpha(\mathbf{x}_g, \mathbf{x}_s, z')dz' \\ +u^W_{s(2)}u^W_{s(1)}\mathcal{F}^*_{gs}\bar{u}^W_{p(2)}u^W_{p(1)}\mathcal{F}_{gs} \int_{z_1}^{z_2} \frac{\omega^2}{c^2_{0(2)}}\alpha(\mathbf{x}_g, \mathbf{x}_s, z')dz'\Bigg],\end{aligned} \tag{2.35}$$

where $u^W_{p(l)}$ and $\bar{u}^W_{p(l)}$ are defined in equations 2.31 and 2.32, and,

$$u^W_{s(l)}(\mathbf{x}_g,\mathbf{x}_s,\omega) = e^{i\omega(c_0^{-1}(\mathbf{x}_g,z_l)+c_0^{-1}(\mathbf{x}_s,z_l)-2c_1^{-1}(z_l))(z_l-z_{l-1})}. \tag{2.36}$$

The first right-hand-side term in equation 2.35 is equivalent to split-step wave-field modelling (equations 2.20 and 2.21). In contrast, the second right-hand-side term is not equivalent to split-step wave-field modelling. Namely because the terms in the operator preceding the integration do not commute. That is,

$$\begin{aligned} u^W_{s(2)}u^W_{s(1)}\mathcal{F}^*_{gs}\bar{u}^W_{p(2)}u^W_{p(1)}\mathcal{F}_{gs} &= u^W_{s(2)}u^W_{s(1)}\mathcal{F}^*_{gs}\bar{u}^W_{p(2)}\mathcal{F}_{gs}\mathcal{F}^*_{gs}u^W_{p(1)}\mathcal{F}_{gs} \\ &\neq u^W_{s(2)}\mathcal{F}^*_{gs}\bar{u}^W_{p(2)}\mathcal{F}_{gs}u^W_{s(1)}\mathcal{F}^*_{gs}u^W_{p(1)}\mathcal{F}_{gs}. \end{aligned} \tag{2.37}$$

To illustrate the difference between split-step wave-field modelling and WKBJ+SS wave-field modelling caused by the inequality in equation 2.37, we consider, again, the Marmousi model. Figure 2.6a plots the Marmousi model, and Figures 2.6b-d plot a near-offset section computed using various forward modelling techniques. In particular, Figure 2.6b is the result of finite difference modelling, Figure 2.6c is the result of split-step modelling, and Figure 2.6d is the result of WKBJ+SS modelling. We note that the split-step modelling result provides a closer match to the finite difference data when compared to the WKBJ+SS result. In particular, spurious energy that is present in the WKBJ+SS result is not present in the split-step result. This is especially noticeable in the events that fall at and beyond 2 seconds of travel time.

2.6 Adjoint operator for regularized migration

In the preceding sections, we reviewed various methods for modelling a scalar wave-field, all providing some linear mapping between scattering potential and the measured wave-field. All of these algorithms stem from the Born approximation (equation 2.4). The adjoint of equation 2.4 is,

$$\begin{aligned} &\alpha^\dagger(\mathbf{x}',z') \\ &= \iiint_{-\infty}^{\infty} f^*(\omega)G_0^*(\mathbf{x}_g,z_g|\mathbf{x}',z';\omega)\frac{\omega^2}{c_0^2(\mathbf{x}',z')}\psi(\mathbf{x}_g,z_g|\mathbf{x}_s,z_s;\omega)G_0^*(\mathbf{x}',z'|\mathbf{x}_s,z_s;\omega)d\mathbf{x}_g d\mathbf{x}_s d\omega. \end{aligned} \tag{2.38}$$

In geophysics, this equation (in various guises) has a well-known and extensively analyzed interpretation. Specifically that it is the downward continuation of the wave-field from the

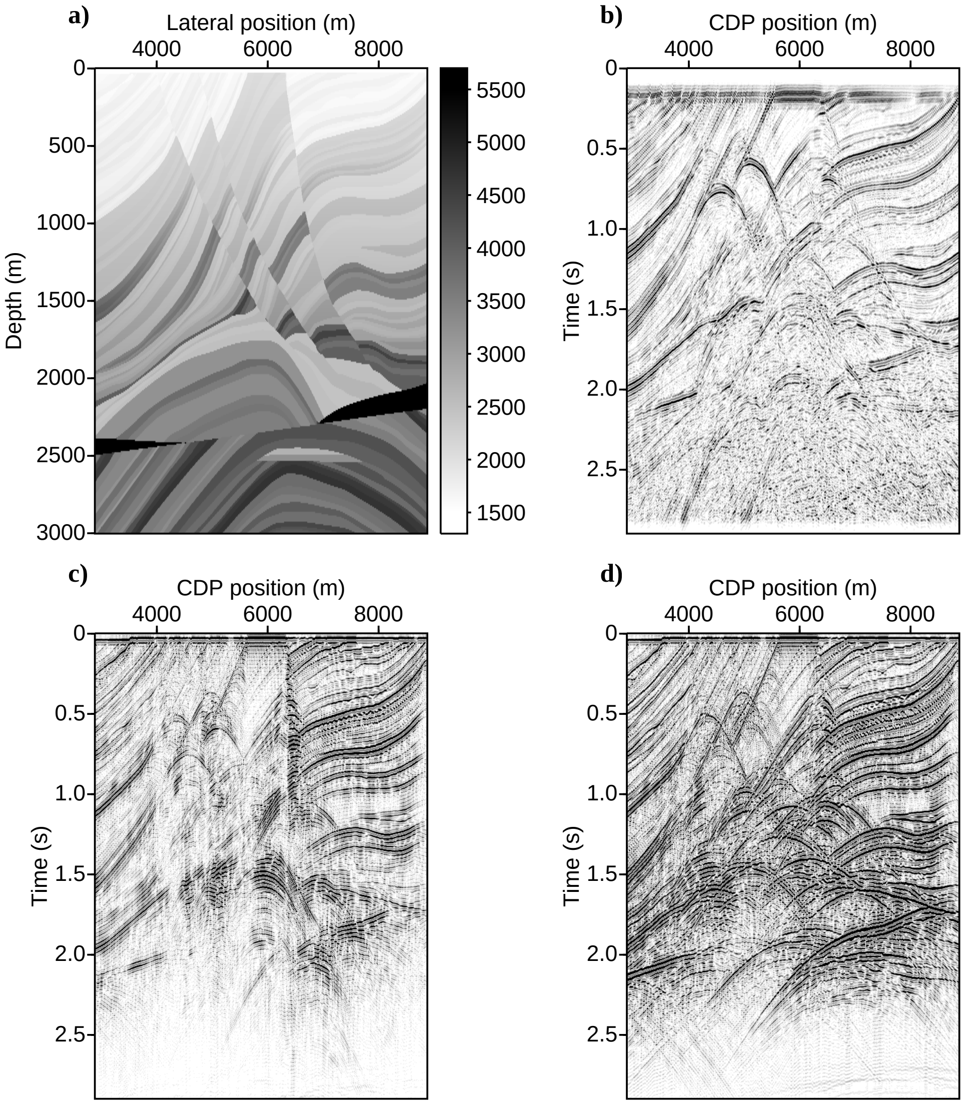

Figure 2.6: The Marmousi velocity model, and b)-d) common offset data sections of the wave-field. The data are computed using b) finite difference, c) split-step modelling and d) WKBJ+SS modelling.

measurement surface to its scattering points. When the wave-field is coincident with the scattering point, time is interpreted to be zero. Indeed it is this interpretation (the so-called imaging condition attributed to Claerbout (1971)) that is most often used to build and interpret migration algorithms, and implies that the adjoint resembles the scattering potential. That is, *the adjoint migrates data.* This interpretation will have further consequence in Chapter 4. For now, we will consider the adjoint of split-step wave-field modelling (equations 2.10 and 2.20). We remind ourselves that with the construction of the adjoint operator, in addition to the forward operator, we have the requisite operators for studying regularized migration.

To find the adjoint of equation 2.20, we recognize its equivalence to a discrete form of the Fredholm integral equation of the first kind which has the general form (e.g. Hansen, 1998),

$$g(x) = \int u(x,z)h(z)dz, \tag{2.39}$$

and the adjoint,

$$h^{\dagger}(z) = \int u^{*}(x,z)g(z)dx, \tag{2.40}$$

where u, g and h are arbitrary functions, and u^* is the adjoint of u. Meanwhile, in equations 2.10 and 2.20, we recognize the form,

$$\psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega_j) = \sum_l u(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_l)\alpha(\mathbf{x}_g, \mathbf{x}_s; z_l), \tag{2.41}$$

so that by analogy to equation 2.40, the adjoint is,

$$\alpha^{\dagger}(\mathbf{x}_g, \mathbf{x}_s; z_l) = \sum_j u^{*}(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_l)\psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega_j), \tag{2.42}$$

where,

$$\begin{aligned} &u^{*}(\mathbf{x}_g, \mathbf{x}_s; \omega, z_l) \\ &\quad = (2\pi)^{-4l} f^{*}(\omega)(u^{*}_{s(l)}\mathcal{F}^{*}_{gs}(\omega/c_{1(l)})^2 u^{*}_{p(l)}\mathcal{F}_{gs}) \cdots (u^{*}_{s(2)}\mathcal{F}^{*}_{gs}u^{*}_{p(2)}\mathcal{F}_{gs})(u^{*}_{s(1)}\mathcal{F}^{*}_{gs}u^{*}_{p(1)}\mathcal{F}_{gs}), \end{aligned} \tag{2.43}$$

and we have used,

$$(u_{s(l)}\mathcal{F}^{*}_{gs}u_{p(l)}\mathcal{F}_{gs})^{*} = (u^{*}_{s(l)}\mathcal{F}^{*}_{gs}u^{*}_{p(l)}\mathcal{F}_{gs}),$$

where $u^{*}_{s(l)}$ and $u^{*}_{p(l)}$ denote the conjugate of $u_{s(l)}$ and $u_{p(l)}$ respectively. In constructing equation 2.43, we have limited the support of α in D_l to $z = z_l$. A similar derivation can be made for the adjoint of Gazdag data modelling, but is not shown here.

We find an efficient numerical implementation of the adjoint operator in equations 2.42 and

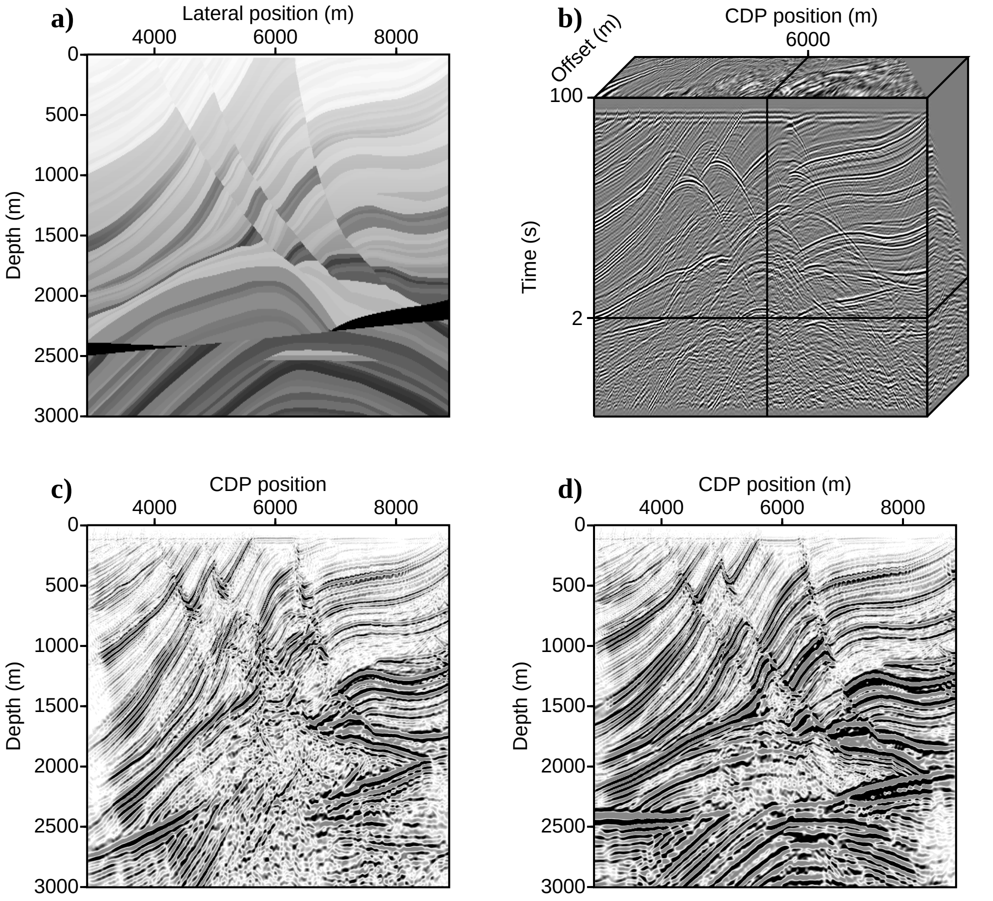

Figure 2.7: a) The Marmousi velocity model, b) The Marmousi data computed using the finite difference approximation to the wave-equation, and the migration results $\alpha^{\dagger}$ using c) the Gazdag adjoint operator and d) the split-step adjoint operator, both applied to the finite difference data in b).

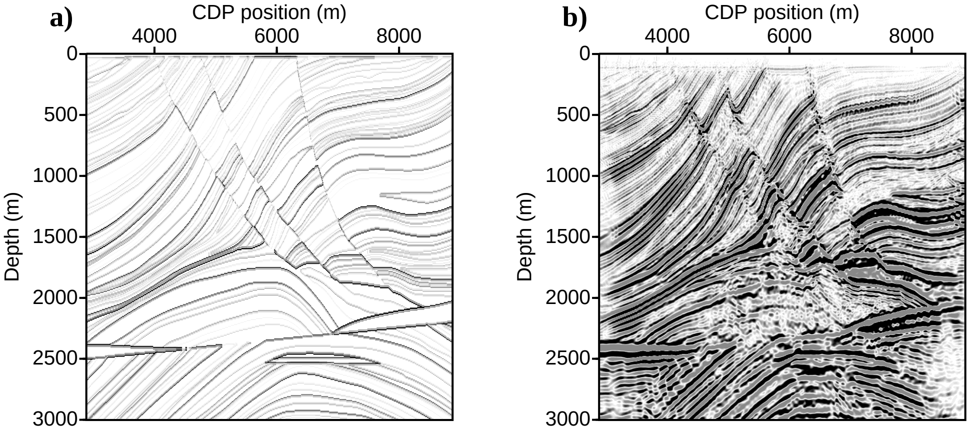

Figure 2.8: We plot a) the true scattering potential α for the Marmousi model, and b) the migration result $\alpha^\dagger$ using the split-step adjoint operator.

2.43 by recognizing an iteration for $u^*(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_l)$. Namely,

$$\begin{aligned}\bar{u}^*(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_l) &= u^*_{s(l)}\mathcal{F}^*_{gs}u^*_{p(l)}\mathcal{F}_{gs}\bar{u}^*(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_{l-1}) \quad , \quad l = 1 \dots n_z \\ u^*(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_l) &= (2\pi)^{-4l} f^*(\omega_j)\bar{u}^*(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_l),\end{aligned} \tag{2.44}$$

and where $\bar{u}^*(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_0) = \psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega_j)$, the measured wave-field. To illustrate the adjoint operator, we continue with the Marmousi example presented in the previous section. In Figure 2.7a, we again plot the Marmousi velocity model. In Figures 2.7c and 2.7d, we plot the results of, respectively, the Gazdag and split-step adjoint operators when applied to the Marmousi finite difference data in Figure 2.7b (previously shown in Figure 2.4b for one realization of offset). In both cases, we plot results for all CDP positions x_m, where at each CDP position, we have integrated $\alpha^\dagger$ over the half-offset dimension x_h, defined as $x_h = (x_g + x_s)/2$. We note that the split-step adjoint provides a reasonable approximation to the discontinuities in the original velocity model, particularly when compared to the result of the Gazdag adjoint operator in Figure 2.7c. To further illustrate the migration result, we plot α computed from equation 2.5 and where the reference velocity c_0 is a smooth version of the velocity c. In particular, Figure 2.8a plots α, and for comparison, Figure 2.8b plots the migration result $\alpha^\dagger$. Notice that the migration result looks like a low resolution version of α.

2.7 A note on model and data dimensions for source-receiver migration

In the preceding analysis, we manipulated the lateral coordinates of the scattering potential $\alpha(\mathbf{x}, z)$. First, we Fourier transformed the Gazdag wave-field modelling operator over lateral geophone and shot coordinates mapping $\alpha(\mathbf{x}, z)$ to $\alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z)$. Subsequently, we added a degree of freedom to α by making $\mathbf{k}_{gx}$ and $\mathbf{k}_{sx}$ independent variables (see equation 2.19), and recognized $\alpha(\mathbf{x}_g, \mathbf{x}_s, z)$ as its four dimensional inverse Fourier transform over lateral coordinates. This is the source-receiver parameterization of the scattering potential, whereby the migrated data (scattering potential) takes on the same lateral coordinates as the data (measured wave-field). Within the regime of the source-receiver parameterization, we, in this section, review an alternate coordinate systems useful for the reflection seismology experiment. Namely, CDP (also known as *midpoint*) $\mathbf{x}_m$ and offset $\mathbf{x}_h$. This is especially useful in the marine acquisition geometry.

In equations 2.41 and 2.42 the forward and adjoint operators for split-step wave-field modelling and migration are written in lateral shot $\mathbf{x}_s$ and receiver $\mathbf{x}_g$ coordinates. We make a change of variables to midpoint $\mathbf{x}_m$ and offset $\mathbf{x}_h$ coordinates (via a 45° rotation). Figure 2.5 illustrates the well-known relation between these two coordinate systems. Specifically, we define (e.g. Clayton and Stolt, 1981),

$$\mathbf{x}_h = (\mathbf{x}_g - \mathbf{x}_s)/2 \qquad \qquad \mathbf{x}_m = (\mathbf{x}_g + \mathbf{x}_s)/2,$$

which we solve for $\mathbf{x}_g$ and $\mathbf{x}_s$, to find,

$$\mathbf{x}_g = \mathbf{x}_m + \mathbf{x}_h \qquad \qquad \mathbf{x}_s = \mathbf{x}_m - \mathbf{x}_h. \tag{2.45}$$

We use equation 2.45 in the Fourier transform of the scattered wave-field at the measurement surface,

$$\begin{aligned}
\psi_s(\mathbf{k}_{gx}, & z_g|\mathbf{k}_{sx}, z_s; t) \\
&= \iint_{-\infty}^{\infty} \psi_s(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; t) e^{i(\mathbf{k}_{sx}\cdot\mathbf{x}_s - \mathbf{k}_{gx}\cdot\mathbf{x}_g)} d\mathbf{x}_g d\mathbf{x}_s \\
&= \iint_{-\infty}^{\infty} \psi_s(\mathbf{x}_m, z_g|\mathbf{x}_h, z_s; t) e^{-i((\mathbf{k}_{gx}-\mathbf{k}_{sx})\cdot\mathbf{x}_m + (\mathbf{k}_{gx}+\mathbf{k}_{sx})\cdot\mathbf{x}_h)} d\mathbf{x}_m d\mathbf{x}_h \qquad (2.46) \\
&= \psi_s(\mathbf{k}_m, z_g|\mathbf{k}_h, z_s; t), \qquad (2.47)
\end{aligned}$$

where we have used, respectively, $e^{-i\mathbf{k}_g\cdot\mathbf{x}_g}$, $e^{i\mathbf{k}_s\cdot\mathbf{x}_s}$, $e^{-i\mathbf{k}_m\cdot\mathbf{x}_m}$ and $e^{-i\mathbf{k}_h\cdot\mathbf{x}_h}$ kernels for the Fourier transforms over $\mathbf{x}_g$, $\mathbf{x}_s$, $\mathbf{x}_m$ and $\mathbf{x}_h$. In equation 2.46 we rotated the frame of reference for the data by 45° (in practise accomplished by a simple re-sorting of data into common midpoint gathers). In equation 2.47, we defined,

$$\mathbf{k}_h = \mathbf{k}_{gx} + \mathbf{k}_{sx} \qquad\qquad \mathbf{k}_m = \mathbf{k}_{gx} - \mathbf{k}_{sx},$$

which, in-turn, leads to,

$$\mathbf{k}_{gx} = (\mathbf{k}_h + \mathbf{k}_m)/2 \qquad\qquad \mathbf{k}_{sx} = (\mathbf{k}_h - \mathbf{k}_m)/2. \tag{2.48}$$

With the change of coordinates from shot and geophone to midpoint and offset, the forward and adjoint operators for wave-field modelling remain largely unchanged. None-the-less, it is worthwhile to write the operators for split-step wave-field modelling using the midpoint-offset parameterization. First, we re-write the l^{th} term (equation 2.20) in the wave-field modelling operator (equation 2.10) in the midpoint-offset coordinate system so that,

$$\begin{aligned}\psi_{s(l)}(\mathbf{x}_m, \mathbf{x}_h; z_0, \omega) = (2\pi)^l f(\omega) \left(u_{s(1)}\mathcal{F}^*_{mh} u_{p(1)}\mathcal{F}_{mh}\right) \cdots \left(u_{s(l-1)}\mathcal{F}^*_{mh} u_{p(l-1)}\mathcal{F}_{mh}\right) \\ \times \int_{z_{l-1}}^{z_l} \left(u_{s(l)}\mathcal{F}^*_{mh} u_{p(l)}\mathcal{F}_{mh}\right) \frac{\omega^2}{c^2_{1(l)}} \alpha(\mathbf{x}_m, \mathbf{x}_h, z')dz',\end{aligned} \tag{2.49}$$

where $\mathcal{F}_{mh}$ is the four dimensional Fourier transform over the midpoint and offset dimensions. The phase shift operator remains similar to its shot-geophone analogue in equation 2.12, but is a function of midpoint and offset rather than shot and geophone so that,

$$u_{p(l)}(\mathbf{k}_m, \mathbf{k}_h, z', \omega) = -\frac{e^{i(k_{gz(l)}+k_{sz(l)})(z'-z_{l-1})}}{16 k_{gz(l)} k_{sz(l)}}, \tag{2.50}$$

where the vertical wave-numbers ($k_{gz(l)}$,$k_{sz(l)}$) are found by substituting equation 2.48 into the dispersion relations (equations 2.13 and 2.14), giving the dispersion relations in midpoint-offset coordinates,

$$k_{gz(l)} = sgn(\omega)\sqrt{\left(\frac{\omega}{c_{0(l)}}\right)^2 - \frac{|\mathbf{k}_h + \mathbf{k}_m|^2}{4}} \tag{2.51}$$

$$k_{sz(l)} = sgn(\omega)\sqrt{\left(\frac{\omega}{c_{0(l)}}\right)^2 - \frac{|\mathbf{k}_h - \mathbf{k}_m|^2}{4}}. \tag{2.52}$$

Likewise, the split-step operator (equation 2.21) becomes,

$$u_{s(l)}(\mathbf{x}_m, \mathbf{x}_h, z', \omega) = e^{i\omega(c^{-1}_{0(l)}(\mathbf{x}_h+\mathbf{x}_m)+c^{-1}_{0(l)}(\mathbf{x}_h-\mathbf{x}_m)-2c^{-1}_{1(l)})(z'-z_{l-1})}. \tag{2.53}$$

Then, in midpoint-offset coordinates, split-step wave-field modelling is given by equations 2.10, and 2.49-2.53. Similarly, we find the adjoint operator (equation 2.42) in midpoint-offset coordinates so that,

$$\alpha^\dagger(\mathbf{x}_m, \mathbf{x}_h; z_l) = \sum_j u^*(\mathbf{x}_m, \mathbf{x}_h; \omega_j, z_l)\psi(\mathbf{x}_m, \mathbf{x}_h; z_0, \omega_j), \tag{2.54}$$

where u^* is given by the trivial 45° rotation of equation 2.43 to midpoint-offset coordinates.

The synthetic Marmousi data is typical of a 2D marine acquisition geometry (towed-streamer) for reflection seismic data, and gives an example where the 45° rotation to midpoint-offset coordinates is beneficial. To illustrate, we plot the acquisition geometry of the Marmousi data in shot-geophone and midpoint-offset coordinates in Figure 2.9. To accommodate the two dimensional fast Fourier transform (integral to the efficiency of the described algorithms), we require ψ and α on a regular grid, either in shot and geophone coordinates, or in midpoint and offset coordinates. To that end, we fill beyond the recording aperture (the white-space in Figures 2.9a and 2.9b) with null data (zero traces). It is evident from Figure 2.9 that for the Marmousi data the midpoint-offset geometry requires fewer zero traces. We note that this may not be true for all acquisition geometries. For example an ocean bottom survey (e.g. Sollid and Ursin, 2003) may show preference to the shot-geophone coordinate system.

2.8 Summary

In this chapter, we constructed forward and adjoint operators for, respectively, wave-field modelling and migration operators using split-step wave-field propagation. In order to derive the operators, we began with constant velocity Green's functions applicable to wave-field propagation with a constant reference velocity. Next, to allow for the velocity model to vary in depth, we employed a Gazdag algorithm, partitioning the reference velocity model into layers. Within each layer the reference velocity was held constant and the scattered wave-field was computed. To propagate the wave-field to and from the layer of interest, we used the direct wave-field. Finally, to allow for the reference velocity to vary in its lateral dimensions, we used the split-step approximation for the Green's function within each layer.

In the case where the reference velocity varies in depth only, the Gazdag wave-propagation algorithm is roughly equivalent to using WKBJ Green's functions. However, we found that when the split-step approximation is used, the Gazdag and WKBJ derivations lead to significantly different operators. Moreover, we found by example that the WKBJ+SS method is less accurate than split-step wave-field modelling.

The split-step wave-field modelling operator derived in this chapter is linear, and in it we recognized a Fredholm integral equation of the first kind. In turn, this made it a relatively

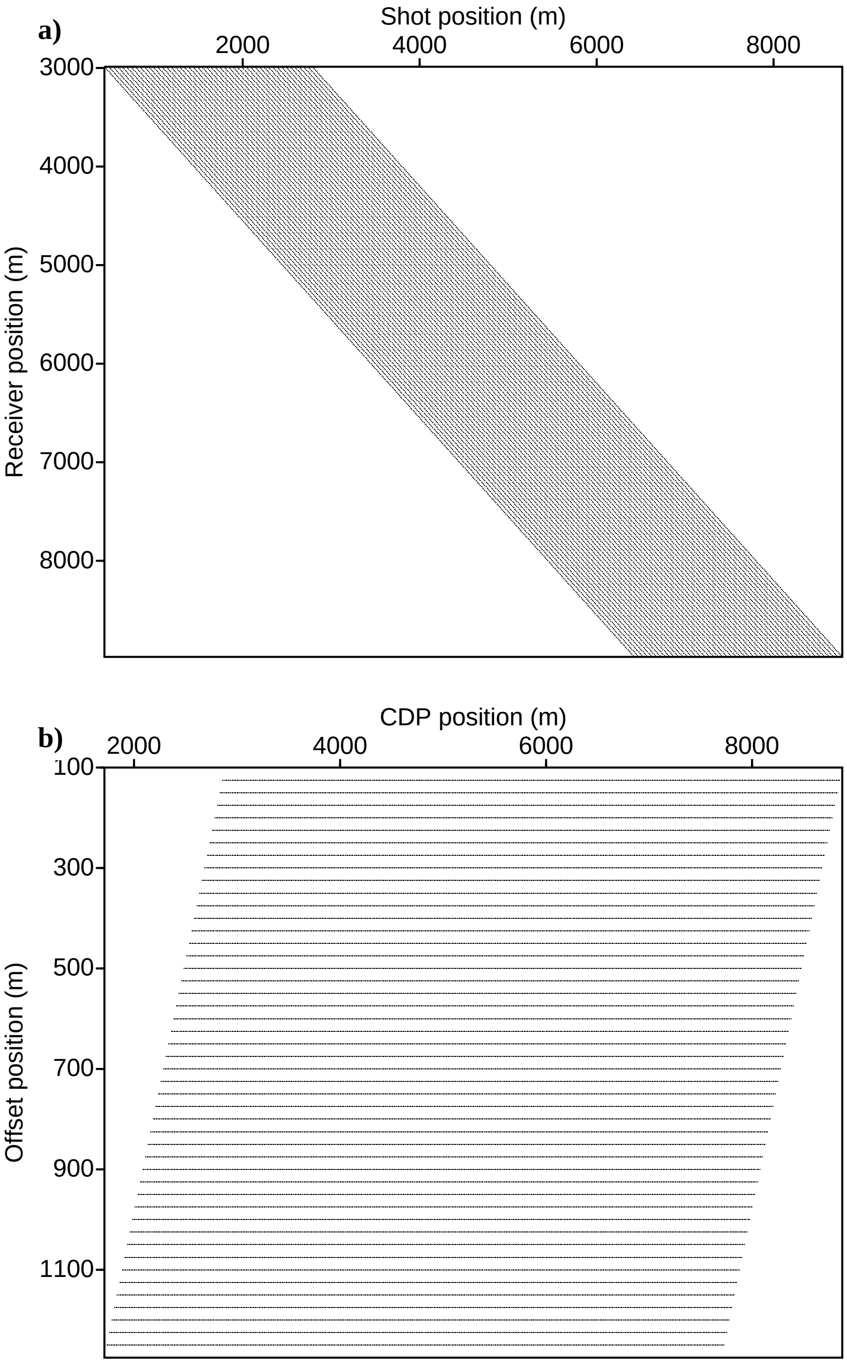

Figure 2.9: We plot the acquisition geometry for the Marmousi data in a) shot-geophone and b) midpoint-offset geometries.

simple matter to find the adjoint to split-step wave-field modelling. The adjoint is called migration, and gives an estimate of the scattering potential. This, in turn, can be interpreted as a map of the earth's velocity structure.

Lastly, we made a 45° rotation in the survey coordinate system, from shot and geophone coordinates to midpoint and offset coordinates. The seismic experiment is most readily parameterized by the positions of its shots and geophones. However, we noted that due to the realities of seismic experiments, it is often more efficient to represent the data in midpoint and offset coordinates.

■ CHAPTER 3

Pre-conditioning of least-squares pre-stack split-step migration and analysis using eigenvalues and Ritz values

3.1 Introduction

In Chapter 2, we used the Born approximation to the acoustic wave-field to develop wave-field modelling and migration operators. The Born approximation is linear, and, as such, the resulting operators can be written using matrix-vector products, $\mathbf{Am} = \mathbf{d}$. In this equation, $\mathbf{d}$ is data (the recording seismic wave-field ψ_s), and we aim to find the scattering potential α housed in the vector $\mathbf{m}$. In Chapter 2, image gathers were found using the migration operator (i.e. the adjoint of the wave-field modelling operator). In this Chapter, we let the image gathers be some model $\mathbf{m}$ that satisfies $\mathbf{Am} = \mathbf{d}$. This can be done in a least-squares sense so that the equation of interest is $\mathbf{A}^H\mathbf{Am} = \mathbf{A}^H\mathbf{d}$, leading to a least-squares migration algorithm (e.g. Nemeth et al., 1999) where the migrated image gathers are found by the least squares inverse rather than by the adjoint.

Finding $\mathbf{m}$ in least-squares migration, then, is largely dependent on the properties of $\mathbf{A}^H\mathbf{A}$. In this chapter, we review the imaging Jacobian $\mathbf{J}$ and its approximate inverse $\mathbf{J}^\dagger$ introduced in Sava et al. (2001) and Sava and Fomel (2003). The Jacobian matrix provides an approximation to $\mathbf{A}^H\mathbf{A}$, so that $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}$ is, in a broad sense, better behaved than $\mathbf{A}^H\mathbf{A}$. In the ideal scenario $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}$ would be the identity matrix; however, the more likely case is that $\mathbf{J}^\dagger$ will work as a pre-conditioner, clustering the eigenvalues of $\mathbf{A}^H\mathbf{A}$, and improving the convergence of iterative solvers. We attempt to quantify this statement by providing an analysis of $\mathbf{A}^H\mathbf{A}$ and $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}$ using, first their eigenvalues, and then their Ritz values, and for the particular case where $\mathbf{A}$ is built using the pre-stack split-step migration operator described in Chapter 2, but where the image gathers are parameterized by ray-parameter rather than by offset. We show that, unfortunately, the Jacobian matrix does not improve the clustering of eigenvalues (or Ritz values) of the least-squares system. The difficulty in the analysis of $\mathbf{A}^H\mathbf{A}$ and $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}$ lie in their size. For example, a subset of the relatively small Marmousi data make $\mathbf{A}^H\mathbf{A}$ a matrix with, roughly, 45 million rows and columns (requiring, in single precision, about 8 Peta-bytes of computer memory). In short, the size of the matrix makes its eigenvalues difficult to compute. Instead, we compute the leading principal minors of similar tridiagonal matrices that are, in turn, found using the conjugate gradient method (Hestenes and Stiefel, 1952). The eigenvalues of these similar matrices are Ritz values of the original matrix, and are shown to be a subset of its eigenvalues.

We begin by re-writing the wave-field modelling (forward) and migration (adjoint) operators for when the image gathers are in midpoint and ray-parameter coordinates rather than midpoint and offset coordinates using the slant-stack transform (Ottolini and Claerbout, 1984). These are the coordinates in which the Jacobian matrix introduced by Sava et al. (2001) is applied. Next, we write down the discrete matrix-vector form of the forward and adjoint operators, and the corresponding set of least-squares equations. We construct the solution to the least-squares equations using the conjugate gradient method which we introduce in this chapter, but which will also be useful in later chapters. With the conjugate gradient method at hand, we review the concept of similar matrices and use them to compute Ritz values from the conjugate gradient method (e.g. Scales, 1989). Next, we derive the Jacobian matrix, using it as a pre-conditioner for a synthetic example with a small acquisition geometry, making an analysis of the eigenvalues of the least-squares equations without and with pre-conditioning. Finally, we review a derivation of the pre-conditioned conjugate gradient method, and use it to compute the Ritz values of the pre-conditioned least-squares equations for a typical acquisition geometry using the synthetic Marmousi data.

3.2 Forward and adjoint operators in midpoint and ray-parameter coordinates

In Chapter 2, we derived forward (wave-field modelling) and adjoint (migration) operators for a source-receiver geometry. In our derivation, the scattering potential and wave-field were parameterized by either shot and receiver coordinates or, alternatively, by midpoint and offset coordinates. In this chapter, we parameterize the scattering potential using midpoint and ray-parameter coordinates, and we, as before, parameterize the wave-field using midpoint and offset coordinates.

The change in coordinates for scattering potential from midpoint and offset to midpoint and ray-parameter is more involved then the simple 45° coordinate rotation used to go from shot and receiver coordinates to midpoint and offset coordinates. In particular, the relation between offset ray-parameter $\mathbf{p}_h = (p_{hx}, p_{hy})$ and lateral wave-numbers ($\mathbf{k}_{gx}$, $\mathbf{k}_{sx}$ and $\mathbf{k}_h$)

is,

$$\mathbf{p}_h = \omega(\mathbf{k}_{gx} + \mathbf{k}_{sx}) = \omega \mathbf{k}_h = \frac{1}{c_1}(\sin\theta_g + \sin\theta_s), \tag{3.1}$$

where θ_g and θ_s were described in Figure 2.2. The transformation from midpoint and offset to midpoint and ray-parameter given by equation 3.1 is implemented by the slant-stack transform (e.g. Ottolini and Claerbout, 1984). For example, in split-step wave-field modelling, equation 2.49 becomes,

$$\begin{aligned} \psi_{s(k)}(\mathbf{x}_m, \mathbf{x}_h; z_0, \omega) &= \mathcal{G}(\mathbf{x}_m, \mathbf{x}_h, \mathbf{p}_h; z', \omega)\alpha(\mathbf{x}_m, \mathbf{p}_h, z') \\ &= (2\pi)^l f(\omega) \left(u_{s(1)}\mathcal{F}^*_{mh} u_{p(1)} \mathcal{F}_{mh}\right) \cdots \left(u_{s(l-1)}\mathcal{F}^*_{mh} u_{p(l-1)}\mathcal{F}_{mh}\right) \\ &\times \int_{z_{l-1}}^{z_l} \left(u_{s(l)}\mathcal{F}^*_{mh} u_{p(l)}\mathcal{F}_{mh}\right) \frac{\omega^2}{c^2_{1(l)}} \mathcal{F}^*_h \int_{-\infty}^{\infty} \delta(\mathbf{p}_h - \omega\mathbf{k}_h)\alpha(\mathbf{x}_m, \mathbf{p}_h, z') d\mathbf{p}_h dz', \end{aligned} \tag{3.2}$$

where u_p and u_s are given by, respectively, equations 2.50 and 2.53, and $\mathcal{F}_h$ is the two dimensional Fourier transform over offset coordinates $\mathbf{x}_h$. As before (Chapter 2), $\mathcal{F}_{mh}$ is the four dimensional Fourier transform over midpoint and offset coordinates. Likewise, the adjoint operator in equation 2.54 becomes (with continuous frequency ω),

$$\begin{aligned} \alpha^\dagger(\mathbf{x}_m, \mathbf{p}_h; z) &= \mathcal{G}^\dagger(\mathbf{x}_m, \mathbf{x}_h, \mathbf{p}_h; z, \omega)\psi_s(\mathbf{x}_m, \mathbf{x}_h; z_0, \omega) \\ &= \iint_{-\infty}^{\infty} \delta(\mathbf{k}_h - \mathbf{p}_h/\omega)\mathcal{F}_h u^*(\mathbf{x}_m, \mathbf{x}_h; \omega, z)\psi_s(\mathbf{x}_m, \mathbf{x}_h; z_0, \omega) d\mathbf{p}_h d\omega, \end{aligned} \tag{3.3}$$

where u^* is shown by equations 2.43 and 2.54. The operators $\mathcal{G}$ and $\mathcal{G}^\dagger$ represent, respectively, the mappings from α to ψ_s, and ψ_s to $\alpha^\dagger$, and will be used in the ensuing analysis. Figures 3.1a and 3.1b illustrates the adjoint operator applied to the Marmousi data in, respectively, midpoint and offset, and midpoint and ray-parameter coordinates. The latter is used in amplitude versus ray-parameter studies (e.g. Kuehl, 2002) which in turn can be used to invert for earth properties.

3.3 Least-squares solution for the scattering potential

In this chapter, we construct the least-squares solution for the scattering potential (i.e. least-squares migration) given the forward and adjoint operators ($\mathcal{G}$ and $\mathcal{G}^\dagger$) in equations 3.2 and 3.3. That is, we choose the scattering potential that gives the smallest distance between observed and modelled data (i.e. measured wave-field). In Section 3.4 we note that more than one instance of the scattering potential may satisfy this criterion. Since we are, ultimately, interested in a numerical implementation, we re-cast equations 3.2 and 3.3 in their discrete matrix-vector form. To that end, we introduce vectors $\mathbf{d}$ and $\mathbf{m}$ such that the entries of $\mathbf{d}$ are realized from the observed wave-field $\psi_s(\mathbf{x}_m, \mathbf{x}_h; \omega)$, and the entries of $\mathbf{m}$ are realized

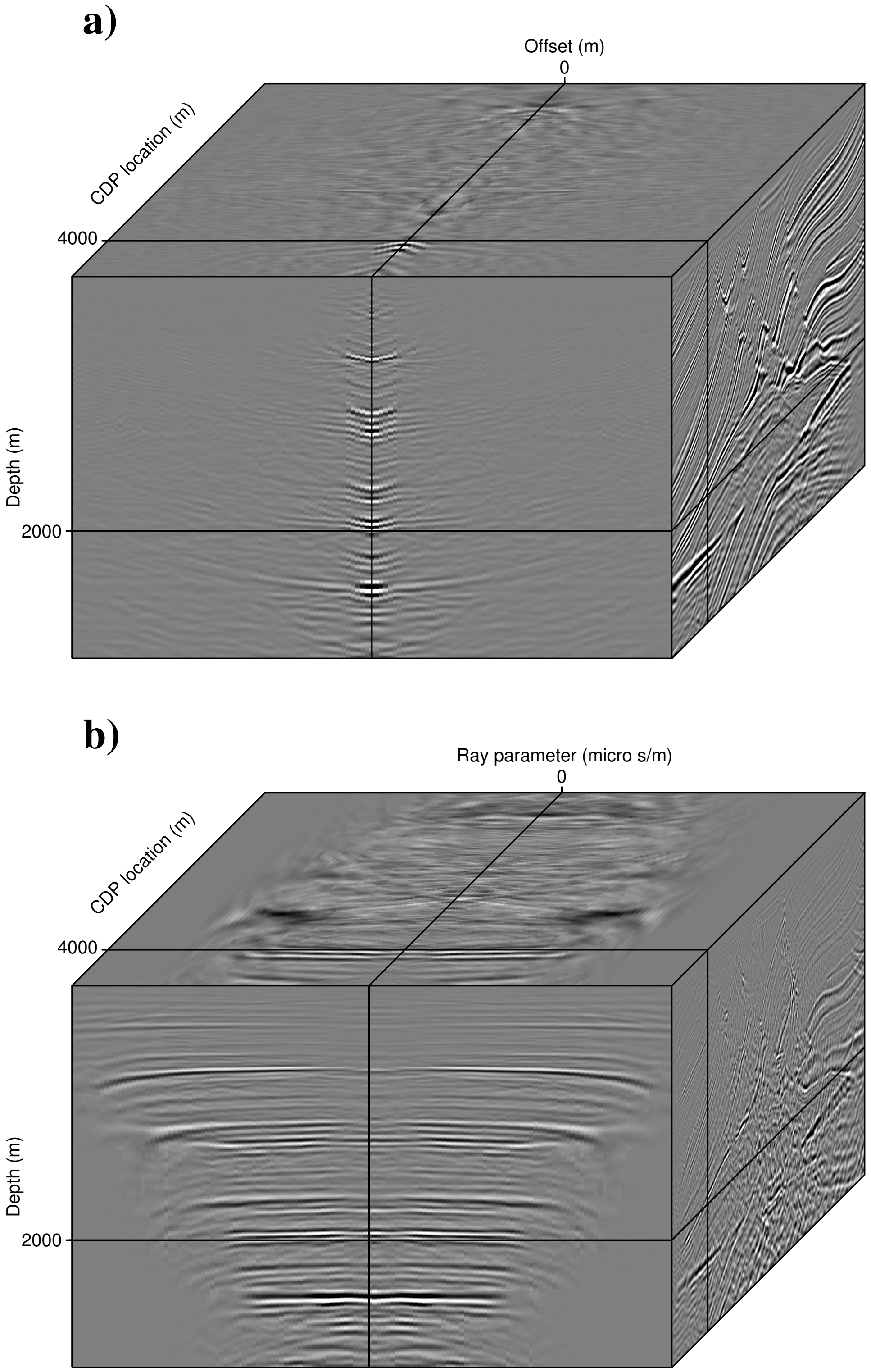

Figure 3.1: We show the result of applying the adjoint operator to the Marmousi data, using a) midpoint and offset coordinates, and b) midpoint and ray-parameter coordinates.

from the scattering potential $\alpha(\mathbf{x}_m, \mathbf{p}_h, z)$. This, in turn, allows us to write the least-squares solution in a discrete vector space.

We let the model vector $\mathbf{m}$ be of length N, and the data vector $\mathbf{d}$ be of length M so that,

$$\mathbf{m}^T = \left[\begin{array}{cccc} m_1 & m_2 & \cdots & m_N \end{array}\right] \qquad \mathbf{d}^T = \left[\begin{array}{cccc} d_1 & d_2 & \cdots & d_M \end{array}\right]. \tag{3.4}$$

Further, we define realizations of ψ_s and α, so that,

$$\begin{aligned} m_i &= \alpha(x_{m(l_2)}, y_{m(l_3)}, p_{hx(l_4)}, p_{hy(l_5)}; z_{(l_1)}) \quad , \quad i = l_5 + L_5(l_4 + L_4(l_3 + L_3(l_2 + L_2 l_1))) \\ d_j &= \psi_s(x_{m(l_2)}, y_{m(l_3)}, x_{h(l_7)}, y_{h(l_8)}; \omega_{(l_6)}) \quad , \quad j = l_8 + L_8(l_7 + L_7(l_3 + L_3(l_2 + L_2 l_6))), \end{aligned} \tag{3.5}$$

where $l_k = 1 \ldots L_k$, $k = 1 \ldots 8$, so that the latent variables are realized on the discrete grid given by,

$$\begin{aligned} z_{(l_1)} &= z_0 + l_1 \Delta z &&, \quad l_1 = 0 \ldots L_1 \\ \omega_{(l_2)} &= \omega_0 + l_2 \Delta\omega &&, \quad l_2 = 0 \ldots L_2 \\ x_{m(l_3)} &= (x_m)_0 + l_3 \Delta x_m &&, \quad l_3 = 0 \ldots L_3 \\ y_{m(l_4)} &= (y_m)_0 + l_4 \Delta y_m &&, \quad l_4 = 0 \ldots L_4 \\ x_{h(l_5)} &= (x_h)_0 + l_5 \Delta x_h &&, \quad l_5 = 0 \ldots L_5 \\ y_{h(l_6)} &= (y_h)_0 + l_6 \Delta y_h &&, \quad l_6 = 0 \ldots L_6 \\ p_{hx(l_7)} &= (p_{hx})_0 + l_7 \Delta p_{hx} &&, \quad l_7 = \ldots L_7 \\ p_{hy(l_8)} &= (p_{hy})_0 + l_8 \Delta p_{hy} &&, \quad l_8 = \ldots L_8. \end{aligned} \tag{3.6}$$

The operators $\mathcal{G}$ and $\mathcal{G}^\dagger$ in equations 3.2 and 3.3 are mapped into matrices, $\mathbf{A}$ and $\mathbf{A}^H$, respectively so that the i^{th} row and j^{th} column of $\mathbf{A}$ is given by,

$$[\mathbf{A}]_{ij} = \mathcal{G}(x_{m(l_3)}, y_{m(l_4)}, x_{h(l_5)}, y_{h(l_6)}, p_{hx(l_7)}, p_{hy(l_8)}; z_{(l_1)}, \omega_{(l_2)}), \tag{3.7}$$

where,

$$\begin{aligned} i &= l_6 + L_6(l_5 + L_5(l_4 + L_4(l_3 + L_3 l_2))) \\ j &= l_8 + L_8(l_7 + L_7(l_4 + L_4(l_3 + L_3 l_1))). \end{aligned} \tag{3.8}$$

Given the definitions for $\mathbf{A}$, $\mathbf{m}$ and $\mathbf{d}$ in equations 3.4-3.8, we recognize $\mathbf{d} = \mathbf{A}\mathbf{m}$ and $\mathbf{m}^\dagger = \mathbf{A}^H\mathbf{d}$ which are the discretized versions of equations 3.2 and 3.3, respectively. Then we define a cost function,

$$\phi(\mathbf{m}) = ||\mathbf{d} - \mathbf{A}\mathbf{m}||_2^2, \tag{3.9}$$

which has its minimum when,

$$\mathbf{A}^H\mathbf{A}\mathbf{m} = \mathbf{A}^H\mathbf{d}, \tag{3.10}$$

and are the classical least-squares normal equations.

In practise the size of $\mathbf{A}^H\mathbf{A}$ and $\mathbf{A}$ make their explicit construction infeasible on modern day computers. For example, the relatively small subset of the Marmousi data used to illustrate the migration and de-migration operators in Chapter 2 and Figure 3.1 correspond to $\mathbf{A}^H\mathbf{A}$ having roughly 45 million rows and columns. To store this matrix on a computer would require (in single precision) about 16 Peta-bytes of computer memory. Therefore, we rely on iterative methods to minimize the cost $\phi(\mathbf{m})$ in equation 3.9.

3.4 Least-squares conjugate gradients

The iterative method we use to find a solution for equation 3.10 is the least-squares conjugate gradient method (Hestenes and Stiefel, 1952). The conjugate gradient method constructs a solution for $\mathbf{m}$ by its projection onto a growing Krylov space, where each conjugate gradient iteration adds an additional dimension to the space. Our choice of method dictates the solution. Namely, it prescribes a search direction for each iteration, and insists that the error in the model is orthogonal (as defined by the Krylov space) to the search direction.

To proceed, we note that $\mathbf{A}^H\mathbf{A}$ in equation 3.10 is Hermitian (self-adjoint). Further, we define $\mathbf{m}_k$ to be the estimate of $\mathbf{m}$ after the k^{th} conjugate gradient iteration, and $\mathbf{e}_k = \mathbf{m} - \mathbf{m}_k$ be its associated error. Further, we define a residual vector after the k^{th} iteration, $\mathbf{r}_k = \mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{m}_k)$. The conjugate gradient method chooses a set of search directions $\mathbf{p}_k$, $k = 1 \dots n$ that are orthogonal in a Krylov space (i.e. $\mathbf{p}_k\mathbf{A}^H\mathbf{A}\mathbf{p}_{k-1} = 0$). The conjugate gradient method is initialized by letting $\mathbf{m}_0 = \mathbf{0}$, and by letting the initial search direction be the residual vector so that $\mathbf{p}_0 = \mathbf{A}^H\mathbf{d}$. Then, it finds $\mathbf{m}_1$ by the projection of $\mathbf{m}$ onto $\mathbf{p}_0$, and under the constraint $\mathbf{p}_0^H\mathbf{A}^H\mathbf{A}\mathbf{e}_1 = 0$. That is the search direction $\mathbf{p}_0$, and the error $\mathbf{e}_1$ are, in a Krylov space, made orthogonal. Generalizing, the projection onto the first n search directions is found using the constraints,

$$\mathbf{p}_k\mathbf{A}^H\mathbf{A}\mathbf{e}_{k-1} = 0 \quad , \quad k = 1 \dots n-1. \tag{3.11}$$

The choice of search directions in the conjugate gradient method has the pleasing result that each new search direction $\mathbf{p}_k$ can be computed using only the previous search direction $\mathbf{p}_{k-1}$ and the current residual vector $\mathbf{r}_k$. There are two additional consequences of the conjugate gradient method that are important to our analysis. First the residual vectors $\mathbf{r}_k$ are mutually orthogonal in an $L2$ sense (i.e. $\mathbf{r}_k^H\mathbf{r}_{k-1} = 0$), and, second, the search directions form the basis for a Krylov space $\mathcal{K}_n$ which after n iterations is given by,

$$\mathcal{K}_n(\mathbf{d}) = \left\{\mathbf{A}^H\mathbf{d}, (\mathbf{A}^H\mathbf{A})\mathbf{A}^H\mathbf{d}, (\mathbf{A}^H\mathbf{A})^2\mathbf{A}^H\mathbf{d}, \dots, (\mathbf{A}^H\mathbf{A})^{n-1}\mathbf{A}^H\mathbf{d}\right\}. \tag{3.12}$$

Algorithm 3.1 summarizes the conjugate gradient method. In it, α_k is the step-length in the

Algorithm 3.1 Least-squares conjugate gradient method for a matrix $\mathbf{A}$, and data $\mathbf{d}$

```
$\mathbf{m}_0 = \mathbf{0}$, $\mathbf{p}_0 = \mathbf{r}_0 = \mathbf{A}^H\mathbf{d}$
for $k = 1$ to $n$ do
    $\alpha_k = ||\mathbf{r}_{k-1}||_2^2/(\mathbf{p}_{k-1}^H\mathbf{A}^H\mathbf{A}\mathbf{p}_{k-1})$
    $\mathbf{m}_k = \mathbf{m}_{k-1} + \alpha_k\mathbf{p}_{k-1}$
    $\mathbf{r}_k = \mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{m}_k)$
    $\beta_k = ||\mathbf{r}_k||_2^2/||\mathbf{r}_{k-1}||_2^2$
    $\mathbf{p}_k = \mathbf{r}_k + \beta_k\mathbf{p}_{k-1}$
end for
```

$(k-1)^{th}$ search direction $\mathbf{p}_{k-1}$, and is determined by the previously mentioned orthogonality between the search direction and the error vector $\mathbf{e}_k$. Further, β_k helps determine the k^{th} search direction, and is found by Gram-Schmidt conjugation. There are well know redundancies in the equations which are not shown in Algorithm 3.1, and with the consequence that in the least-squares conjugate gradient method $\mathbf{A}$ and $\mathbf{A}^H$ need only be applied once per iteration. To show this, we make the following two observations. First for the residual vector $\mathbf{r}_k$, we can write,

$$\begin{aligned}\mathbf{r}_k = \mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{m}_k) &= \mathbf{A}^H(\mathbf{d} - \mathbf{A}(\mathbf{m}_{k-1} + \alpha_k\mathbf{p}_{k-1})) \\ &= \mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{m}_{k-1}) - \alpha_k\mathbf{A}^H\mathbf{A}\mathbf{p}_{k-1} = \mathbf{r}_{k-1} - \alpha_k\mathbf{A}^H\mathbf{A}\mathbf{p}_{k-1},\end{aligned} \tag{3.13}$$

and, second, $\mathbf{A}^H\mathbf{A}\mathbf{p}_{k-1}$ appears in the computation of the step-size α_k, confirming our assertion that $\mathbf{A}$ and $\mathbf{A}^H$ need only be applied once per iteration. We note, however, that when taking advantage of equation 3.13, we have to be careful of numerical round-off errors caused by its iteration on the residual vector. In practise, every 10 conjugate gradients iterations we use the non-iterative formula for $\mathbf{r}_k$, removing cumulative round-off errors.

In geophysical applications, it is often the case that the least-squares normal equations (3.10) are ill-conditioned so that it is satisfied by more than one realization of $\mathbf{m}$. In this case the solution found by the conjugate gradient method is dictated by its projection onto the constructed Krylov space. In particular, the solution at the k^{th} iteration is constrained by equation 3.11. This is in contrast to constraints that appear in alternative iterative methods such as the *generalized minimal residual method* that does the projection onto the Krylov space using the minimum of the $L2$ norm of the residual vector $\mathbf{r}_k$ (Saad and Schultz, 1986).

The conjugate gradient method is widely used in geophysical applications (e.g. Fomel, 2007; Pidlisecky et al., 2007; Chunduru et al., 1997; Pilkington, 1997). It is guaranteed to converge in N iterations where N is the dimension of the model vector $\mathbf{m}$. Moreover, it does not require the explicit construction of $\mathbf{A}$. This is especially important for the applications considered in this thesis where $\mathbf{A}$ is prohibitively large and cannot be explicitly constructed and stored within the memory of a modern computer. Further, given the size of $\mathbf{A}$ it would be unreasonable to run the conjugate gradient method through its N iterations. However, running the algorithm for fewer iterations allows us to relate the properties of the constructed Krylov space to the spectrum (eigenvalues) of $\mathbf{A}^H\mathbf{A}$. Moreover, this encourages the pre-conditioning of $\mathbf{A}^H\mathbf{A}$, allowing for clustering of eigenvalues and convergence in fewer iterations, and which we examine in Sections 3.6 and 3.7. Prior to discussing pre-conditioning within the context of the conjugate gradient method, it is important to understand the relation between the eigenvalues of $\mathbf{A}^H\mathbf{A}$ and the iterations of the conjugate gradient method, and which we make an effort to describe in the next section.

3.5 Similar matrices and Ritz values

We use Ritz values to characterize the spectral properties of $\mathbf{A}^H\mathbf{A}$, and which we find using the conjugate gradient method. In particular, the conjugate gradient method can be used to construct a similar tri-diagonal matrix which after the n^{th} conjugate gradient iteration will have n rows and columns. Its eigenvalues are readily computed and are called Ritz values of $\mathbf{A}^H\mathbf{A}$, which are, in turn, a subset of its eigenvalues.

We begin with the definition of similar matrices: we let $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{M}$ be square matrices, and $\mathbf{M}$ be invertible. If $\mathbf{C} = \mathbf{M}^{-1}\mathbf{B}\mathbf{M}$, then $\mathbf{B}$ and $\mathbf{C}$ are similar matrices, and share the same eigenvalues. The proof is straight-forward. In particular,

$$\mathbf{B}\mathbf{x} = \lambda\mathbf{x} \Rightarrow \mathbf{M}\mathbf{C}\mathbf{M}^{-1}\mathbf{x} = \lambda\mathbf{x} \Rightarrow \mathbf{C}(\mathbf{M}^{-1}\mathbf{x}) = \lambda(\mathbf{M}^{-1}\mathbf{x}). \tag{3.14}$$

As a corollary, equation 3.14 shows that if $\mathbf{x}$ is an eigenvector of $\mathbf{B}$, then $\mathbf{M}^{-1}\mathbf{x}$ is an eigenvector of $\mathbf{C}$. The corollary is not integral to our discussion. In the ensuing analysis it is important to consider the slightly more complicated case where $\mathbf{M}$ is a matrix with M rows and N columns, and where $M > N$. Moreover, we can assume that the columns of $\mathbf{M}$ are orthonormal, so that $\mathbf{M}^H\mathbf{M} = \mathbf{I}_N$ where $\mathbf{I}_N$ is the identity matrix with N rows and columns. That is $\mathbf{M}^H$ is the left inverse of $\mathbf{M}$. In this case, we write $\mathbf{C} = \mathbf{M}^H\mathbf{B}\mathbf{M}$, and show that $\mathbf{B}$ and $\mathbf{C}$ are similar matrices by the proof,

$$\mathbf{C}\mathbf{x} = \lambda\mathbf{x} \Rightarrow \mathbf{M}^H\mathbf{B}\mathbf{M}\mathbf{x} = \lambda\mathbf{x} \Rightarrow \mathbf{M}^H\mathbf{B}\hat{\mathbf{x}} = \lambda\mathbf{M}^H\hat{\mathbf{x}} \Rightarrow \mathbf{M}^H\left(\mathbf{B} - \lambda\mathbf{I}_N\right)\hat{\mathbf{x}} = \mathbf{0}, \tag{3.15}$$

where we have let $\hat{\mathbf{x}} = \mathbf{M}\mathbf{x}$. Equation 3.15 shows that if λ is an eigenvalue of $\mathbf{B}$, then it is also an eigenvalue of $\mathbf{C}$, in turn, showing that $\mathbf{C}$ and $\mathbf{B}$ are similar.

With the definition of similar matrices in equation 3.15, we return to the normal equations for least-squares migration (3.10). We are interested in characterizing the spectral properties of $\mathbf{A}^H\mathbf{A}$. We use the conjugate gradient method to find similar matrices $\mathbf{A}^H\mathbf{A}$ and $\mathbf{T}_n$, and where $\mathbf{T}_n$ is a tri-diagonal matrix built using the conjugate gradient method. After each conjugate gradient iteration, one row and column can be added to $\mathbf{T}_n$ so after n iterations $\mathbf{T}_n$ has n rows and columns. To proceed, we define matrices $\boldsymbol{\Delta}_n$, $\mathbf{R}_n$, $\mathbf{P}_n$ and $\mathbf{B}_n$ such that,

$$\mathbf{R}_n = \begin{bmatrix} \mathbf{r}_0 & \mathbf{r}_1 & \dots & \mathbf{r}_{n-1} \end{bmatrix} \qquad \mathbf{P}_n = \begin{bmatrix} \mathbf{p}_0 & \mathbf{p}_1 & \dots & \mathbf{p}_{n-1} \end{bmatrix},$$

where $\mathbf{r}_k$ and $\mathbf{p}_k$ for $k = 0 \dots n-1$ are column vectors. In particular $\mathbf{r}_k$ is the residual vector and $\mathbf{p}_k$ is the search direction for the k^{th} least-squares conjugate gradient iteration defined in Section 3.4. The matrix $\boldsymbol{\Delta}_n$ is a square and diagonal matrix, with its k^{th} diagonal element being $||\mathbf{r}_k||_2$, $k = 0 \dots n-1$. We note that $\mathbf{R}_n$ and $\mathbf{P}_n$ each have N rows and n columns, while $\mathbf{B}_n$ and $\boldsymbol{\Delta}_n$ are square matrices with n rows and columns. Given $\mathbf{P}_n$ and $\mathbf{R}_n$, the matrix $\mathbf{B}_n$ is defined implicitly such that,

$$\mathbf{R}_n = \mathbf{P}_n\mathbf{B}_n. \tag{3.16}$$

In equation 3.16, we use the relation from the conjugate gradient method, $\mathbf{p}_k = \mathbf{r}_k + \beta_k\mathbf{p}_{k-1}$ (Algorithm 3.1), so that,

$$\mathbf{B}_n = \begin{bmatrix} 1 & -\beta_1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & -\beta_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -\beta_{n-1} \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}.$$

Further, we define the matrix $\hat{\mathbf{R}}_n = \mathbf{R}_n\boldsymbol{\Delta}_{\mathbf{n}}^{\mathbf{-1}}$ so that the columns of $\hat{\mathbf{R}}_n$ are orthonormal. Since $\hat{\mathbf{R}}_n$ has orthonormal columns, by equation 3.15, $\mathbf{T}_n = \hat{\mathbf{R}}_n^H\mathbf{A}^H\mathbf{A}\hat{\mathbf{R}}_n$ and $\mathbf{A}^H\mathbf{A}$ are similar matrices, and from equation 3.16 we can write,

$$\mathbf{T}_n = \hat{\mathbf{R}}_n^H\mathbf{A}^H\mathbf{A}\hat{\mathbf{R}}_n = \boldsymbol{\Delta}_n^{-1}\mathbf{B}_n^H\mathbf{P}_n^H\mathbf{A}^H\mathbf{A}\mathbf{P}_n\mathbf{B}_n\boldsymbol{\Delta}_n^{-1}. \tag{3.17}$$

Furthermore, by construction the columns of $\mathbf{P}_n$ are orthogonal in the Krylov space shown in equation 3.12. Hence, $\mathbf{P}_n^H\mathbf{A}^H\mathbf{A}\mathbf{P}_n$ must be a diagonal matrix with its k^{th} diagonal element being $\delta_k^p = \mathbf{p}_k^H \mathbf{A}^H \mathbf{A} \mathbf{p}_k$. Further, letting $\delta_k^r = \sqrt{1/(\mathbf{r}_k^H \mathbf{r}_k)}$, we find (after some algebra),

$$\mathbf{T}_n = \begin{bmatrix} \delta_0^p(\delta_0^r)^2 & -\delta_0^p\delta_0^r\delta_1^r\beta_1 & 0 & 0 & \cdots & 0 \\ -\delta_0^p\delta_0^r\delta_1^r\beta_1 & \delta_0^p(\delta_1^r)^2\beta_1^2 + \delta_1^p(\delta_1^r)^2 & -\delta_1^p(\delta_1^r)^2\beta_2 & 0 & \cdots & 0 \\ 0 & \delta_0^p(\delta_1^r)^2\beta_2 & \delta_1^p(\delta_1^r)^2\beta_2^2 + \delta_2^p(\delta_2^r)^2 & -\delta_2^p(\delta_2^r)^2\beta_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}. \tag{3.18}$$

To simplify equation 3.18, we note its relation to the step-lengths α_k and Gram-Schmidt conjugation factors β_k in the conjugate gradient method. Namely, we find,

$$\delta_k^p(\delta_k^r)^2 = \frac{\mathbf{p}_k^H \mathbf{A}^H \mathbf{A} \mathbf{p}_k}{\mathbf{r}_k^H \mathbf{r}_k} = \alpha_k^{-1}, \tag{3.19}$$

$$\delta_k^p \delta_k^r \delta_{k+1}^r = \delta_k^p(\delta_k^r)^2 \delta_{k+1}^r/\delta_k^r = \frac{\mathbf{p}_k^H \mathbf{A}^H \mathbf{A} \mathbf{p}_k}{\mathbf{r}_k^H \mathbf{r}_k} \sqrt{\frac{\mathbf{r}_k^H \mathbf{r}_k}{\mathbf{r}_{k+1}^H \mathbf{r}_{k+1}}} = \alpha_k^{-1} \beta_{k+1}^{-1/2}, \tag{3.20}$$

and,

$$\delta_k^p(\delta_{k+1}^r)^2 = \delta_k^p(\delta_k^r)^2(\delta_{k+1}^r/\delta_k^r)^2 = \frac{\mathbf{p}_k^H \mathbf{A}^H \mathbf{A} \mathbf{p}_k}{\mathbf{r}_k^H \mathbf{r}_k} \frac{\mathbf{r}_k^H \mathbf{r}_k}{\mathbf{r}_{k+1}^H \mathbf{r}_{k+1}} = \alpha_k^{-1} \beta_{k+1}. \tag{3.21}$$

Substituting the results from equations 3.19-3.21 into equation 3.18 gives for $\mathbf{T}_n$,

$$\begin{bmatrix} \alpha_0^{-1} & -\alpha_0^{-1}\beta_1^{1/2} & 0 & 0 & \cdots & 0 \\ -\alpha_0^{-1}\beta_1^{1/2} & \beta_1/\alpha_0 + \alpha_1^{-1} & -\alpha_1^{-1}\beta_2^{1/2} & 0 & \cdots & 0 \\ 0 & -\alpha_2^{-1}\beta_2^{1/2} & \beta_2/\alpha_1 + \alpha_2^{-1} & -\beta_3^{-1/2}\alpha_2^{-1} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\beta_{n-1}/\alpha_{n-2} + \alpha_{n-1}^{-1} \end{bmatrix}. \tag{3.22}$$

We notice that in equation 3.22, $\mathbf{T}_n$ has diagonal elements given by,

$$[\mathbf{T}_n]_{ii} = \beta_k/\alpha_{k+1} + \alpha_k^{-1},$$

for $k = 0 \ldots n-1$ and $\beta_0 = 0$, and off-diagonal elements are given by,

$$[\mathbf{T}_n]_{k,k+1} = [\mathbf{T}_n]_{k+1,i} = -\alpha_k^{-1}\beta_{k+1}^{1/2},$$

for $k = 0 \ldots n-2$. The eigenvalues of $\mathbf{T}_n$ are called the Ritz values of $\mathbf{A}^H\mathbf{A}$. Since $\mathbf{T}_n$ and $\mathbf{A}^H\mathbf{A}$ are similar matrices, it must be that Ritz values of $\mathbf{A}^H\mathbf{A}$ are also a subset of its eigenvalues. Therefore, we choose to use the Ritz values to characterize the spectral properties of $\mathbf{A}^H\mathbf{A}$. The end result of our analysis is that the matrix $\mathbf{T}_n$ can be computed using the step-lengths and Gram-Schmidt coefficients found in the least-squares conjugate gradient method. In turn, the eigenvalues of $\mathbf{T}_n$ are readily computed (due to its tri-diagonal form), and are the Ritz values of $\mathbf{A}^H\mathbf{A}$.

3.6 The Jacobian matrix

The least-squares solution using the conjugate gradient method finds a regularized inverse (*pseudo-inverse*) of $\mathbf{A}^H\mathbf{A}$. The solution that we use in this thesis comes from the optimal projection onto a Krylov space (the optimal projection being defined by equation 3.11). Finding the solution using the conjugate gradient method is practical, but represents a large computational expense. To reduce the computational burden, or eliminate it all-together, authors (e.g. Rickett, 2003; Yu et al., 2006) have proposed estimates for the pseudo-inverse $(\mathbf{A}^H\mathbf{A})^\dagger$. Sava et al. (2001) propose a pseudo-inverse that they call the Jacobian matrix $\mathbf{J}$, so that, $\mathbf{J}^\dagger \approx (\mathbf{A}^H\mathbf{A})^\dagger$. Applying $\mathbf{J}^\dagger$ to both sides of equation 3.10 gives,

$$\mathbf{J}^\dagger\mathbf{A}^H\mathbf{d} = \mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}\mathbf{m} \approx \mathbf{m}.$$

Hence, some clever choice for $\mathbf{J}^\dagger$ removes the necessity of inverting $\mathbf{A}^H\mathbf{A}$. Even if $\mathbf{J}^\dagger$ is not the pseudo-inverse of $\mathbf{A}^H\mathbf{A}$, it may work to cluster its eigenvalues, allowing iterative methods such as conjugate gradients to converge in fewer iterations, and this is the line of reasoning that we pursue. We use an estimate of $\mathbf{J}^\dagger$ derived from forward and adjoint wave-field modelling operators built using WKBJ Green's functions. This is similar to Sava et al. (2001), who showed the effectiveness of $\mathbf{J}^\dagger$ to correct amplitude versus ray-parameter gathers found by application of the adjoint $\mathcal{G}^\dagger$ in equation 3.3. Here we illustrate its effectiveness (or lack there-of) as a pre-conditioner for the least-squares system of equations.

We find $\mathbf{J}^\dagger$ using estimates of the forward wave-field modelling operator $\mathcal{G}$, and the adjoint migration operator $\mathcal{G}^\dagger$, and where the estimates of $\mathcal{G}$ and $\mathcal{G}^\dagger$ are constructed using WKBJ Green's functions (see Section 2.5). In Chapter 2, forward and adjoint operators were derived for when the image gathers were constructed in shot and geophone coordinates or midpoint and offset coordinates (equation 2.28). In this chapter, we are interested in analogous operators for when image gathers are represented by midpoint and ray-parameter. In particular,

$$\psi_s^W = \hat{\mathcal{G}}\alpha = \int_{-\infty}^{\infty} e^{i\int_{z_m}^{z'}(k_{gz}(z'')+k_{sz}(z''))dz''} \int_{-\infty}^{\infty} \delta(\mathbf{p}_h - \omega\mathbf{k}_h)\alpha(\mathbf{k}_{mx}, \mathbf{p}_h, z')d\mathbf{p}_h dz', \quad (3.23)$$

and,

$$\alpha^\dagger = \hat{\mathcal{G}}^\dagger \psi_s^W = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \delta(\mathbf{k}_h - \mathbf{p}_h/\omega) e^{-i\int_{z_m}^{z'}(k_{gz}(z'')+k_{sz}(z''))dz''} \psi_s^W(\mathbf{k}_{mx}, \mathbf{k}_{hx}; \omega) d\mathbf{p}_h d\omega. \tag{3.24}$$

We define the Jacobian operator so that,

$$\mathcal{J}(\alpha) = \alpha^\dagger = \hat{\mathcal{G}}^\dagger \hat{\mathcal{G}} \alpha.$$

In particular, substitution of equation 3.23 into equation 3.24 and letting $k_z = k_{gz} + k_{sz}$ gives,

$$\begin{aligned}
\alpha^\dagger(\mathbf{k}_m, \mathbf{p}_h, z) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \delta(\mathbf{k}_h - \mathbf{p}_h/\omega) e^{-i\int_{z_m}^{z} k_z dz''} \int_{z_m}^{z} e^{i\int_{z_m}^{z'} k_z dz''} \\
&\qquad \times \int_{-\infty}^{\infty} \delta(\mathbf{p}_h - \omega \mathbf{k}_h) \alpha(\mathbf{k}_m, \mathbf{p}_h, z') d\mathbf{p}_h dz' d\omega d\mathbf{k}_h \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \delta(\mathbf{k}_h - \mathbf{p}_h/\omega)\delta(\mathbf{p}_h - \omega\mathbf{k}_h) \\
&\qquad \times \int_{z_m}^{z} e^{i\int_{z}^{z'} k_z dz''} \alpha(\mathbf{k}_m, \mathbf{p}_h, z') dz' d\omega d\mathbf{p}_h d\mathbf{k}_h \\
&= \int_{-\infty}^{\infty}\int_{z_m}^{z} e^{i\int_{z}^{z'} k_z dz''} \alpha(\mathbf{k}_m, \mathbf{p}_h, z') dz' d\omega,
\end{aligned}$$

where $\mathbf{k}_h = \omega \mathbf{p}_h$. Using stationary phase arguments gives (Kuehl, 2002),

$$\alpha^\dagger(\mathbf{k}_m, \mathbf{p}_h, z) = \int_{-\infty}^{\infty} e^{-ik_z z} \int_{z_m}^{z} e^{ik_z z'} \alpha(\mathbf{k}_m, \mathbf{p}_h, z') dz' d\omega.$$

Next, we use the dispersion relation (equation 2.8) to make a change of variables from ω to k_z, and use the support of α to change the limits of integration, so that,

$$\begin{aligned}
\alpha^\dagger(\mathbf{k}_m, \mathbf{p}_h, z) &= \int_{-\infty}^{\infty} e^{-ik_z z} \int_{-\infty}^{\infty} e^{ik_z z'} \alpha(\mathbf{k}_m, \mathbf{p}_h, z') dz' \frac{d\omega}{dk_z} dk_z \\
&= \int_{-\infty}^{\infty} e^{-ik_z z} \alpha(\mathbf{k}_m, \mathbf{p}_h, k_z) \frac{d\omega}{dk_z} dk_z, \\
&= 2\pi \mathcal{F}_z^{-1} \left(\alpha(\mathbf{k}_m, \mathbf{p}_h, k_z) \frac{d\omega}{dk_z} \right),
\end{aligned}$$

where $\mathcal{F}_z$ is the one-dimensional Fourier transform over depth, and we have let $e^{ik_z z}$ be its Fourier kernel. Hence,

$$\alpha^\dagger(\mathbf{k}_m, \mathbf{p}_h, k_z) = 2\pi \frac{d\omega}{dk_z} \alpha(\mathbf{k}_m, \mathbf{p}_h, k_z),$$

and,

$$\alpha^\dagger(\mathbf{k}_m, \mathbf{p}_h, k_z) = \mathcal{J}(\alpha) = 2\pi \frac{d\omega}{dk_z} \alpha(\mathbf{k}_m, \mathbf{p}_h, k_z) \tag{3.25}$$

$$\alpha(\mathbf{k}_m, \mathbf{p}_h, k_z) = \mathcal{J}^{-1}(\alpha^\dagger) = \frac{1}{2\pi} \frac{dk_z}{d\omega} \alpha^\dagger(\mathbf{k}_m, \mathbf{p}_h, k_z). \tag{3.26}$$

Finally, recalling the dispersion relations (for $\omega > 0$) in equation 2.52, and using the definition of the ray-parameter in equation 3.1, we find,

$$k_z = k_{gz} + k_{sz} = \sqrt{\frac{\omega^2}{c_0^2} - \frac{1}{4}|\omega \mathbf{p}_h + \mathbf{k}_m|^2} + \sqrt{\frac{\omega^2}{c_0^2} - \frac{1}{4}|\omega \mathbf{p}_h - \mathbf{k}_m|^2}. \tag{3.27}$$

Taking the derivative of k_z in equation 3.27 with respect to ω and substituting the result into equation 3.26 gives,

$$\mathcal{J}^{-1}(\alpha^\dagger) = \frac{1}{4\pi}\left[k_{gz}^{-1}\left(\frac{\omega}{c_0^2} - \frac{1}{4}(\omega \mathbf{p}_h + \mathbf{k}_m)\cdot \mathbf{p}_h\right) + k_{sz}^{-1}\left(\frac{\omega}{c_0^2} - \frac{1}{4}(\omega \mathbf{p}_h - \mathbf{k}_m)\cdot \mathbf{p}_h\right)\right]\alpha^\dagger, \tag{3.28}$$

where assuming $\omega > 0$ and $k_z > 0$ (see Appendix B),

$$\omega = \sqrt{\frac{k_z^2 + |\mathbf{k}_m|^2}{4k_z^2/c_0^2 - |\mathbf{p}_h|^2(|\mathbf{k}_m|^2 + k_z^2)}}. \tag{3.29}$$

In equation 3.28, $\mathcal{J}^{-1}$ is an operator acting on α in a continuous space. To find its discrete form, we introduce the matrix $\hat{\mathbf{J}}^\dagger$ so that,

$$\mathcal{J}_j^{-1} = \mathcal{J}^{-1}(k_{mx(l_3)}, k_{my(l_4)}, p_{hx(l_7)}, p_{hy(l_8)}; k_{z(l_1)}; \alpha^\dagger),$$

where j was given in equation 3.8, and $\mathcal{J}_j^{-1}$ is the j^{th} diagonal element of $\hat{\mathbf{J}}^\dagger$. The matrix $\hat{\mathbf{J}}^\dagger$ operates in midpoint wave-number and depth wave-number. To find the equivalent operator $\mathbf{J}^\dagger$ in midpoint and depth, we write,

$$\mathbf{J}^\dagger = \mathbf{F}_{mz}^{-1}\hat{\mathbf{J}}^\dagger \mathbf{F}_{mz},$$

where $\mathbf{F}_{mz}$ is the three dimensional Fourier transform over midpoint and depth dimensions.

We illustrate the effect of $\mathbf{J}^\dagger$ using the data and model shown in Figure 3.2. Figure 3.2a is the acoustic velocity model (half-space over a half-space) that we use to generate data using a finite difference approximation to the acoustic wave-equation. Figure 3.2b is the acquisition geometry for the experiment showing the survey geometry with 20 common depth point locations, and 41 offsets for each common depth point data gather. Finally, Figure 3.2c is a representative midpoint gather for the experiment, and corresponds to common depth point

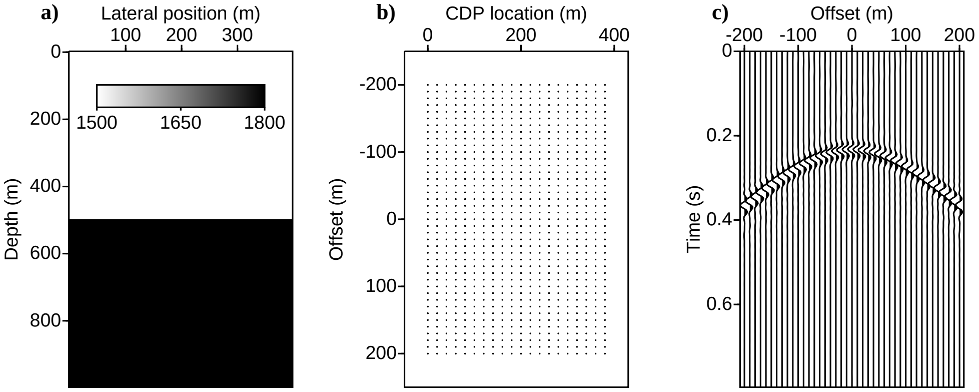

Figure 3.2: Example 1, a) acoustic velocity model (the colour-bar is shown in units of metres per second) , b) acquisition geometry, and c) a single midpoint gather corresponding to $x_m = 200m$. The traces in the data have been normalized giving, artificially, a flat amplitude response as a function of offset.

$x_m = 200m$ in Figure 3.2b. For the sake of our analysis of $\mathbf{J}^\dagger$, the amplitude of the traces in the data are normalized so that each trace has unit energy. This, artificially, introduces a flat amplitude versus offset response into the data which we should expect to see in the amplitude versus ray-parameter image gathers when the Jacobian is used to correct the adjoint. We illustrate this in Figure 3.3. In particular, Figure 3.3a plots $\mathbf{A}^H\mathbf{d}$, and in Figure 3.3b we plot $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{d}$. In these plots the image gather corresponds to the common depth point data gather in Figure 3.2c, and we see that the inverse Jacobian *flattens* the amplitude in the image gather as a function of ray-parameter. This result is similar to Sava et al. (2001), and lends confidence to our formulation and implementation of the Jacobian matrix.

For our second example, we reduce the size of the problem so that it is possible to store $\mathbf{A}^H\mathbf{A}$ in the computer, and so that we are able to compute its eigenvalues. In particular, Figure 3.4 shows the model and data for the experiment. The model is unchanged from the first example, but the size of the acquisition is reduced, and shown in Figure 3.4b. This allows us to compute and store $\mathbf{A}^H\mathbf{A}$ which we plot (for its first 120 rows and columns) in Figure 3.4c. Given $\mathbf{A}^H\mathbf{A}$, we compute its pseudo-inverse,

$$(\mathbf{A}^H\mathbf{A})^\dagger = \mathbf{U}\mathbf{\Sigma}^\dagger\mathbf{U}^H,$$

where $\mathbf{U}$ and $\mathbf{\Sigma}$ are, respectively, matrices with the eigenvectors, and eigenvalues of $\mathbf{A}^H\mathbf{A}$.

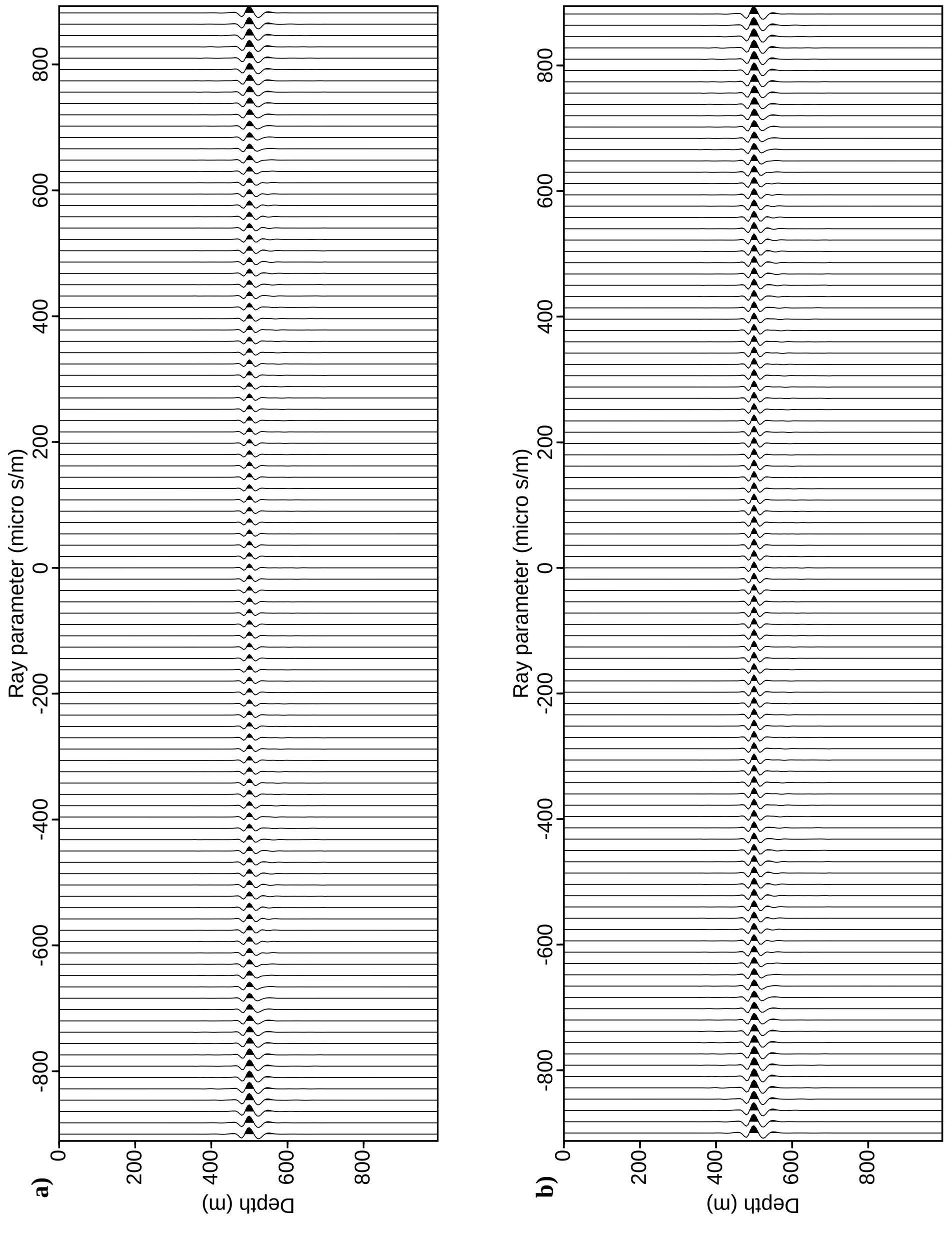

Figure 3.3: Example 1, a single amplitude versus ray-parameter image gather for $x_m = 200m$ a) with and b) without the inverse Jacobian matrix applied.

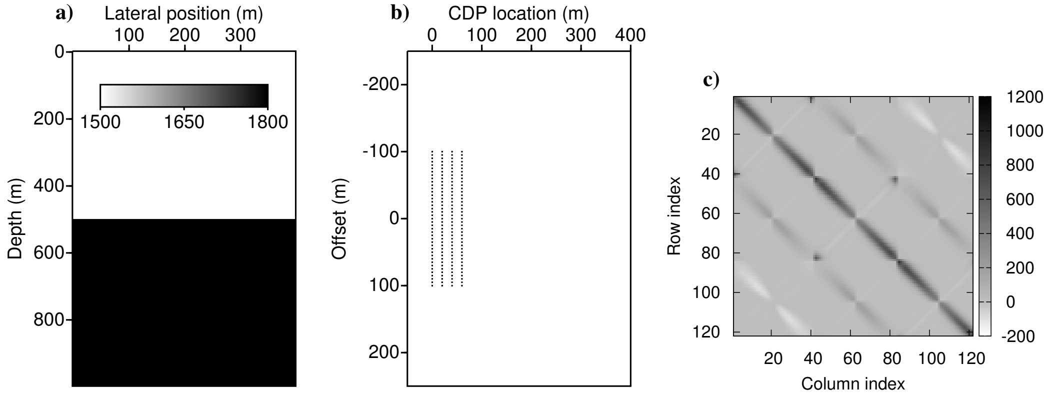

Figure 3.4: Example 2, a) acoustic velocity model (the colour-bar is shown in units of metres per second), b) acquisition geometry, and c) the first 120 rows and columns of $\mathbf{A}^H\mathbf{A}$.

The matrix $\mathbf{\Sigma}$ is a diagonal matrix with diagonal elements,

$$\left[\mathbf{\Sigma}^\dagger\right]_{ii} = \begin{cases} \sigma_{ii}^{-1} & , \quad \sigma_{ii} > \epsilon \\ 0 & , \quad \sigma_{ii} \leq \epsilon, \end{cases}$$

where ϵ is some prescribed tolerance, and σ_i is the i^{th} diagonal element of $\mathbf{\Sigma}$. The pseudo-inverse is plotted in Figures 3.5a and 3.5c for its first 120 rows and columns where Figure 3.5a plots the diagonal elements of $(\mathbf{A}^H\mathbf{A})^\dagger$. Meanwhile, the inverse Jacobian $\mathbf{J}^\dagger$ is plotted in Figure 3.5b and 3.5d where, again, Figure 3.5b plots the diagonal elements of the matrix in Figure 3.5d. Next, to illustrate the effect of the pseudo-inverse and inverse Jacobian, we plot $(\mathbf{A}^H\mathbf{A})^\dagger\mathbf{A}^H\mathbf{A}$ and $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}$ in Figures 3.5g and 3.5h, respectively, for their first 120 rows and columns. In Figures 3.5e and 3.5f, we plot their respective diagonal elements. Finally, in Figure 3.6, we plot the normalized eigenvalues of $\mathbf{A}^H\mathbf{A}$, $(\mathbf{A}^H\mathbf{A})^\dagger\mathbf{A}^H\mathbf{A}$, and $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}$ using, respectively, the dotted, dashed, and solid lines.

Our analysis shows that while the Jacobian matrix can be used to correct for amplitude (as shown in our first example), it is not necessarily effective as a pre-conditioner for least-squares migration (as shown in the second example).

3.7 Pre-conditioned conjugate gradients

The analysis in the preceding section is interesting, but can not be applied to typical seismic exploration acquisition geometries, namely due to the size of the resulting migration operators. To deal with large operators, we need to incorporate the Jacobian matrix into an

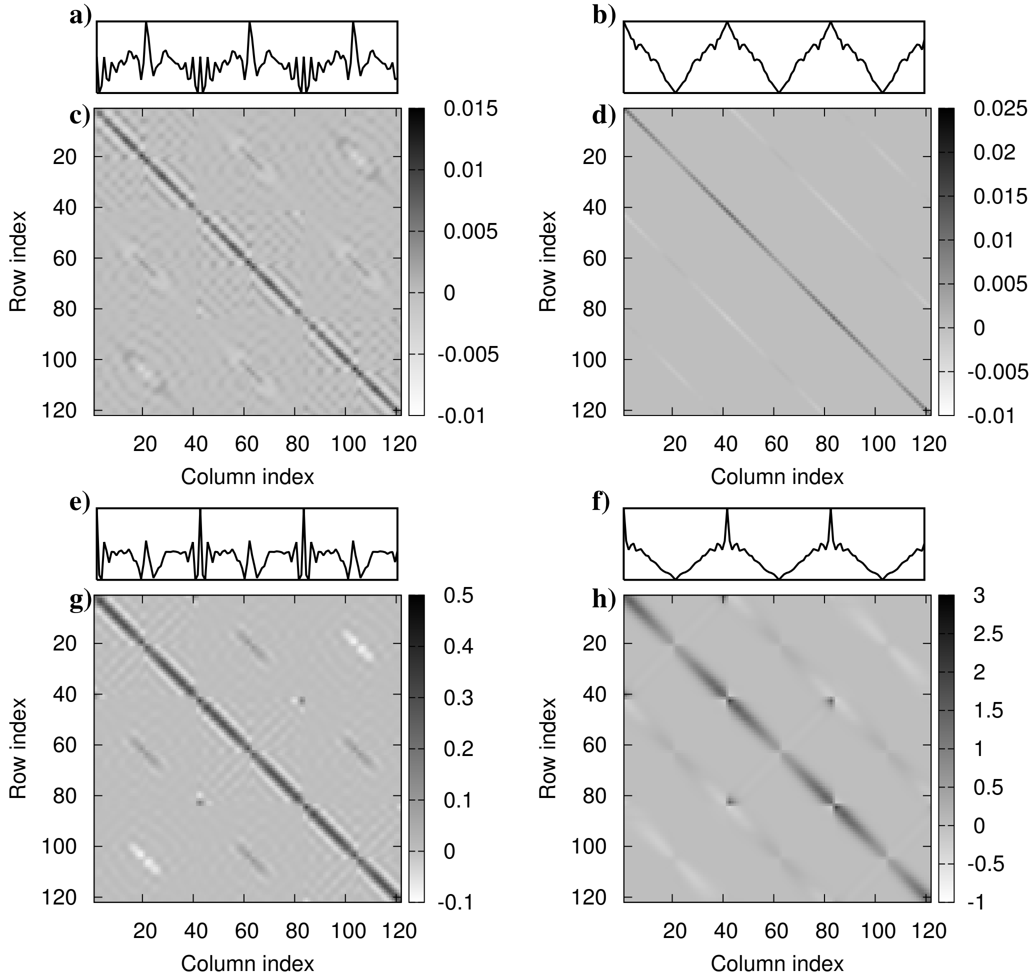

Figure 3.5: Example 2, using the first 120 rows and columns of matrices, we plot c) $(\mathbf{A}^H\mathbf{A})^\dagger$ with its diagonal elements in a), d) $\mathbf{J}^\dagger$ with its diagonal elements in b), g) $(\mathbf{A}^H\mathbf{A})^\dagger(\mathbf{A}^H\mathbf{A})$ with its diagonal elements in e), and h) $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}$ with its diagonal elements in f).

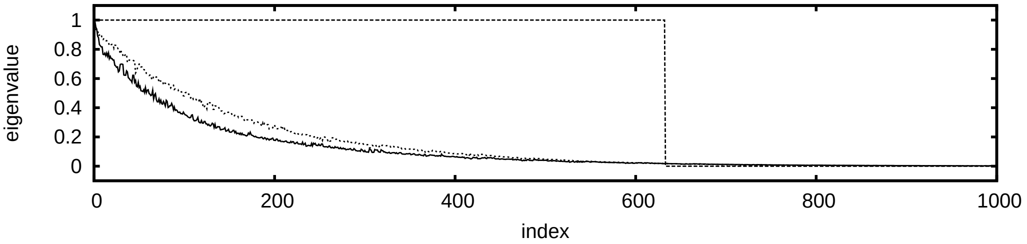

Figure 3.6: Example 2, the normalized eigenvalues of $\mathbf{A}^H\mathbf{A}$ (dotted line), $(\mathbf{A}^H\mathbf{A})^\dagger\mathbf{A}^H\mathbf{A}$ (dashed line), and $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}$ (solid line).

iterative method. In this section we use similar matrices to write down a pre-conditioned conjugate gradient method. In turn, this is used to find the solution to the pre-conditioned least-squares normal equations given by,

$$\mathbf{J}^\dagger \mathbf{A}^H \mathbf{A}\mathbf{m} = \mathbf{J}^\dagger \mathbf{A}^H \mathbf{d}. \tag{3.30}$$

In the previous section, we showed that for small acquisition geometries using the Jacobian matrix as a pre-conditioner does not cluster the eigenvalues of the least-squares system of equations. In this section, we show that the same conclusion can be shown for its Ritz values, and an industry standard 2D marine acquisition geometry (the Marmousi data). We begin by writing down the pre-conditioned least-squares conjugate gradient method used to solve equation 3.30. Then, we use the pre-conditioned least-squares conjugate gradient method to compute Ritz values, drawing conclusions about the effectiveness of the Jacobian matrix $\mathbf{J}^\dagger$ as a pre-conditioner.

The pre-conditioned conjugate gradient method can be derived by introduction of a matrix $\mathbf{E}$ such that $\mathbf{J} = \mathbf{E}\mathbf{E}^H$, and so that from equations 3.30,

$$\mathbf{E}^{-H}\mathbf{E}^{-1}\mathbf{A}^H\mathbf{A}\mathbf{m} = \mathbf{E}^{-H}\mathbf{E}^{-1}\mathbf{A}^H\mathbf{d} \Rightarrow \mathbf{E}^{-1}\mathbf{A}^H\mathbf{A}\mathbf{E}^{-H}\widehat{\mathbf{m}} = \mathbf{E}^{-1}\mathbf{A}^H\mathbf{d}, \tag{3.31}$$

where $\mathbf{m} = \mathbf{E}^{-H}\widehat{\mathbf{m}}$. It is important to note that the $\mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}$ and $\mathbf{E}^{-1}\mathbf{A}^H\mathbf{A}\mathbf{E}^{-H}$ are similar matrices, sharing the same eigenvalues. In particular, for some vector $\mathbf{v}$, we can write,

$$\begin{aligned}\mathbf{E}^{-1}\mathbf{A}^H\mathbf{A}\mathbf{E}^{-H}\mathbf{v} = \lambda\mathbf{v} \Rightarrow \mathbf{E}^{-H}\mathbf{E}^{-1}\mathbf{A}^H\mathbf{A}\mathbf{E}^{-H}\mathbf{v} &= \lambda(\mathbf{E}^{-H}\mathbf{v}) \\ \Rightarrow \mathbf{J}^\dagger\mathbf{A}^H\mathbf{A}(\mathbf{E}^{-H}\mathbf{v}) &= \lambda(\mathbf{E}^{-H}\mathbf{v}).\end{aligned}$$

This means that the effect that $\mathbf{J}^\dagger$ has in pre-conditioning the system in equation 3.30 is the same as the effect of $\mathbf{E}$ in equation 3.31. Direct substitution of the normal equations (3.31) into the conjugate gradient method (Algorithm 3.1) gives Algorithm 3.2 (see Appendix B). As was the case with the conjugate gradient method in Algorithm 3.1, the pre-conditioned conjugate gradient method in Algorithm 3.2 contains redundancies allowing for the forward and adjoint operators to be applied only once per iteration. Additionally, we notice that while $\mathbf{E}$ is important to the construction of the pre-conditioned conjugate gradient method, it does not appear in the final result (see Appendix B).

For an example, we consider, again, the synthetic Marmousi data. For this data, the migration operator is far too large to store as a matrix. Instead, we use the conjugate gradient and pre-conditioned conjugate gradient methods to compute its Ritz values that are, in turn, a subset of the eigenvalues for, respectively, the least-squares operator without and with pre-conditioning using the Jacobian matrix. In Figure 3.7, we show the experiment for the Marmousi data. In particular, Figure 3.7a plots the velocity model, and Figure 3.7b

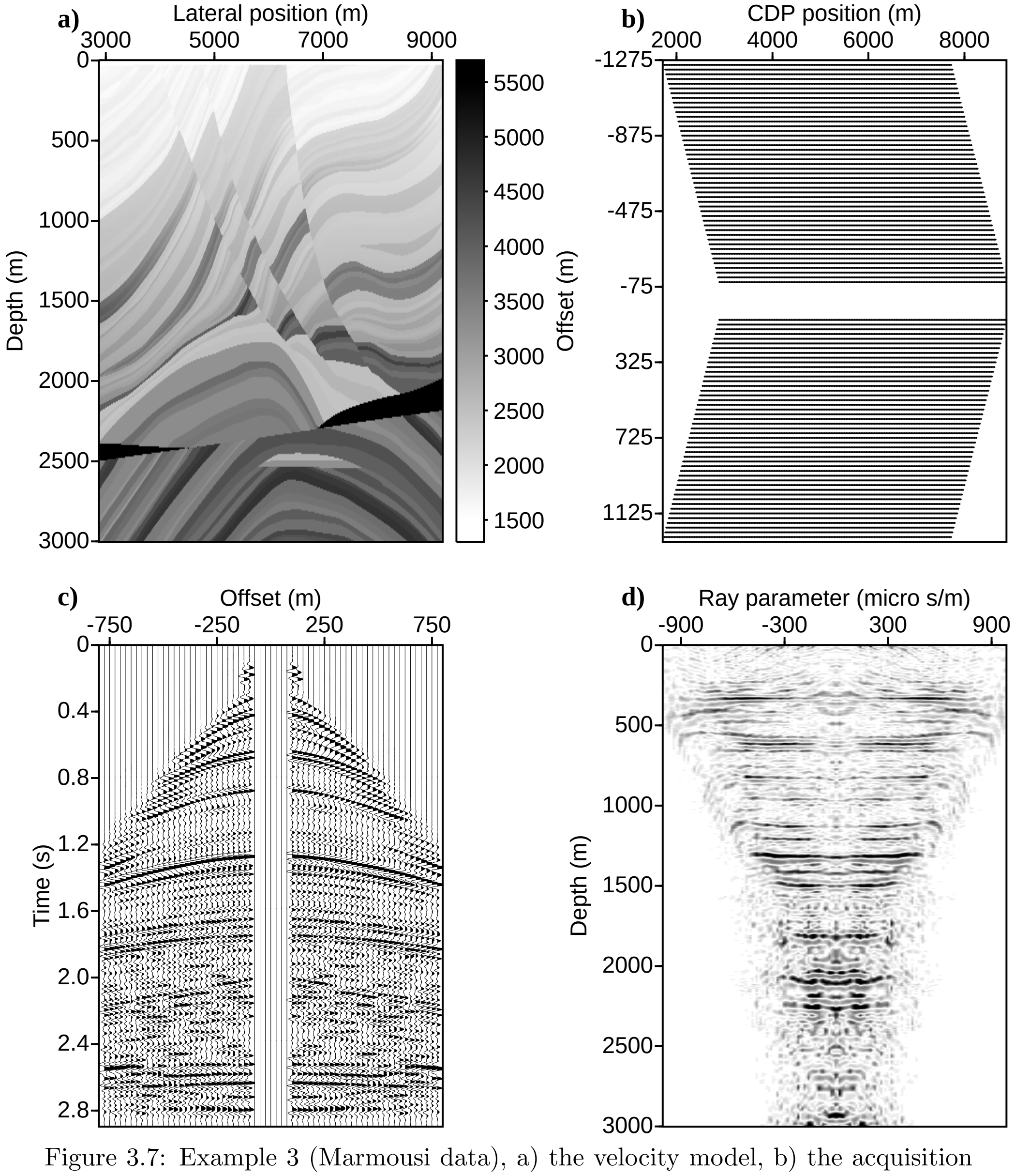

Figure 3.7: Example 3 (Marmousi data), a) the velocity model, b) the acquisition geometry, c) an example common depth point gather, and d) its corresponding amplitude versus ray-parameter image gather found using the least-squares conjugate gradient method.

Algorithm 3.2 The preconditioned conjugate gradient method for the pre-conditioned least-squares system in equation 3.30.

```
$\mathbf{m}_0 = \mathbf{0}$, $\mathbf{r}_0 = \mathbf{A}^H\mathbf{d}$, $\mathbf{p}_0 = \mathbf{J}^\dagger\mathbf{r}_0$
for $k = 1$ to $n$ do
    $\alpha_k = (\mathbf{r}_k^H\mathbf{J}^\dagger\mathbf{r}_k)/(\mathbf{p}_{k-1}^H\mathbf{A}^H\mathbf{A}\mathbf{p}_{k-1})$
    $\mathbf{m}_k = \mathbf{m}_{k-1} + \alpha_k\mathbf{p}_{k-1}$
    $\mathbf{r}_k = \mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{m}_k)$
    $\beta_k = (\mathbf{r}_k^H\mathbf{J}^\dagger\mathbf{r}_k)/(\mathbf{r}_{k-1}^H\mathbf{J}^\dagger\mathbf{r}_{k-1})$
    $\mathbf{p}_k = \mathbf{J}^\dagger\mathbf{r}_k + \beta_k\mathbf{p}_{k-1}$
end for
```

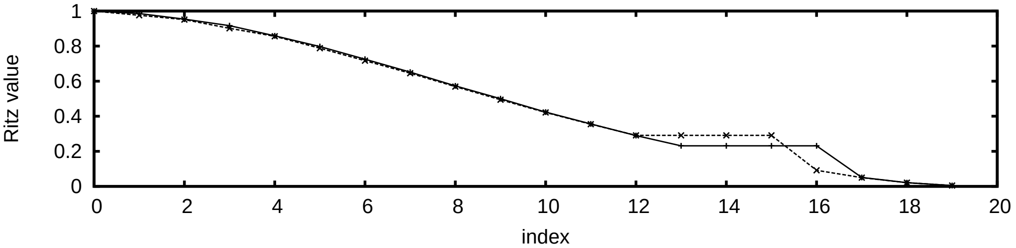

Figure 3.8: Example 3 (Marmousi data), the Ritz values compute for 20 iterations of the least-squares conjugate gradient method (solid line), and b) the pre-conditioned least-squares conjugate gradient method (dashed line).

plots the acquisition geometry. Figure 3.7c shows data for a single common depth point position, and Figure 3.7d shows the corresponding amplitude versus ray-parameter gather computed using 20 iterations of the least-squares conjugate gradient method. Using the pre-conditioned conjugate gradient method for 20 iterations produces an equivalent result. Figure 3.8 plots the normalized Ritz values computed for the first 20 iterations of the least-squares conjugate gradient method (solid line), and pre-conditioned least-squares conjugate gradient method (dashed line). It is evident that pre-conditioning the least-squares equations with the Jacobian matrix does not cluster the Ritz values. This is analogous to the conclusion reached for the clustering of eigenvalues in Section 3.6 where an atypical small survey acquisition was used.

3.8 Summary

In this Chapter, we showed that the while Jacobian matrix does correct for amplitude in amplitude versus ray-parameter image gathers, it, unfortunately, does not work as a pre-conditioner for least-squares migration. This Chapter introduced two additional and important concepts to this thesis. First, it provided an example of regularized migration in that we found migrated image gathers by solving a set of least-squares normal equations. The regularization was governed by a projection of the solution onto a Krylov space using the conjugate gradient method. Second, it introduced the conjugate gradient method, itself, which will be used extensively throughout the remainder of the thesis.

■ CHAPTER 4

Regularized migration with sparseness constraints

4.1 Introduction

In Chapter 2, we derived forward and adjoint operators for least-squares migration with a variable reference velocity. Within the context of source-receiver migration, we considered three parameterizations for the scattering potential, common shot image gathers, common midpoint image gathers, and amplitude versus ray-parameter image gathers, the latter being introduced in Chapter 3. In this chapter, we concentrate on source-receiver migration for either common shot image gathers or common midpoint image gathers. In particular, we apply our prior expectation that energy from data gathers focuses to where shot and receiver are coincident in the image gathers. Our method is similar to Wang and Sacchi (2007) in that it regularizes the common image gathers using a Cauchy norm (sparse prior), but differs in its choice of model space. Whereas Wang and Sacchi (2007) use amplitude versus ray-parameter image gathers, we use either common shot or common midpoint image gathers. In these domains, we expect energy to migrate to where shot and receiver are coincident, or offset is null (Claerbout, 1971). This, in turn, motivates the use of a sparse prior in all dimensions of each common image gather. In contrast, Wang and Sacchi (2007) use a sparse prior in the depth direction, and a smooth prior in the ray-parameter direction. Our choice of sparse prior (the Cauchy norm) leads to a non-quadratic optimization problem that is solved using iterative re-weighted least-squares (IRLS) (e.g. Gersztenkorn et al., 1986).

We begin with a review of Claerbout's imaging condition, and, in particular, how it relates to the forward and adjoint operators developed in Chapter 2. This leads us to choose a sparseness criterion for regularized wave-equation migration which we introduce using Bayes theorem and a Cauchy prior. In addition, we use a mixed Cauchy-Gaussian prior, such that the Cauchy prior is used within each common midpoint image gather and the Gaussian prior is used across image gathers. We show the effect of the sparse prior with synthetic examples. We show two examples. In the first, we use a point scatterer in a variable reference wave-speed, and in the second we consider two specular reflectors, also in a variable reference wave-speed.

4.2 Source-receiver migration and Claerbout's imaging condition

In Chapter 2, we derived equations for source-receiver wave-equation migration. In equation 2.11, we found the parameterization $\alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z)$, and by extension, wrote $\alpha = \alpha(\mathbf{x}_g, \mathbf{x}_s, z)$. This choice has interesting consequences. For example, in Chapter 3 it allowed for the re-parameterization of α in depth, midpoint and ray-parameter. Here, we use it to predict the behaviour of $\alpha(\mathbf{x}_g, \mathbf{x}_s, z)$ using Claerbout's imaging condition.

For the scattered wave-field $\psi_s(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega)$, Claerbout's imaging condition as stated in Stolt (1978) is,

$$\alpha(\mathbf{x}, z) \sim \psi_s(\mathbf{x}, z|\mathbf{x}, z; t = 0). \tag{4.1}$$

The ideas contained within equation 4.1 were introduced in Chapter 2. Again, we state its interpretation that when the wave-field is coincident with the scattering point, time is taken to be be zero. In source-receiver migration, the scattering potential is parameterized by the survey parameters, and the interpretation becomes slightly more involved. To understand this, we analyze the adjoint to the Born approximation in equation 2.38, and which we re-write here for constant velocity Green's functions (equation 2.7), so that for a single frequency (see Appendix C),

$$\alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z; \omega) \propto e^{-i\omega\cos\theta_g\tau_{gz}}\psi_s(\mathbf{k}_{gx}, z_g|\mathbf{k}_{sx}, z_s; \omega)e^{-i\omega\cos\theta_s\tau_{sz}}, \tag{4.2}$$

where τ_{gz} is the vertical travel-time for the plane-wave propagating from the measurement surface $(\mathbf{x}_g, z_g)$ to the scattering point $(\mathbf{x}, z)$, and τ_{sz} is the vertical travel-time for the plane-wave propagating from the scattering point to the source $(\mathbf{x}_s, z_s)$. In particular,

$$\tau_{gz} = \frac{z - z_g}{c_0} \qquad\qquad \tau_{sz} = \frac{z - z_s}{c_0}.$$

The angle θ_g was defined in Chapter 2, and is the angle to the vertical of the normal to the receiver side plane-wave propagating from the measurement surface to the scattering point. Likewise, θ_s is the angle to the vertical of the normal to the source side plane-wave propagating from the scattering point to the measurement surface. In equation 4.2, the seismic source is excited at time $t = 0$, and $\exp(-i\omega\cos\theta_g\tau_{gz})$ is proportional to the Green's function that propagates the wave-field from the measurement surface to the scattering

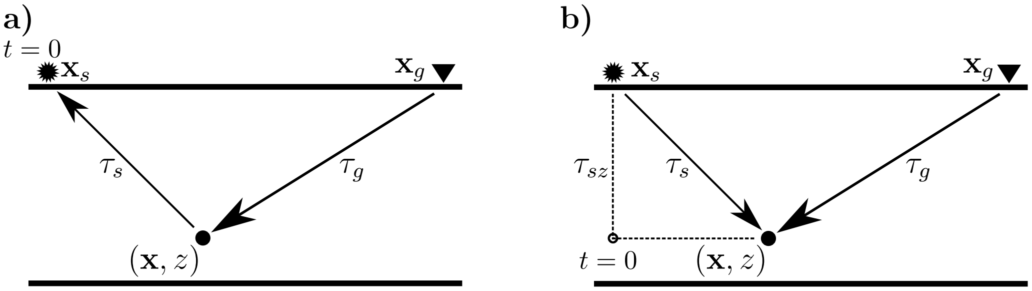

Figure 4.1: We illustrate the shift in the time axis for the adjoint operator: a) the travel-time t axis origin is coincident with origin of the experiment (when the seismic shot is excited), b) the vertical travel-time axis origin is coincident with the scattering point. In b), the lateral component of travel-time remains coincident with the origin of the experiment.

point. Similarly, $\exp(-i\omega\cos\theta_s\tau_{sz})$ is proportional to the Green's function that propagates the wave-field from the scattering point to the source (the origin of the experiment).

Following the logic in Claerbout's imaging condition, we re-interpret the vertical travel-time axis so that the wave-field is coincident (in depth) with the scattering point at zero vertical travel-time. Under this new frame of reference, τ_{sz} is negative and τ_{gz} is positive, so that equation 4.2 becomes,

$$\alpha(\mathbf{k}_{gx}-\mathbf{k}_{sx}, z; \omega) \propto e^{-i\omega\cos\theta_g\tau_{gz}}\psi_s(\mathbf{k}_{gx}, z_g|\mathbf{k}_{sx}, z_s; \omega)e^{i\omega\cos\theta_s\tau_{sz}}. \tag{4.3}$$

In equation 4.3, we see that both Green's functions propagate the wave-field toward the depth of the scattering point, but in the shifted frame of reference, $\exp(i\omega\cos\theta_s\tau_{sz})$ is propagating the wave-field through negative vertical travel-time τ_{sz} (Figure 4.1 gives a schematic, showing the shift of the time axis).

We note that the shift of the travel-time axis origin is limited to its vertical component. Therefore, in the projection of travel-time onto the horizontal, the time origin for the experiment remains at $\mathbf{x}_s$ (illustrated in Figure 4.1). This leads to a further consequence, namely that when travel-time $t=0$ is coincident with the scattering point in depth, it is also coincident with the source location in the lateral dimensions. In turn, this affects our expectation of $\alpha(\mathbf{x}_g, \mathbf{x}_s, z)$, and which is also described by the imaging condition. In particular, we assume that within this new frame of reference the adjoint experiment is causal, meaning that for $t<0$ the wave-field is null. This means that in equation 4.3, $\alpha(\mathbf{k}_{gx}-\mathbf{k}_{sx}, z)$, or equivalently $\alpha(\mathbf{x}_g, \mathbf{x}_s, z)$, will have energy only where $\mathbf{x}_g$ is coincident with $\mathbf{x}_s$.

The preceding description of Claerbout's imaging condition allows us to place our prior expectation onto $\alpha(\mathbf{x}_g, \mathbf{x}_s, z)$. Namely that its energy is isolated to where $\mathbf{x}_g = \mathbf{x}_s$, and where there is some non-zero scattering potential. In migration, it is typically assumed that the reference velocity c_0 is chosen so that in depth the scattering potential is sparse, tracking the earth's reflectors. Therefore, in each pre-stack common shot image gather (or common midpoint image gather), the scattering potential will be sparse.

4.3 Sparse inversion

The inversion method that we use is posed as the minimization of an objective function, similar to Wang and Sacchi (2007), and Sacchi et al. (1998). We let $\mathbf{d}$ be observed seismic data organized in adjacent common shot or common midpoint data gathers in space and frequency, so that its realizations are taken from $\psi_s(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega)$. Likewise, we let $\mathbf{m}$ be the scattering potential organized in adjacent common image gathers, and realized from $\alpha(\mathbf{x}_g, \mathbf{x}_s, z)$. Additionally, we let $\mathbf{A}$ be the split-step wave-field modelling operator derived in Chapter 2. The methods for mapping from the continuous functions and operators to vectors $\mathbf{m}$ and $\mathbf{d}$, and the matrix $\mathbf{A}$ were shown in Chapter 3. Finally, we define a noise vector, $\mathbf{n} = \mathbf{d} - \mathbf{A}\mathbf{m}$.

We derive a cost function for minimization using Bayes rule. From this, we find a set of non-linear normal equations that are solved using iterative re-weighted least squares (IRLS). To proceed, we let $\mathbf{d}$, $\mathbf{m}$ and $\mathbf{n}$ be random vectors, and state Bayes rule (e.g. Ulrych et al., 2001),

$$p(\mathbf{m}|\mathbf{d}) = \frac{p(\mathbf{d}|\mathbf{m})p(\mathbf{m})}{p(\mathbf{d})}, \tag{4.4}$$

where the denominator,

$$p(\mathbf{d}) = \int_{-\infty}^{\infty} p(\mathbf{d}|\mathbf{m})p(\mathbf{m})d\mathbf{m}, \tag{4.5}$$

is constant. It is constant because it integrates over all possible choices for $\mathbf{m}$. In the numerator of equation 4.4, $p(\mathbf{d}|\mathbf{m})$ is called the data likelihood function, and is a measure of how well the forward modelled data matches the observed data. The numerator also houses the prior probability density function for the model $p(\mathbf{m})$ (in our case, the model is the scattering potential in common shot, or common midpoint, image gathers). Finally, $p(\mathbf{m}|\mathbf{d})$ is the posterior probability density function. In what follows, we consider the data-likelihood and the prior probability density functions separately before substituting them in Bayes rule, giving the posterior probability density function. To find an optimal realization of $\mathbf{m}$, we maximize the posterior density. The resulting estimate of $\mathbf{m}$ is called the maximum posterior probability (MAP) estimator.

First, we consider the data likelihood probability density function, letting $p(\mathbf{d}|\mathbf{m}) = p(\mathbf{n})$, and where we take the components of $\mathbf{n}$, n_i, $i = 1 \dots M$ to be independent Gaussian distributed random variables such that $n_i \sim N(0, \sigma_{n(i)})^1$, and,

$$p(\mathbf{n}) = \prod_{i=1}^{M} p(n_i) = \prod_{i=1}^{M} \frac{1}{\sqrt{2\pi}\sigma_{n(i)}} \exp\left(-\frac{n_i^* n_i}{2\sigma_{n(i)}^2}\right), \tag{4.6}$$

where n_i^* is the complex conjugate of n_i. Given a mean and variance, and in the absence of other information, equation 4.6 is the maximum entropy distribution for the noise vector (see Appendix F.1, or Cover and Thomas, 2006, Ch. 12), and therefore, we can consider it to be a suitable choice for the data-likelihood probability density function.

Second, we consider, in equation 4.4, the prior probability density function $p(\mathbf{m})$. We let m_i, $i = 1 \ldots N$ be the components of $\mathbf{m}$. We assume that m_i are independently and identically distributed random variables so that, as with the data likelihood, $p(\mathbf{m}) = p(m_1)p(m_2)\ldots p(m_N)$. Finally, we assume that m_i, $i = 1 \ldots N$ are distributed as Cauchy probability density functions. Hence,

$$p(\mathbf{m}) = \prod_{i=1}^{N} \frac{1}{2\pi\sigma_m^2} \left(1 + \frac{m_i^* m_i}{2\sigma_m^2}\right)^{-1}, \tag{4.7}$$

where σ_m is called the scale parameter (e.g. Johnson and Kotz, 1972). It is interesting to compare the Cauchy probability density function to the Gaussian probability density function. In Figure 4.2, we plot the Cauchy probability density function, along with the Gaussian probability density function, and notice that the Cauchy function has longer *tails* than the Gaussian function. The long tails increase the chance of recovering few large elements to explain the energy in $\mathbf{m}$, in turn giving its sparse character. In contrast, the Gaussian distribution lacks long tails so that if we used it for the prior, then the MAP estimator would be more likely to explain the energy of $\mathbf{m}$ with many small elements clustered around zero, rather than few (sparsely distributed) large elements. We note that using a Gaussian prior probability function would, in particular, lead to a damped least-squares formulation (the undamped least-squares solution was studied in Chapter 3).

Given the data likelihood in equation 4.6 and the prior in equation 4.7, we write down the posterior probability density function, and find a realization of $\mathbf{m}$, say $\mathbf{m}_*$, such that it is maximum (i.e. $\mathbf{m}_*$ is the MAP estimator of $\mathbf{m}$). In particular, substituting equations 4.6

[1] $N(0, \sigma_{n(i)})$ denotes a Gaussian probability density function with mean 0 and standard deviation $\sigma_{n(i)}$

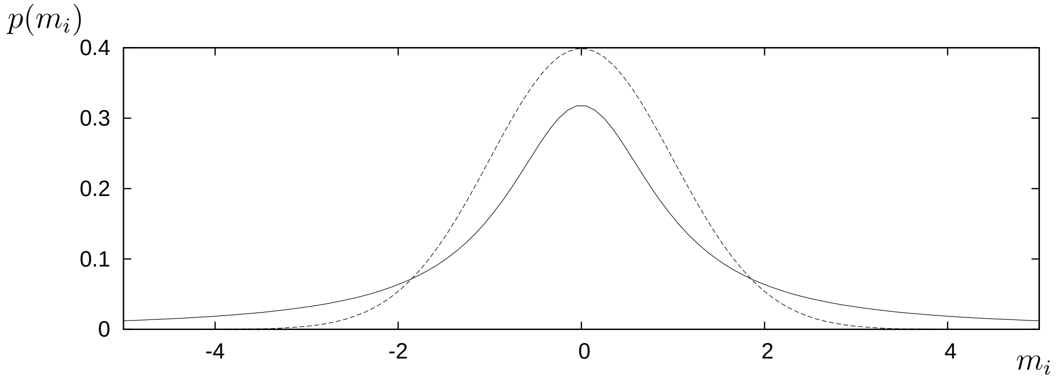

Figure 4.2: We plot probability density functions for the Gaussian and Cauchy distributions. The Cauchy distribution is the solid line, and the Gaussian distribution is the dashed line. We notice that the Cauchy distribution has longer tails than the Gaussian distribution, promoting a sparse solution for the MAP estimator.

and 4.7 into the numerator of equation 4.4, gives the MAP estimator,

$$\begin{aligned}\mathbf{m}_* = \arg\max_{\mathbf{m}} p(\mathbf{m}|\mathbf{d}) =& \arg\max_{\mathbf{m}} \prod_{i=1}^{M} \exp\left(-\frac{n_i^* n_i}{2\sigma_{n(i)}^2}\right) \prod_{i=1}^{N} \left(1+\frac{m_i^* m_i}{2\sigma_m^2}\right)^{-1} \\ =& \arg\max_{\mathbf{m}} \exp\left[-\sum_{i=1}^{M} \frac{n_i^* n_i}{2\sigma_{n(i)}^2} - \sum_{i=1}^{N} \ln\left(1+\frac{m_i^* m_i}{2\sigma_m^2}\right)\right] \quad (4.8) \\ =& \arg\min_{\mathbf{m}} \left[||\mathbf{W}_d \mathbf{n}||_2^2 + \sum_{i=1}^{N} \ln\left(1+\frac{m_i^* m_i}{2\sigma_m^2}\right)\right], \quad (4.9)\end{aligned}$$

where $\mathbf{W}_d$ is a diagonal matrix such that its i^{th} diagonal element is,

$$[\mathbf{W}_d]_{ii} = \frac{1}{\sqrt{2}\sigma_{n(i)}}. \quad (4.10)$$

Equivalently, $\mathbf{m}_*$ in equation 4.9 is found by minimization of the cost function,

$$\phi(\mathbf{m}) = ||\mathbf{W}_d(\mathbf{d}-\mathbf{A}\mathbf{m})||_2^2 + \sum_{i=1}^{N} \ln\left(1+\frac{m_i^* m_i}{2\sigma_m^2}\right). \quad (4.11)$$

For convenience, we define a matrix $\widehat{\mathbf{W}}_d$ so that $\mathbf{W}_d = \sqrt{2}\sigma_n^{-1}\widehat{\mathbf{W}}_d$, where in the simplest case (when n_i are identically distributed), $\widehat{\mathbf{W}}_d = \mathbf{I}$ and $\mathbf{I}$ is the identity matrix. Taking the gradient of equation 4.11 with respect to $\mathbf{x}$, and setting this result to zero, gives the normal equations,

$$\left(\mathbf{A}^H \widehat{\mathbf{W}}_d^H \widehat{\mathbf{W}}_d \mathbf{A} + \sigma_n^2 \mathbf{Q}\right) \mathbf{m} = \mathbf{A}^H \widehat{\mathbf{W}}_d^H \widehat{\mathbf{W}}_d \mathbf{d}, \quad (4.12)$$

where $\mathbf{Q}$ is a diagonal matrix such that its i^{th} diagonal element is,

$$[\mathbf{Q}]_{ii} = \left(\sigma_m^2 + m_i^* m_i/2\right)^{-1}. \tag{4.13}$$

In equation 4.13, we recognize that $\mathbf{Q}$ is a function of the model $\mathbf{m}$, and, therefore, equation 4.12 is nonlinear.

To solve the non-linear normal equations (4.12), we use IRLS, an iterative algorithm (e.g. Scales and Gersztenkorn, 1988). During the course of each IRLS iteration $\mathbf{Q}$ is held constant, making equation 4.12 temporarily linear, and allowing for a linear conjugate gradient algorithm to solve for $\mathbf{m}$. In turn, this solution updates $\mathbf{Q}$ before proceeding to the next IRLS iteration. In particular, we let $\mathbf{m}_0 = \mathbf{0}$, and $\mathbf{m}_j$, $k = 1, 2 \ldots$ be the solution to equation 4.12 after the k^{th} IRLS iteration. Additionally, we let $\mathbf{Q}_k$, $k = 1, 2 \ldots$ be the matrix $\mathbf{Q}$ after the k^{th} IRLS iteration. In each iteration, $\mathbf{m}_{k-1}$ is used to construct $\mathbf{Q}_k$ according to equation 4.13 (see Algorithm 4.1).

Algorithm 4.1 Iterative re-weighted least-squares

```
$\mathbf{m}_0 = \mathbf{0}$
for $k = 1$ to $n$ do
    construct $\mathbf{Q}_k$ using $\mathbf{m}_{k-1}$
    $\mathbf{m}_k = \left(\mathbf{A}^H \widehat{\mathbf{W}}_d^H \widehat{\mathbf{W}}_d \mathbf{A} + \sigma_n^2 \mathbf{Q}_k\right)^{-1} \mathbf{A}^H \widehat{\mathbf{W}}_d^H \widehat{\mathbf{W}}_d \mathbf{d}^{obs}$ (via conjugate gradients)
end for
```

Two parameters control the influences of the data likelihood and prior density function on the inversion. Namely, the scale parameter in the Cauchy probability density function σ_m, and the standard deviation of the noise σ_n. First, we consider the scale parameter. If we choose a large scale parameter, so that for all i, $[\mathbf{Q}]_{ii} >> m_i^* m_i$, then $\mathbf{Q} \sim \sigma_m^2 \mathbf{I}$, and equation 4.12 becomes equivalent to the damped least-squares normal equations, giving a *small*, rather than a sparse solution. If, on the other hand, $\sigma_m << m_i^* m_i$, then a sparse solution is promoted. However, if σ_m is chosen too small when m_i are also small, the normal equations become unstable due to the inversion in equation 4.13. Second, we consider the standard deviation of the noise. The noise variance controls a *trade-off* between fitting the data and honouring the prior distribution for the model. If we choose a small σ_n, then the inversion will work to fit the data more than it works to honour the model prior. Conversely, a large σ_n will work to honour the model prior more than it works to fit the data. The noise variance plays an additional role, making the inversion well conditioned. We safely (given the discussion in Chapter 3) assume that the matrix $\mathbf{A}^H \widehat{\mathbf{W}}_d^H \widehat{\mathbf{W}}_d \mathbf{A}$ is rank-deficient, and that $\sigma_n^2 \mathbf{Q}$ acts as a regularization term, giving a well-conditioned matrix for inversion. Selecting σ_n too small will, therefore, produce ill-conditioned normal equations. Having said that, in this thesis we solve the linearized normal equations using linear conjugate gradients so that choosing a small σ_n finds an optimal solution for the constructed Krylov space (see Chapter 3). Within the context of Bayes, this is also called the maximum likelihood solution, or, in other words, the solution that finds where $p(\mathbf{d}|\mathbf{m})$ is maximum.

In the two examples in Sections 4.5 and 4.6, we fix σ_m to a small number, promoting a sparse solution, and then choose σ_n based on an $L{-}curve$ analysis. In this analysis (e.g. Lawson and Hanson, 1974), we plot data misfit versus model norm, and where the curve is parameterized by σ_n. The resulting curve tends to be shaped like the letter L. We use the heuristic that the optimal σ_n is found at the bend in the L, or, in other words, where the second derivative of the curve is maximum. We take a liberty here, as the $L - curve$ analysis is typically used in the damped least-squares problem, and here we apply it to normal equations derived using a sparse regularization; although, recently it has been studied for the class of $L1$ problems using the related Pareto curves (Hennenfent et al., 2008).

4.4 Inversion with a mixed norm Cauchy-Gaussian model prior

Thus far, we have motivated a sparse (Cauchy) model prior using Claerbout's imaging condition. In particular, we argued that each common shot, or common midpoint, image gather is characterized by a sparse prior probability density function. While this prior is effective, it lacks fair treatment of all model dimensions. While it justifiably accounts for sparsity in depth and offset, it neglects to justify its prior for the shot dimensions (in the case of common shot image gathers) or midpoint dimensions (in the case of common midpoint image gathers). In this section, we consider the problem for common midpoint image gathers, and so that in the midpoint dimension, we assume only the mean and variance of the corresponding prior probability density function. This means that, as was the case for the data likelihood, a Gaussian prior is applicable. Hence, we would like to characterize the model with a Gaussian prior in the midpoint dimensions, and a sparse prior in the depth and offset dimensions. This idea is also mentioned in Hermmann (2009), and can be thought of as an extension to the sparse inversion discussed in Section 4.3 and Kaplan (2008). The solutions of this type for general inversion problems are discussed in, for example, Kowalski and Torrésani (2009) (and references therein) and are applied to a geophysical problem in Rodriguez et al. (2010). The method results in what is commonly referred to as a *mixed norm.* In this section we describe the changes to the sparse inversion shown in Section 4.3 when the mixed norm is substituted for the norm resulting from the Cauchy model prior.

In equation 4.8 the contribution from the model prior probability density function is,

$$\phi_c(\mathbf{m};\sigma_m) = \sum_{i=1}^{N} \ln\left(1 + \frac{m_i^* m_i}{2\sigma_m^2}\right), \tag{4.14}$$

so that equation 4.8 can be written as,

$$\mathbf{m}_* = \arg\max_{\mathbf{m}} \exp\left[-\sum_{i=1}^{M} \frac{n_i^* n_i}{2\sigma_{n(i)}^2} - \phi_c(\mathbf{m})\right]$$

The function ϕ_c is the model norm for the sparse prior. The mixed norm replaces ϕ_c with $\phi_{c,g}$ so that,

$$\phi_{c,g}(\mathbf{m};\sigma_m) = \sum_{i=1}^{N_c} \ln\left(1 + \frac{\left(\sqrt{\sum_{j=1}^{N_g} m_{i,j}^* m_{i,j}}\right)^2}{2\sigma_m^2}\right). \tag{4.15}$$

In equation 4.15 the elements of $\mathbf{m}$ are $m_{i,j}$ for $i = 1 \ldots N_c$ and $j = 1 \ldots N_g$, and where $N = N_c N_g$, so that,

$$\mathbf{m}^T = \left[\begin{array}{cccccccc} m_{1,1} & m_{1,2} & \cdots & m_{1,N_g} & m_{2,1} & m_{2,2} & \cdots & m_{N_c,N_g} \end{array}\right].$$

In the language of the mixed norm (Kowalski and Torrésani, 2009), i indexes the *group* and j indexes the *members* of the i^{th} group. In other words, the model norm function in equation 4.15 assigns a Gaussian-like prior among the members of each group, and assigns a Cauchy-like prior across groups. For the regularized migration in this section, then, we assign the midpoint dimension to the index j, and we assign the depth and offset dimensions to the index i. In turn, this assigns a Gaussian-like prior to the midpoint dimension and a Cauchy-like prior to the depth and offset dimensions.

The insertion of the mixed norm in equation 4.15 leads to the normal equations (4.12), but with $\mathbf{Q}$ replaced with a new matrix $\mathbf{Q}_{c,g}$ and which is derived from the mixed norm $\phi_{c,g}$. In particular, taking the gradient of $\phi_{c,g}$ with respect to $\mathbf{m}$ gives,

$$\nabla\phi_{c,g}(\mathbf{m}) = 2\mathbf{Q}_{c,g}\mathbf{m},$$

where $\mathbf{Q}_{c,g}$ is a diagonal matrix such that its i^{th} diagonal element is,

$$[\mathbf{Q}_{c,g}]_{ii} = \left(\sigma_m^2 + \frac{\sum_{j=1}^{N_g} m_{k,l}^* m_{k,l}}{2}\right)^{-1},$$

and where $i = kM + l$, but where we note that the value of $[\mathbf{Q}_{c,g}]_{ii}$ is independent of the index l.

In the example in Section 4.6 we show, in addition to the sparse inversion described in Section 4.3, the mixed norm inversion described in this section and which gives a more robust result. We, of course, attribute this robustness to the prior information removed (sparsity) from the midpoint dimension. We note that the mixed norm solution is engineered from rather than derived from Bayes theorem. That is, we did not find equation 4.15 by the direct application of Gaussian and Cauchy density functions to Bayes theorem. Instead, we substituted a new model norm for equation 4.14 that retained, for the groups, the form that follows from a Cauchy prior, but uses the $L2$ norm for the members within each group, and which is motivated by a Gaussian prior. While this engineered result lacks rigour, its appeal to our intuition, and its utility (as will be seen in Section 4.6) warrants its inclusion in our discussion.

4.5 Example: point diffractor

The model and data for the first example are shown in Figure 4.3. Figure 4.3a plots the reference velocity model $c_0(x, z)$, and Figure 4.3b plots the earth's velocity model so that the scattering potential is the point diffractor shown in Figure 4.3b. Using the model in Figure 4.3b, we conduct a synthetic reflection seismic experiment. We place a seismic source and geophones at depth $z_m = 350m$, and work to construct common midpoint data gathers by conducting several experiments where in each experiment, the lateral positions of the source and geophones are changed. The midpoints run between $0m$ and $400m$, spaced every $20m$, and the offsets $((x_g - x_s)/2)$ are spaced every $10m$, again, between $0m$ and $400m$. Data is generated one shot gather at a time using a finite difference approximation (with a stencil that is 4th order in space, and 2nd order in time) to the scalar wave-equation with absorbing boundary conditions. The data are plotted in Figure 4.3c. To construct the $L - curve$, we let $\sigma_m = 10^{-5}$, and solve equation 4.12 for several realizations of σ_n, generating the curve in Figure 4.4. The figure is annotated with values of σ_n^2 (the noise variance).

In Figure 4.4 we see the expected behaviour of the $L - curve$. Namely that as σ_n increases so does the data misfit, while the model norm decreases. Conversely as σ_n is made smaller, the data misfit shrinks, and the model norm grows. Moreover, we see the bend in the $L - curve$ (where $\sigma_n^2 \approx 0.004$), and which is typically identified with the optimal choice for σ_n. We consider the solution for three choices of σ_n that represent three regimes on the $L - curve$. First, we consider the case where $\sigma_n^2 = 0.1$ where honouring the model prior is more important than fitting the data. Second, we consider the case where $\sigma_n^2 = 0.00001$, and where the model prior is given little to no weight. Third, we use the optimal choice of σ_n, where the second derivative of the $L - curve$ is maximum.

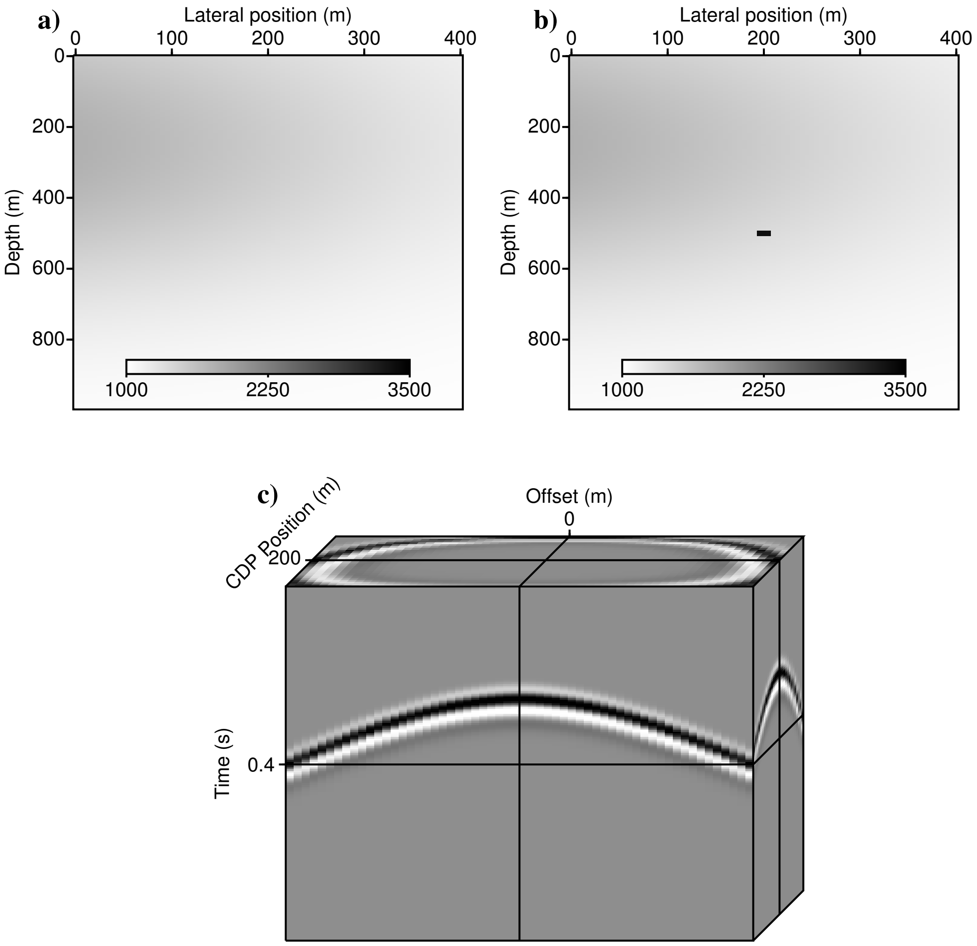

Figure 4.3: Point diffractor example: a) the reference velocity model, b) the earth's velocity model, and c) the data. The units for the colour-bars are metres per second.

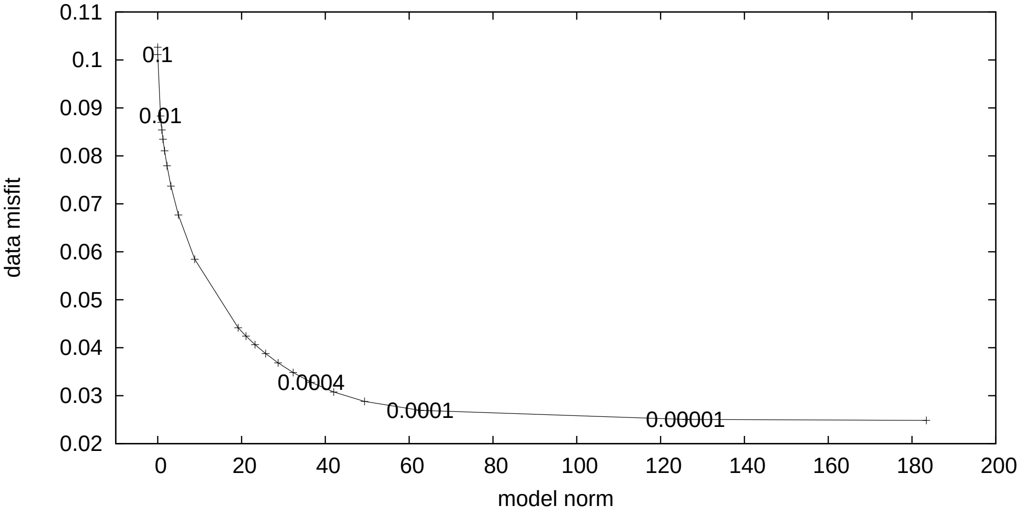

Figure 4.4: Point diffractor example, the $L-curve$ computed from several instances of the MAP estimator for equation 4.12. Each instance of the MAP estimator corresponds to a different choice for σ_n. The values of σ_n^2 are annotated along the curve.

Figure 4.5 plots the results for $\sigma_n^2 = 0.1$. Figure 4.5a is $\mathbf{m}_1$, the solution after the first IRLS iteration, and Figures 4.5b and 4.5c are, respectively $\mathbf{m}_2$ and $\mathbf{m}_3$, the solutions after the second and third IRLS iterations. The solution $\mathbf{m}_2$ has the desired sparse character, but very small amplitudes. The small amplitudes are afforded by our choice of σ_n which doesn't require the model to fit the data. Due to the small amplitudes, the solution becomes unstable, and diverges from the sparse solution. This is evident in Figure 4.5c where the solution becomes less sparse, as compared to Figure 4.5b.

Figure 4.7 plots the results for $\sigma_n^2 = 0.00001$, giving, effectively, the maximum likelihood (minimum norm) solution. In other words, the algorithm makes a best effort to fit the data while ignoring the model norm. This makes the problem essentially linear, allowing convergence in one IRLS iteration. Figures 4.7a-c show the results (common midpoint image gathers) after, respectively, the first, second and third IRLS iterations, and illustrates convergence in one iteration.

Finally, Figure 4.7 plots the optimal solution (according to the heuristic of the $L - curve$) for our particular choice of the Cauchy scale parameter σ_m, and in Figures 4.7a-c we show the result after, respectively, the first, second and fifth IRLS iterations. This solution, gives both a sparse solution, and a solution that adequately (within the bounds of the split-step approximation to the physics) fits the data. We note that for this example the sparse prior is applicable in all dimensions of the model, including the midpoint dimension.

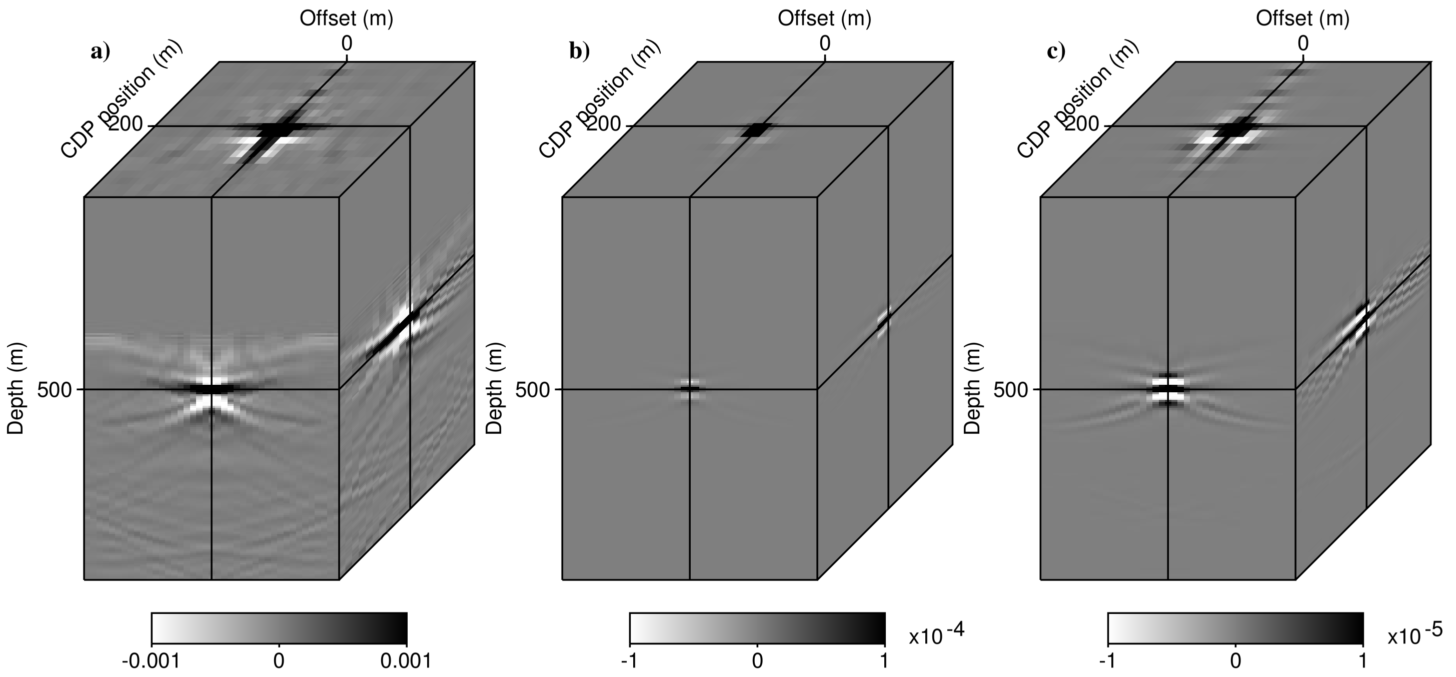

Figure 4.5: Point diffractor example for a large σ_n, the common image gathers after the a) first b) second and c) third IRLS iterations.

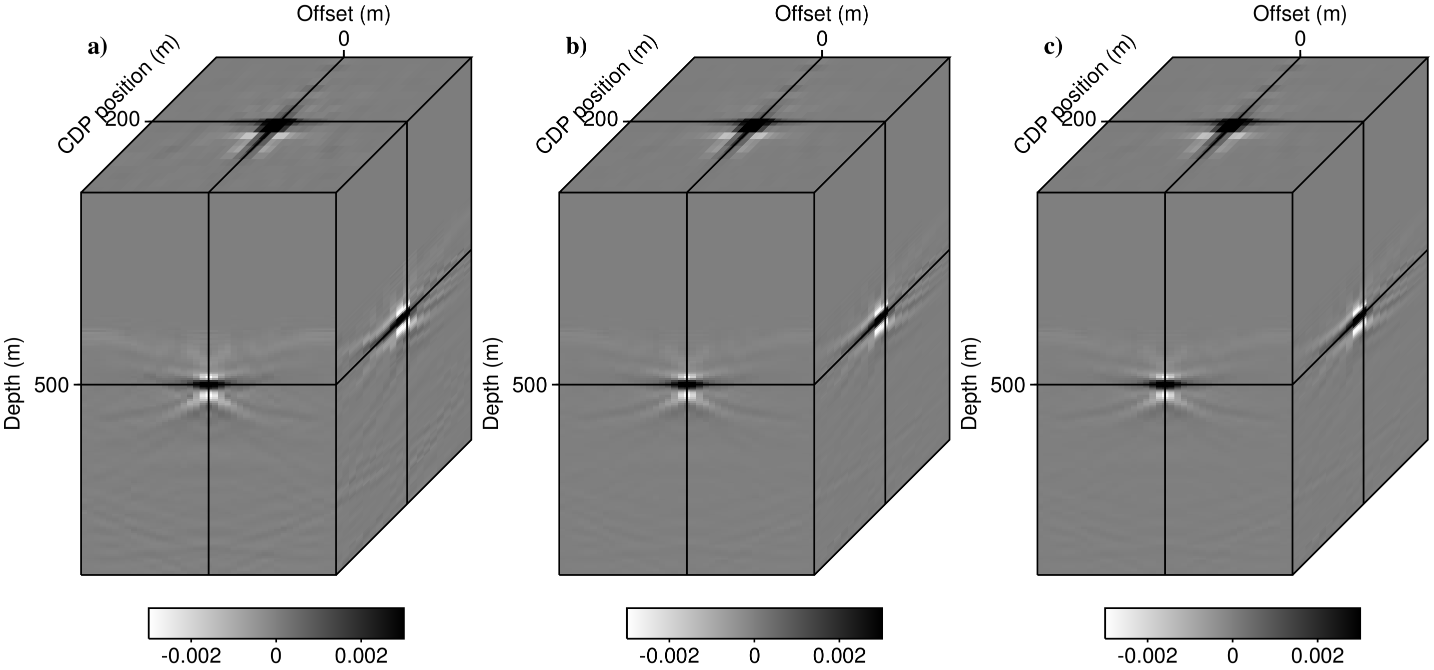

Figure 4.6: Point diffractor example for a small σ_n, the common image gathers after the a) first b) second and c) third IRLS iterations.

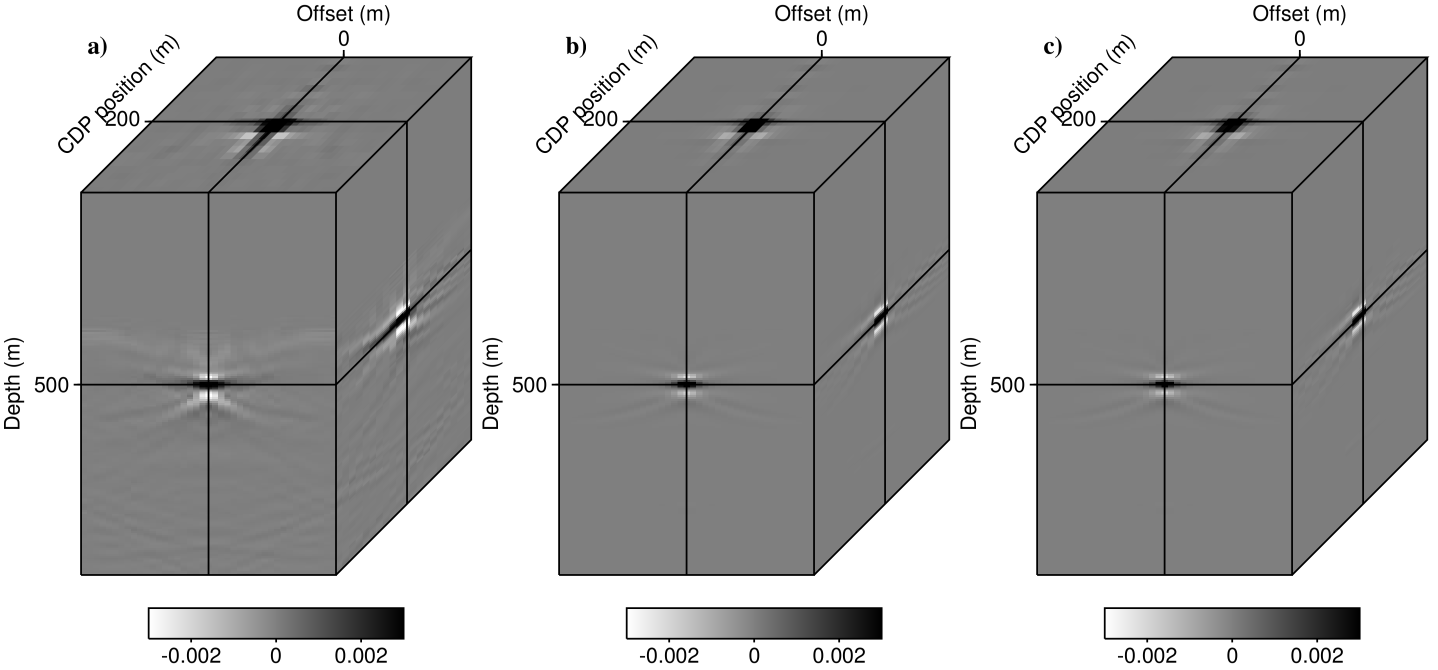

Figure 4.7: Point diffractor example for optimal σ_n, the common image gathers after the a) first b) second and c) fifth IRLS iterations.

4.6 Example: specular reflections

To convince ourselves that the method has merit for data containing specular reflections, as well as the point diffractor in the previous example, we repeat the same experiment. But this time, we use the earth model in Figure 4.8b, and the reference velocity model in Figure 4.8a. As before, we plot the data in Figure 4.8c. Since the reflectors are purely specular, an image of the reflectors (in time) can be seeing by looking at a zero-offset slice through the data (shown on the right side of the data cube in Figure 4.8c). We note that we expect the specular reflector to produce a model that is not sparse in the midpoint dimension. Therefore, we compare the spares inversion (Section 4.3) result with the mixed norm (Section 4.3) result, the latter of which does not impose a sparsity assumption in the midpoint dimension.

As in the previous example, we set $\sigma_m = 10^{-5}$, and search for an optimal σ_n using the $L - curve$ analysis. The $L - curve$ is shown in Figure 4.9a for the sparse inversion, and Figure 4.9b for the mixed norm inversion. Again, we plot solutions for three choices of σ_n where we either under-fit the data (Figure 4.10), over-fit the data (Figure 4.11), or chose the optimal solution (where the second derivative of the $L - curve$ is maximum (Figure 4.12)).

For the sparse inversion, the results, even for the optimal σ_n, are less convincing than the point-diffractor example. None-the-less, in the offset dimension the solution becomes, as expected, more sparse. In contrast, and as expected, the results from the mixed norm inversion are sparse in offset and depth, but maintain the character of the reflector in midpoint.

4.7 IRLS convergence

As already mentioned, IRLS is used to produce the results in this chapter. IRLS is an iterative algorithm used to solve equation 4.12 in which $\mathbf{Q}$ is a function of $\mathbf{m}$, making the equation nonlinear. However, during the course of each IRLS iteration $\mathbf{Q}$ is held constant, making equation 4.12 temporarily linear, and allowing for a linear conjugate gradient algorithm to solve for $\mathbf{m}$. In turn, this solution updates $\mathbf{Q}$ before proceeding to the next IRLS iteration. Therefore, there are two levels of convergence that need to be considered. The first is the convergence of $\mathbf{Q}$, and the second is the convergence of the conjugate gradient algorithm at each IRLS iteration. Further, we expect that at the conclusion of each IRLS iteration, the recovered model (common image gathers) will reproduce, within the bounds of some misfit, the data (common midpoint gathers) according to the forward operator (equations 2.10 and 2.16). That is, we assume two things: first that the inverse problem is inherently underdetermined, and second that the approximation to the physics (split-step de-migration with

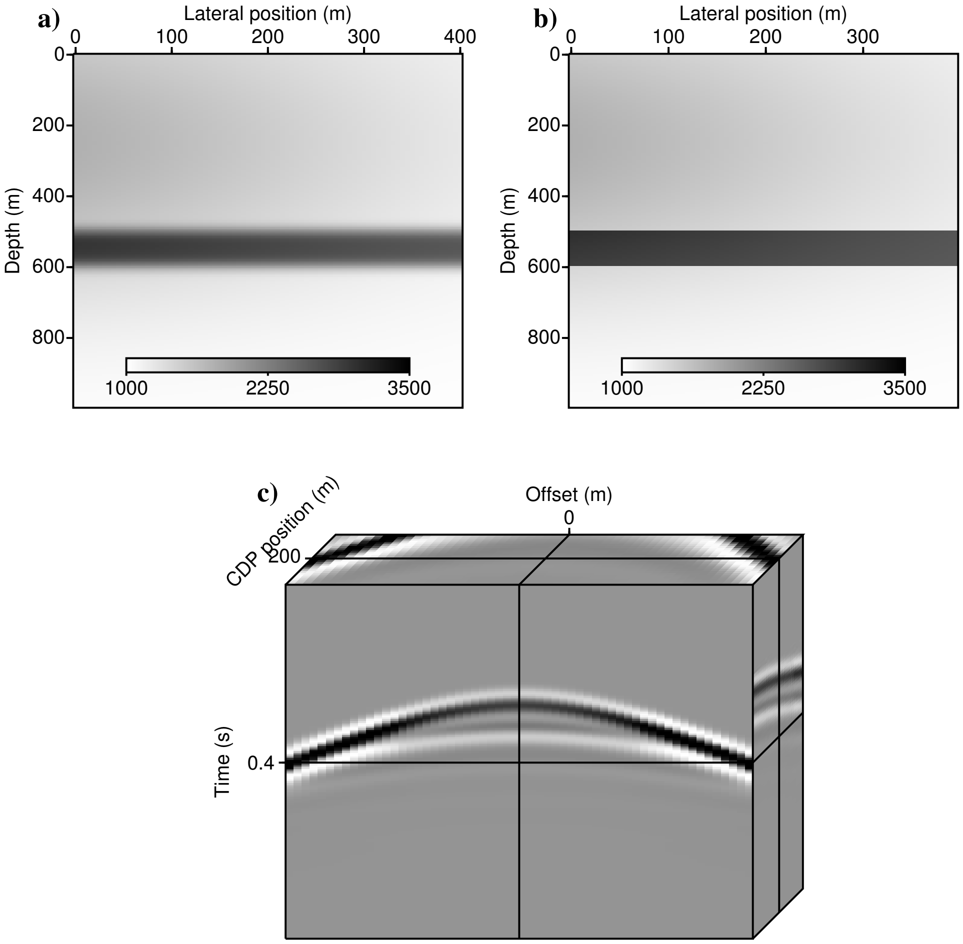

Figure 4.8: Specular reflection example, a) the reference velocity model, b) the earth's velocity model, and c) the data. The units for the colour-bars are metres per second.

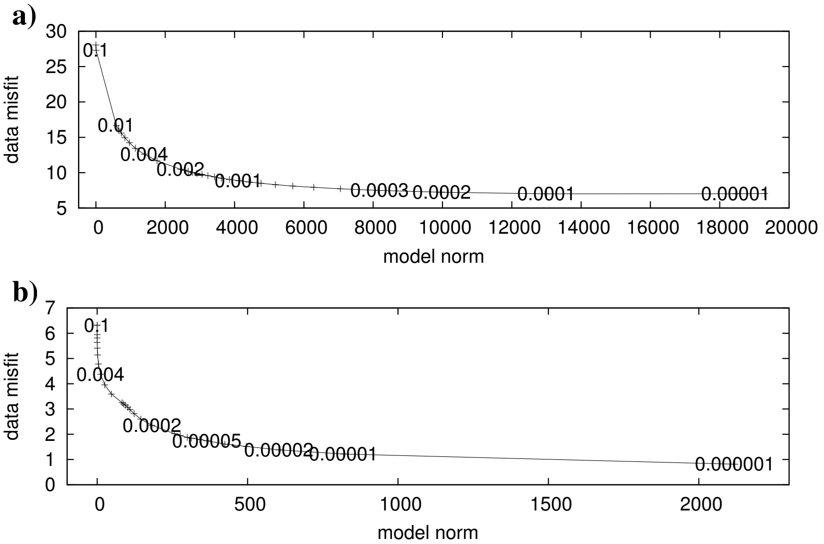

Figure 4.9: Specular reflection example, the $L - curves$ computed from several instances of the MAP estimator for equation 4.12, using a) $\mathbf{Q}$ (the sparse norm) and b) $\mathbf{Q}_{c,g}$ (the mixed norm). Each instance of the MAP estimator corresponds to a different choice for σ_n. The values of σ_n^2 are annotated along the curves.

a reference velocity model) does not model the data without error. Figure 4.13 illustrates convergence for the specular reflection example. In particular, it plots the data misfit for each of the five IRLS iterations, illustrating the desired convergence at each IRLS step, as well as the convergence of the IRLS algorithm.

4.8 Summary

In this chapter, we discussed novel regularization techniques for source-receiver migration. In particular, our choice of model space (pre-stack common midpoint or common shot image gathers) allowed us to assume that within each gather the solution is sparse. We motivated the assumption using the imaging condition, and illustrated its use with two synthetic data examples. In the first example, we used a point diffractor, and in the second a model that produced specular reflections. In addition, we used a mixed norm to remove the sparsity assumption from the midpoint (or shot) dimension. Not surprisingly, the mixed norm solution gave more robust results, especially when applied to the specular reflection example in Section 4.6. While the examples give promising results, further tests on more complicated data (containing a more realistic sequence of reflectors) showed a lack of robustness, often improving the resolution of some reflectors, while destroying the continuity of others. Hence, there is an opportunity for further research to improve the robustness of the method.

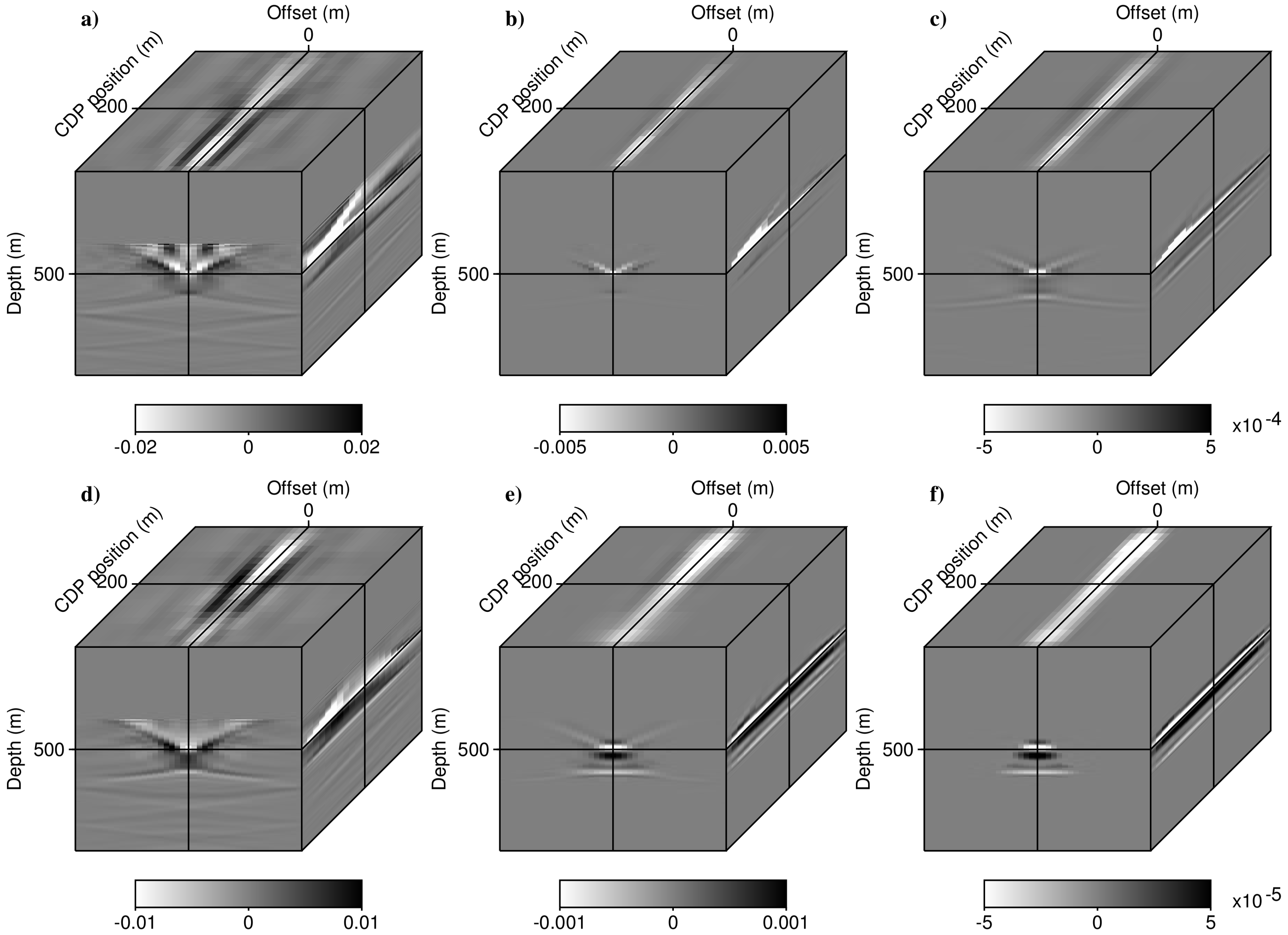

Figure 4.10: Specular reflecton example for a large σ_n, the common image gathers computed using $\mathbf{Q}$ (the sparse norm) after the a) first b) second and c) third IRLS iterations, and the common image gathers computed using $\mathbf{Q}_{g,c}$ (the mixed norm) after the d) first e) second and f) third IRLS iterations.

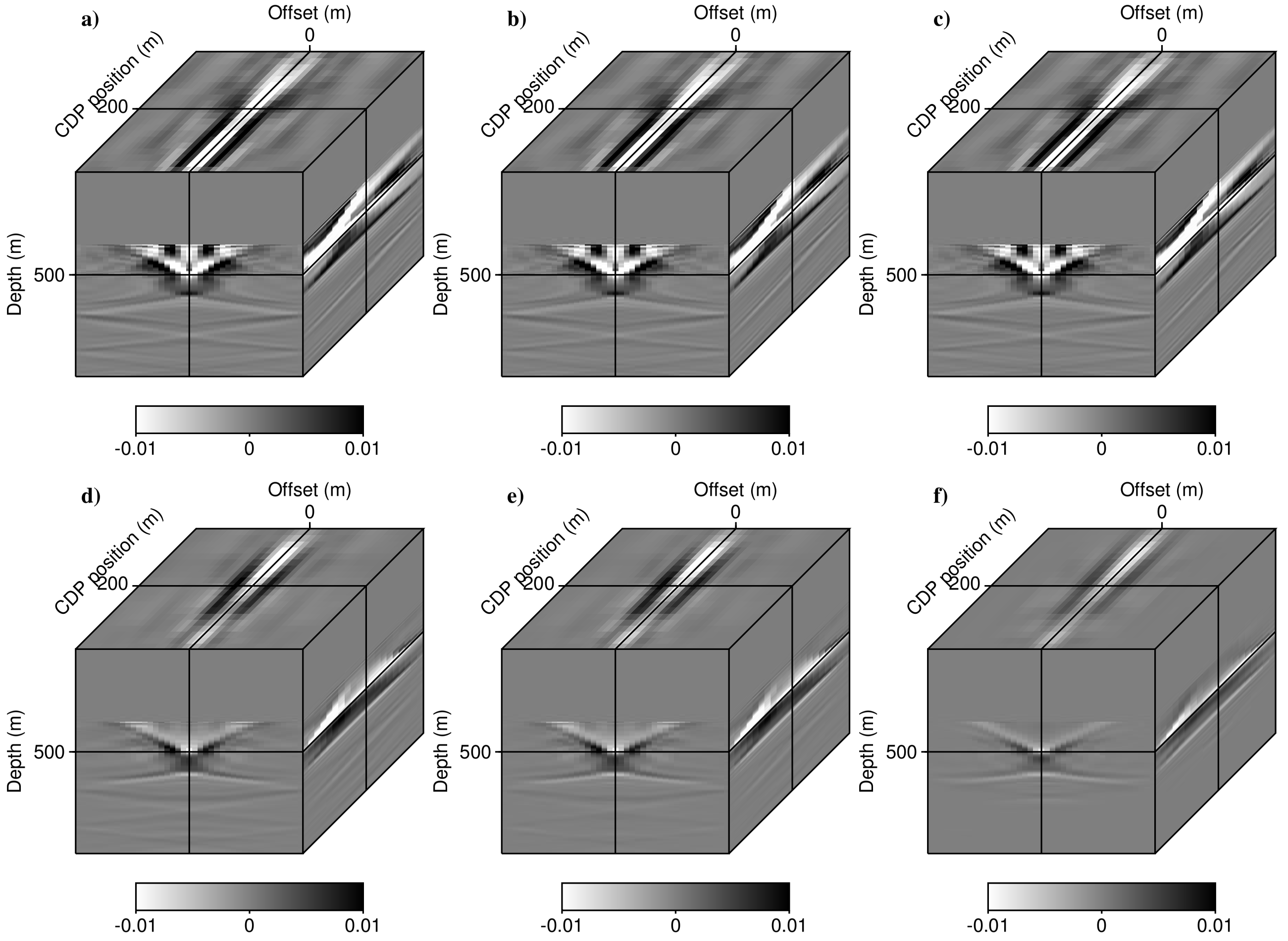

Figure 4.11: Specular reflection example for a small σ_n, the common image gathers computed using $\mathbf{Q}$ (the sparse norm) after the a) first b) second and c) third IRLS iterations, and the common image gathers computed using $\mathbf{Q}_{g,c}$ (the mixed norm) after the d) first e) second and f) third IRLS iterations.

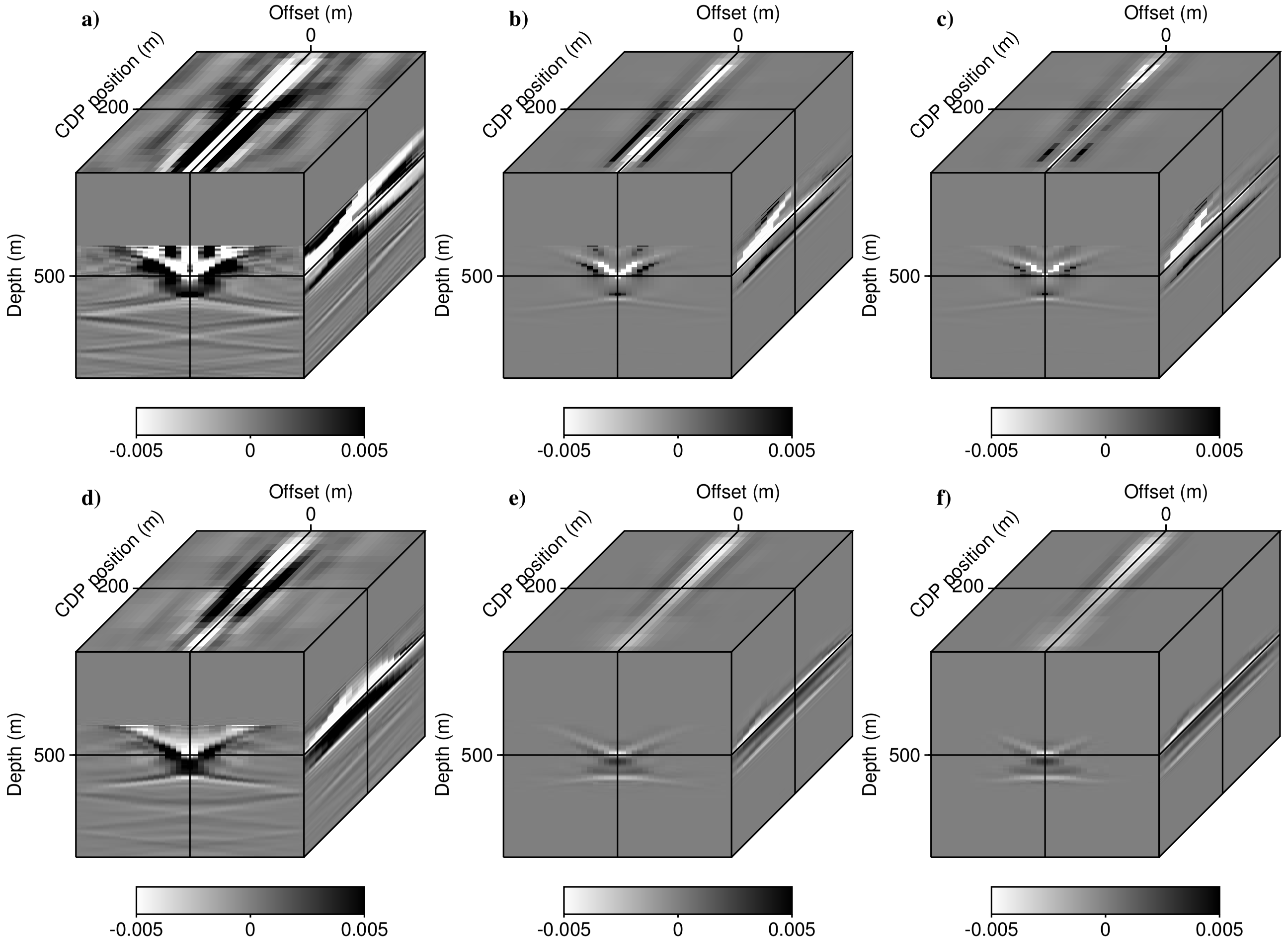

Figure 4.12: Specular reflection example for optimal σ_n, the common image gathers computed using $\mathbf{Q}$ (the sparse norm) after the a) first b) second and c) fifth IRLS iterations, and the common image gathers computed using $\mathbf{Q}_{g,c}$ (the mixed norm) after the d) first e) second and f) fifth IRLS iterations.

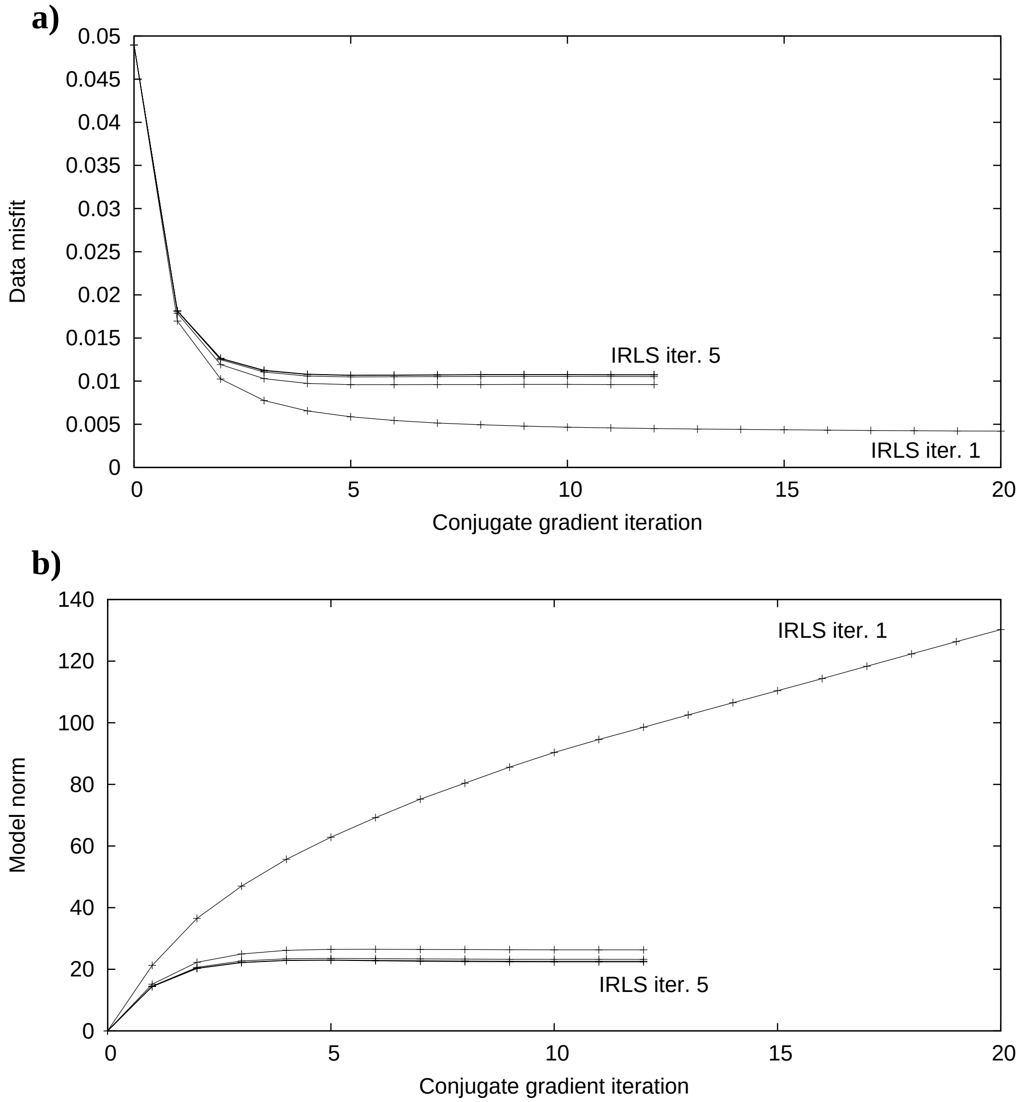

Figure 4.13: Point diffractor example, a) the data misfit and b) the model norm, both plotted against the conjugate gradient iteration. Each curve is produced by its corresponding IRLS step, with five IRLS iterations in total.

■ CHAPTER 5

Derivation of forward and adjoint operators for shot-profile least-squares migration

5.1 Introduction

In Chapter 2, we derived wave-field modelling (forward) and migration (adjoint) operators for least-squares migration using a Gazdag like algorithm and the split-step approximation. In doing so, we chose a source-receiver parameterization for the scattering potential α such that its lateral dimensions are parameterized by the pre-stack acquisition geometry (source and receiver locations or midpoint and offset locations). The source-receiver parameterization fell out of the mathematics, and can be attributed to the representation of the Green's function in Section 2.3. In equation 2.6, for example, the Green's function is represented in a mix of space and wave-number coordinates. The resulting wave-field modelling and migration algorithms are typically referred to as either survey-sinking or double-square-root migration. In these approaches to migration the Fourier transform of the scattering potential $\alpha(\mathbf{x}, z)$ is $\alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z)$, and where $\mathbf{k}_{gx} - \mathbf{k}_{sx}$ is the Fourier conjugate variable of lateral space $\mathbf{x}$. In turn this allows the parameterization of α in terms of the acquisition geometry so that, by extension, we write $\alpha(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z)$ and $\alpha(\mathbf{x}_g, \mathbf{x}_s, z)$ (see for example equation 2.11 in Chapter 2).

In this chapter, we derive similar split-step wave-field modelling (forward) and migration (adjoint) operators for the shot-profile parameterization of the scattering potential. In contrast to the source-receiver parameterization, the lateral dimensions of α need not be related to the acquisition geometry. Moreover, the shot-profile parameterization lends to shot-profile wave-field modelling and migration operators that are applicable to a single shot gather. Although the equivalence between the shot-profile and source-receiver geometries has been shown (Wapenaar and Berkhout, 1987; Biondi, 2003), from a practitioners point of view the difference in the respective parameterizations of the pre-stack migrated image gathers is interesting (Jeannot, 1988). First, it has a practical advantage in that implementing an algorithm applicable to one shot is more flexible than implementing an algorithm applicable to an entire three dimensional seismic survey. Second, it allows for more flexible distributions of seismic source energy (i.e. the forcing term in the Helmholtz equation). This is the subject of *blended migration* and is discussed in, for example, Sava (2007), Berkhout et al. (2009) and Dai and Schuster (2009). Third, the shot-profile parameterization will allow for the data reconstruction algorithms that we introduce in Chapter 6.

We begin with the shot-profile derivation of the wave-field modelling (forward) and migration (adjoint) operators using the shot-profile geometry and the Born approximation. Previously, Rickett (2001, 2003) described similar algorithms for wave-field modelling and migration using operator notation. In this chapter, we write down, explicitly, the algebra, and use the operators in a least-squares migration algorithm, showing results from two examples. The first uses a four layer acoustic model, and corrupts the data with random noise and dead traces. The second example uses a single shot gather from the Sigsbee 2a model. Not surprisingly, the derivation of the shot-profile wave-field modelling and migration operators is similar to the derivation of the analogous source-receiver operators presented in Chapter 2. The main difference lies in the formulation of the Green's functions, and this, in turn, leads to the shot-profile parameterization of the scattering potential, and the shot-profile operators. A further consequence of the shot-profile parameterization is that the algorithm can be applied one shot at at time. This is in contrast to the source-receiver geometry which requires a rectilinear grid such that all four dimensions of the survey acquisition are considered simultaneously (see Section 2.7).

5.2 Shot-profile wave-field modelling

As was done in Chapter 2 for the source-receiver parameterization of the scattering potential, we outline the construction of the wave-field modelling (forward) operator, but this time for the shot-profile parameterization of the scattering potential. In particular, we use the Born approximation to the wave-field in which 1) we note the form of the Green's function in space and frequency for a constant reference velocity, 2) we modify the Green's function using the split-step approximation, and 3) use the Green's function in a Gazdag like algorithm in analogy to a similar derivation in Chapter 2. The end result of our derivation is the shot-profile wave-field modelling operator, and which in the literature is often called shot-profile *de-migration*.

The Born approximation to the wave-field was given in equation 2.2. In particular, we wrote $\psi = \psi_d + \psi_s$ where ψ is the wave-field, $\psi_d = f(\omega)G_0$ is the direct wave-field, and ψ_s is the scattered wave-field (equation 2.4). Integral to the Born approximation is the Green's function G_0, and to find a shot-profile parameterization of the scattering potential, we use a Green's function represented in space and frequency. In particular, we use the Green's function in equation 2.6, but after taking its two-dimensional Fourier transform over the geophone wave-number dimensions $\mathbf{k}_{gx}$. In addition, we replace $\mathbf{x}_g$ with $\mathbf{x}$ and $\mathbf{k}_{gx}$ with $\mathbf{k}_x$ so that,

$$G_0(\mathbf{x}, z|\mathbf{x}', z'; \omega) = \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} \left(-\frac{1}{i4k_z}\right) e^{-i\mathbf{k}_x\cdot(\mathbf{x}'-\mathbf{x})} e^{ik_z|z-z'|} d\mathbf{k}_x. \tag{5.1}$$

The change of notation reminds us that in the derivation of the shot-profile wave-field modelling operator, the parameterization of the scattering potential is independent of the survey geometry. For the sake of completeness, we write down the corresponding dispersion relation,

$$k_z = sgn(\omega)\sqrt{\frac{\omega^2}{c_0^2} - \mathbf{k}_x \cdot \mathbf{k}_x},$$

which is analogous to the dispersion relations in equation 2.8.

To allow for the reference velocity model c_0 to vary in its lateral dimension so that $c_0 = c_0(\mathbf{x})$, we again apply the split-step approximation in equation 2.17 to the Green's function in equation 5.1 so that it becomes,

$$G_0(\mathbf{x}, z|\mathbf{x}', z'; \omega) = e^{i\omega(c_0^{-1}(\mathbf{x})-c_1^{-1})|z-z'|} \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} \left(-\frac{1}{i4k_z}\right) e^{-i\mathbf{k}_x\cdot(\mathbf{x}'-\mathbf{x})} e^{ik_z|z-z'|} d\mathbf{k}_x, \tag{5.2}$$

where,

$$k_z = sgn(\omega)\sqrt{\frac{\omega^2}{c_1^2} - \mathbf{k}_x \cdot \mathbf{k}_x}, \tag{5.3}$$

and c_1^{-1} is the mean of $c_0^{-1}(\mathbf{x})$.

To allow the migration velocity model c_0 to vary in both $\mathbf{x}$ and z we employ, in conjunction with equation 5.2, the wave-field propagator described in Gazdag (1978) and Section 2.4.1 of this thesis. In particular, the reference velocity $c_0(\mathbf{x}, z)$ is partitioned into n_z domains (layers), and the contribution to the total scattered wave-field from the l^{th} layer is given by $\psi_{s(l)}$. The total total scattered wave-field is given by the sum over the terms $\psi_{s(l)}$, $l = 1 \dots n_z$ (see equation 2.10). Recall that within each layer there is a corresponding Green's function $G_{0(l)}$ (equation 5.2) built using its constant (with respect to depth) reference velocity $c_{0(l)}(\mathbf{x})$ that, in turn, approximates $c_{0(l)}(\mathbf{x}, z)$ for $z_{l-1} \leq z < z_l$. Moreover, to accommodate the split-step approximation in D_l, we let $c_{1(l)}^{-1}$ be the mean of $c_{0(l)}^{-1}(\mathbf{x})$. As in Chapter 2, we derive expressions for $\psi_{s(1)}$ and $\psi_{s(2)}$, using these to infer a general form for $\psi_{s(l)}$.

In the first layer D_1, we define a measurement surface z_0 so that $z_g = z_s = z_0$, and substitute

$G_{0(1)}$ into equation 2.4 finding (see Appendix D),

$$\psi_{s(1)}(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega)$$
$$= \left(\frac{\omega}{c_{1(1)}}\right)^2 \left(\frac{1}{2\pi}\right)^4 \int_{z_0}^{z_1} u^1_{s(1)} \mathcal{F}^*_g u^1_{p(1)} \mathcal{F}_g \left[u^1_{s(1)} \mathcal{F}^*_g u^1_{p(1)} g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega)\right] \alpha(\mathbf{x}_g, z')dz', \tag{5.4}$$

where $\mathcal{F}_g$ denotes the unnormalized two-dimensional Fourier transform over lateral coordinates $\mathbf{x}_g$, and $\mathcal{F}^*_g$ its adjoint operation, so that the corresponding inverse Fourier transform is $(2\pi)^{-2}\mathcal{F}^*_g$. The functions $u^1_{p(1)}$ and $u^1_{s(1)}$ in equation 5.4 are analogous to $u_{p(1)}$ and $u_{s(1)}$ in, respectively, equations 2.12 and 2.21 which were used in the source-receiver wave-field modelling and migration operators (they can be thought of as their single-square-root counter-parts). In particular, $u^1_{p(1)}$ is,

$$u^1_{p(l)}(\mathbf{k}_{gx}, z'; \omega) = -\frac{e^{ik_{gz(l)}(z'-z_{l-1})}}{i4k_{gz(l)}}, \tag{5.5}$$

and $u^1_{s(1)}$ is,

$$u^1_{s(l)}(\mathbf{x}_g, z'; \omega) = e^{i\omega(c^{-1}_{0(l)}(\mathbf{x}_g) - c^{-1}_{1(l)})(z'-z_{l-1})}, \tag{5.6}$$

both for $l = 1$. The function,

$$g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega) = f(\omega)e^{-i\mathbf{k}_{gx}\cdot\mathbf{x}_s}, \tag{5.7}$$

is the synthetic source used in shot-profile migration algorithms, and $\mathbf{k}_{gz(l)}$ is short-hand for $\mathbf{k}_{gz}(c_{1(l)})$ given by the dispersion relation in equation 5.3. To simplify the notation in this chapter, we will drop the 1 in the superscripts of $u^1_{p(l)}$ and $u^1_{s(l)}$, while remembering that we are using the shot-profile rather than the source-receiver operators. Equations 5.5 and 5.6 are constructed from $G_{0(l)}$, governing the propagation of energy in D_l. The detailed relation between $G_{0(l)}$, $u_{p(l)}$, $u_{s(l)}$ and g are presented in Appendix D, and the physical meaning of the Green's functions in modelling the wave-field was given in Chapter 2. In addition, we made a simplifying approximation for one of the amplitude terms in equation 2.4 before arriving at equation 5.4. Namely, we let,

$$(\omega/c_{0(l)}(\mathbf{x}'))^2\alpha(\mathbf{x}', z') \approx (\omega/c_{1(l)})^2\alpha(\mathbf{x}', z'), \tag{5.8}$$

and continue using this simplification in the remaining derivations.

The construction of $\psi_{s(2)}$ follows the same three step procedure outlined in Chapter 2, but using the Green's function from equation 5.2, and gives the following contribution to the scattered wave-field from the second layer (see Appendix D),

$$\psi_{s(2)}(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega) = \left(\frac{\omega}{c_{1(2)}}\right)^2 \left(\frac{1}{2\pi}\right)^8 \int_{z_1}^{z_2} u_{s(1)}\mathcal{F}_g^* u_{p(1)}\mathcal{F}_g u_{s(2)}\mathcal{F}_g^* u_{p(2)}\mathcal{F}_g \\ \times \left[u_{s(2)}\mathcal{F}_g^* u_{p(2)}\mathcal{F}_g u_{s(1)}\mathcal{F}_g^* u_{p(1)} g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega)\right] \alpha(\mathbf{x}_g, z')dz', \tag{5.9}$$

where $u_{p(2)}$ and $u_{s(2)}$ are given by, respectively, equations 5.5 and 5.6 with $l = 2$. Generalizing to the l^{th} layer D_l, we find,

$$\psi_{s(l)}(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega) \\ = \left(\frac{\omega}{c_{1(l)}}\right)^2 \left(\frac{1}{2\pi}\right)^{4l} \int_{z_{l-1}}^{z_l} u_{s(1)}\mathcal{F}_g^* u_{p(1)}\mathcal{F}_g u_{s(2)}\mathcal{F}_g^* u_{p(2)}\mathcal{F}_g \cdots u_{s(l)}\mathcal{F}_g^* u_{p(l)}\mathcal{F}_g \\ \times \left[u_{s(l)}\mathcal{F}_g^* u_{p(l)}\mathcal{F}_g u_{s(l-1)}\mathcal{F}_g^* u_{p(l-1)}\mathcal{F}_g \cdots u_{s(1)}\mathcal{F}_g^* u_{p(1)} g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega)\right] \alpha(\mathbf{x}_g, z')dz', \tag{5.10}$$

where, again, $u_{p(l)}$ and $u_{s(l)}$ are given by equations 5.5 and 5.6.

The wave-field modelling operator is, then, defined by equations 2.10 and 5.10. To develop an efficient implementation, we must analyze the sum in equation 2.10. We begin by adding $\psi_{s(1)}$ and $\psi_{s(2)}$, approximating the integrals with Riemann sums, so that,

$$\psi_{s(1)} + \psi_{s(2)} \\ = \Delta z u_{s(1)}\mathcal{F}_g^* u_{p(1)}\mathcal{F}_g\{(2\pi)^{-4}(\omega/c_{1(1)})^2[u_{s(1)}\mathcal{F}_g^* u_{p(1)} g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega)]\alpha(\mathbf{x}_g, z_1) \\ + (2\pi)^{-8}(\omega/c_{1(2)})^2 u_{s(2)}\mathcal{F}_g^* u_{p(2)}\mathcal{F}_g[u_{s(2)}\mathcal{F}_g^* u_{p(2)}\mathcal{F}_g u_{s(1)}\mathcal{F}_g^* u_{p(1)} g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega)]\alpha(\mathbf{x}_g, z_2)\}, \tag{5.11}$$

where we assume a constant thickness Δz for each layer D_l. In equation 5.11, we arranged terms such that an iteration and a recursion can be recognized. In particular, generalizing to n_z layers, we recognize in equation 5.11 an iterative method for downward continuing the source side wave-field into the earth,

$$v_{s(1)}(\mathbf{x}_g, \omega; \mathbf{x}_s) = u_{s(1)}\mathcal{F}_g^* u_{p(1)} g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega) \\ v_{s(l)}(\mathbf{x}_g, \omega; \mathbf{x}_s) = u_{s(l)}\mathcal{F}_g^* u_{p(l)}\mathcal{F}_g v_{s(l-1)} \quad , \quad l = 2 \ldots n_z. \tag{5.12}$$

We recognize a recursion ($l = n_z \ldots 1$) for constructing the wave-field at the measurement surface,

$$\psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega) = \Delta z v_1 \\ v_l(\mathbf{x}_g, \omega; \mathbf{x}_s) = u_{s(l)}\mathcal{F}_g^* u_{p(l)}\mathcal{F}_g \left(v_{s(l)}(2\pi)^{-4l}(\omega/c_{1(l)})^2\alpha(\mathbf{x}_g, z_l) + v_{l+1}\right), \tag{5.13}$$

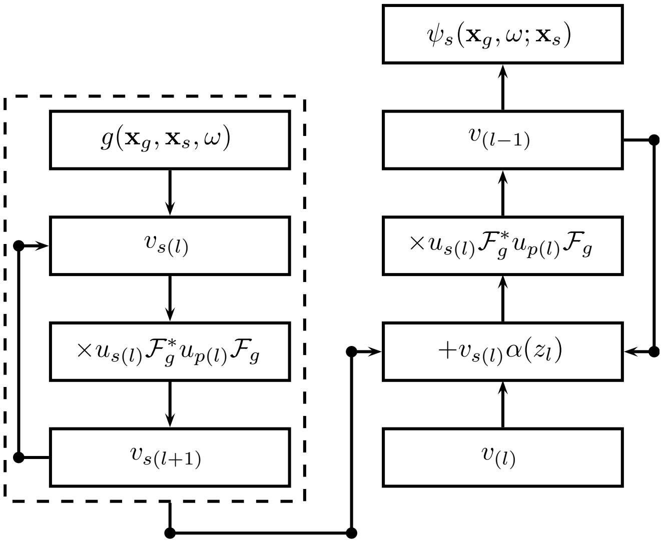
$\psi_s(\mathbf{x}_g, \omega; \mathbf{x}_s)$
$g(\mathbf{x}_g, \mathbf{x}_s, \omega)$
$v_{(l-1)}$
$v_{s(l)}$
$\times u_{s(l)}\mathcal{F}_g^* u_{p(l)}\mathcal{F}_g$
$\times u_{s(l)}\mathcal{F}_g^* u_{p(l)}\mathcal{F}_g$
$+v_{s(l)}\alpha(z_l)$
$v_{s(l+1)}$
$v_{(l)}$

Figure 5.1: We illustrate the split-step shot-profile wave-field modelling operator in equations 5.12 and 5.13 using a flow-chart. The left-hand side is the iteration in equation 5.12, and the right-hand side is the recursion in equation 5.13.

where $v_{n_z+1} = 0$. Hence, as noted in Rickett (2001), the algorithm requires two passes through depth, first to compute and store $v_{s(l)}$ in equation 5.12, and second to compute the recursion in equation 5.13.

Equations 5.12 and 5.13 constitute an algorithm for wave-field modelling which is, in other words, the forward operator for shot-profile least-squares migration. Because we have parameterized the equations for the shot-profile geometry rather than the source-receiver geometry, we can model each shot gather independently. Figure 5.1 provides a flow-chart for the split-step shot-profile wave-field modelling operator. The left-hand side of the flow-chart illustrates the iteration in equation 5.12 which propagates the source side wave-field into the earth, saving it at each depth step (denoted by the index l). The right hand-side of the flow-chart denotes the recursion in equation 5.13, showing the construction of the wave-field. The flow-chart depicts a recursion that begins at depth and terminates at the measurement surface.

5.3 Shot-profile wave-field migration

Equations 2.10 and 5.10 are wave-equation shot-profile wave-field modelling using split-step operators. For least-squares shot-profile migration, they are the forward operator.

In this section, we compute the corresponding adjoint operator, enabling a least-squares formulation.

To proceed, we re-write the forward operator (equations 2.10 and 5.10) so that a discretized Fredholm integral equation of the first kind is recognized. In particular, we have for lateral shot location $\mathbf{x}_s$,

$$\psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega_j) = \Delta z \sum_l u(\omega_j, z_l; \mathbf{x}_g, \mathbf{x}_s)\alpha(\mathbf{x}_g, z_l; \mathbf{x}_s), \tag{5.14}$$

where ω_j is some realization of ω, and,

$$\begin{aligned} u(\omega_j, z_l; \mathbf{x}_g, \mathbf{x}_s) &= (2\pi)^{-4l}(\omega/c_{1(l)})^2 \\ &\times u_{s(1)}\mathcal{F}_g^* u_{p(1)}\mathcal{F}_g \cdots u_{s(l)}\mathcal{F}_g^* u_{p(l)}\mathcal{F}_g \left[u_{s(l)}\mathcal{F}_g^* u_{p(l)}\mathcal{F}_g \cdots u_{s(1)}\mathcal{F}_g^* u_{p(1)} g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega_j)\right]. \end{aligned} \tag{5.15}$$

With de-migration cast into the form of equations 5.14 and 5.15, we write down their adjoint (e.g. Hansen, 1998),

$$\alpha^\dagger(\mathbf{x}_g, z_l; \mathbf{x}_s) = \Delta z \sum_j u^*(\omega_j, z_l; \mathbf{x}_g, \mathbf{x}_s)\psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega_j), \tag{5.16}$$

where, as before, $(\cdot)^*$ denotes the adjoint, so that,

$$\begin{aligned} u^*(\omega_j, z_l; \mathbf{x}_g, \mathbf{x}_s) &= (2\pi)^{-4l}(\omega_j/c_{1(l)})^2 \\ &\times \left[u_{s(l)}^*\mathcal{F}_g^* u_{p(l)}^*\mathcal{F}_g \cdots u_{s(1)}^*\mathcal{F}_g^* u_{p(1)}^* g^*(\mathbf{k}_{gx}, \mathbf{x}_s, \omega_j)\right] u_{s(l)}^*\mathcal{F}_g^* u_{p(l)}^*\mathcal{F}_g \cdots u_{s(1)}^*\mathcal{F}_g^* u_{p(1)}^*\mathcal{F}_g, \end{aligned} \tag{5.17}$$

and we have used,

$$(\mathcal{F}_g^* u_{p(l)}\mathcal{F}_g)^* = \mathcal{F}_g^* u_{p(l)}^*\mathcal{F}_g.$$

In the adjoint (equations 5.16 and 5.17), we recognize an example of the often studied shot-profile migration algorithm (e.g. Sava and Hill, 2009). In particular, there are two iterations for the downward continuation of the source and receiver side wave-fields, as well as the imaging condition (the sum over frequency indices j in equation 5.16). In particular, the iteration that downward continues the source side wave-field from depth index $l-1$ to depth index l is,

$$\begin{aligned} v_{s(1)}^*(\mathbf{x}_g, \omega_j; \mathbf{x}_s) &= u_{s(1)}^*\mathcal{F}_g^* u_{p(1)}^* g^*(\mathbf{k}_{gx}, \mathbf{x}_s, \omega_j) \\ v_{s(l)}^*(\mathbf{x}_g, \omega_j; \mathbf{x}_s) &= u_{s(l)}^*\mathcal{F}_g^* u_{p(l)}^*\mathcal{F}_g v_{s(l-1)}^* \quad , \quad l = 2\ldots n_z. \end{aligned} \tag{5.18}$$

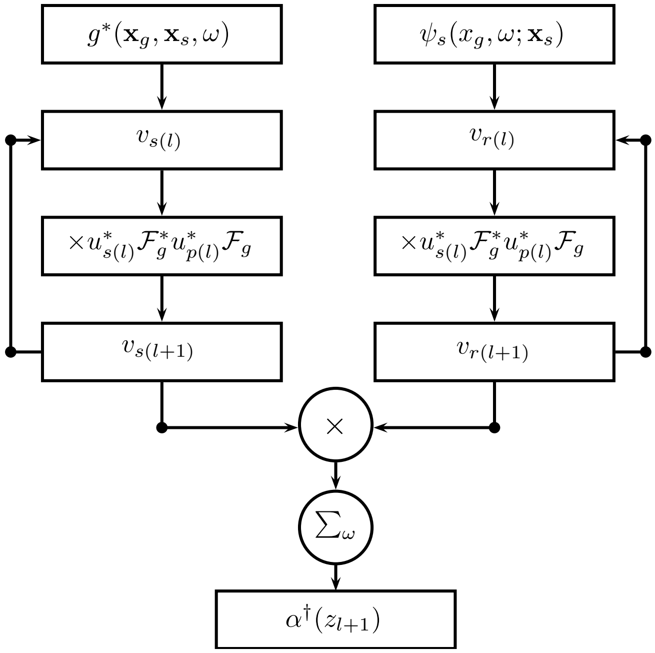
$g^*(\mathbf{x}_g, \mathbf{x}_s, \omega)$
$\psi_s(x_g, \omega; \mathbf{x}_s)$
$v_{s(l)}$
$v_{r(l)}$
$\times u^*_{s(l)}\mathcal{F}^*_g u^*_{p(l)}\mathcal{F}_g$
$\times u^*_{s(l)}\mathcal{F}^*_g u^*_{p(l)}\mathcal{F}_g$
$v_{s(l+1)}$
$v_{r(l+1)}$
$\times$
$\sum_\omega$
$\alpha^\dagger(z_{l+1})$

Figure 5.2: We illustrate the shot-profile split-step migration operator using a flow-chart. The left-hand and right-hand sides of the flow-chart depict the downward continuations of, respectively, the source- and receiver-side wave-fields (equations 5.18 and 5.19). The imaging condition in equation 5.20 is depicted in the × and $\sum_\omega$ entries of the flow-chart.

Second, the downward continuation of the receiver side wave-field is,

$$\begin{aligned} v^*_{r(1)}(\mathbf{x}_g, \omega_j; \mathbf{x}_s) &= u^*_{s(1)}\mathcal{F}^*_g u^*_{p(1)}\mathcal{F}_g \psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega) \\ v^*_{r(l)}(\mathbf{x}_g, \omega_j; \mathbf{x}_s) &= u^*_{s(l)}\mathcal{F}^*_g u^*_{p(l)}\mathcal{F}_g v^*_{r(l-1)} \quad , \quad l = 2 \ldots n_z, \end{aligned} \tag{5.19}$$

so that equation 5.16 becomes,

$$\alpha^\dagger(\mathbf{x}_g, z_l; \mathbf{x}_s) = \Delta z \sum_j \left(\frac{\omega}{c_{1(l)}}\right)^2 \left(\frac{1}{2\pi}\right)^{4l} v^*_{s(l)}(\mathbf{x}_g, \omega_j; \mathbf{x}_s) v^*_{r(l)}(\mathbf{x}_g, \omega_j; \mathbf{x}_s). \tag{5.20}$$

Equations 5.16 and 5.17 constitute split-step shot-profile migration, and for efficiency is implemented using the iterations (downward continuations) in equations 5.18 and 5.19, and the imaging condition in equation 5.20. A schematic of the algorithm is shown in the flow-chart in Figure 5.2.

5.4 Shot-profile least-squares migration

Following the same methodology as previous chapters (Chapters 3 and 4), we let $\mathbf{d}$ be a vector of length M realized from $\psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega)$, and where,

$$M = \sum_{q=1}^{n_s} n_\omega n_{g(q)},$$

n_s are the number of shot gathers, $n_{g(q)}$ are the number of geophones in the q^{th} shot gather, and n_ω are the number of frequencies. Likewise, we let $\mathbf{m}$ be a vector of length N realized from $\alpha(\mathbf{x}_g, z; \mathbf{x}_s)$, where,

$$N = \sum_{q=1}^{n_s} n_{g(q)} n_z,$$

and, as before, n_z are the number of depths in the migrated image gathers. Then we let $\mathbf{A}$ be the $M \times N$ matrix built from the shot-profile wave-field modelling operator defined by equations 5.12 and 5.13, and $\mathbf{A}^H$ its adjoint (migration operator) defined by equations 5.16 and 5.17. Then $\mathbf{A}$ maps from $\mathbf{m}$ to $\mathbf{d}$, and to find optimal migrated images, we solve the set of least-squares normal equations,

$$(\mathbf{A}^H \mathbf{W}_d^H \mathbf{W}_d \mathbf{A} + \mu \mathbf{I})\mathbf{m} = \mathbf{A}^H \mathbf{W}_d^H \mathbf{W}_d \mathbf{d}, \tag{5.21}$$

for $\mathbf{m}$. The least-squares normal equations can be found from a suitable interpretation of Bayes formula. We neglect the derivation as it follows analogously to the derivation of the normal equations in Chapter 4, except where, here, we use a Gaussian distribution (with independent and identically distributed random variables) rather than the Cauchy distribution for the model prior. In equation 5.21, $\mathbf{W}_d$ is a data weighting matrix derived from an estimate of the noise covariance matrix (see Chapter 4), and in this chapter is used to allow for incomplete data. Finally, $\mu = \sigma_n^2/\sigma_m^2$ is a trade-off parameter weighting the relative importance of the data likelihood and model prior, and where σ_n^2 is the variance of the noise in the observed data, and σ_m^2 is the variance for the model prior (Gaussian probability density functions; again we refer the interested reader to Chapter 4). We solve equation 5.21 by the conjugate gradient method, and implicit construction of the matrices (see Chapter 3).

In building the least-squares system of equations, we have made some choices. For example, we assume that equation 5.21 is for all shot gathers. That is, $\mathbf{d}$ contains all shot gathers in the survey, and we solve equation 5.21 once. Alternatively, one could let $\mathbf{d}$ contain one shot gather from the survey, and solve equation 5.21 once for each shot gather. However, this choice should have no consequence on the character of the final result. Additionally, we made a choice to represent α in the pre-stack domain, and the alternative post-stack representation is discussed in Section 5.5. We are tasked with making these choices exactly because of the shot-profile parameterization of the scattering potential. If, on the other hand, we choose a source-receiver parameterization of the scattering potential (Chapter 2), then data are required on some rectilinear grid in shot and receiver (or midpoint and offset) coordinates. This is to accommodate the four dimensional Fourier transforms required by the source-receiver split-step (de-)migration algorithm. Moreover, the model space becomes less physical, as the offset coordinate does not have a one-to-one correspondence with the earth model. Instead, it must be interpreted in terms of the source-receiver imaging condition (Chapter 4).

The matrices in the least-squares normal equations (5.21) are large. For the Sigsbee 2a example (Section 5.7) with 241 shots, $\mathbf{A}$ has approximately 200 million rows and 140 million columns. This makes its explicit computation and storage infeasible, as was also illustrated in Chapter 3. Instead, the construction is implicit so that, for example, given some $\mathbf{m}$, we compute $\mathbf{d} = \mathbf{Am}$ using an operator that implements equations 5.12 and 5.13. Likewise, given some $\mathbf{d}$, we compute $\mathbf{m}^\dagger = \mathbf{A}^H\mathbf{d}$ using an operator that implements equations 5.19 and 5.20. We verify that our operators make a forward/adjoint pair using the dot-product test (e.g. Claerbout, 1992). The conjugate gradient algorithm allows us to solve the least-squares normal equations with the forward/adjoint operators, and without the explicit storage of the matrices, or their inverse matrices.

5.5 Joint shot-profile least-squares migration

In Section 5.4, we choose to represent α in the pre-stack domain. One could alter the forward operator $\mathbf{A}$ to include the sum over shots. This would shrink the size of the model space, perhaps producing a more efficient inversion, but at the cost of a more restrictive model space (fewer degrees of freedom). In Chapter 6 we introduce a data reconstruction algorithm that requires a post-stack representation of α. In this chapter, we experiment with the post-stack representation of α, described in this section, for its effect on the migrated image.

We can accommodate this post-stack representation of α by altering the definitions in Section 5.4. In particular, we redefine $\mathbf{m}$ and N such that $\mathbf{m}$ becomes a vector of length N realized from,

$$\alpha(\mathbf{x}_g, z) = \sum_{q=1}^{n_s} \alpha(\mathbf{x}_g, z; \mathbf{x}_{s(q)}),$$

where $q = 1 \ldots n_s$, $N = n_z n_g$, and n_g are the number of geophones for any given shot. Please note that this requires a common parameterization of the geophone dimensions for all shots. It also requires us to redefine the matrix $\mathbf{A}$. In particular we let $\mathbf{A}_q$ be the $M \times N$ matrix built from the shot-profile wave-field modelling operator defined by equations 5.12 and 5.13, and $\mathbf{A}_q^H$ its adjoint (migration operator) defined by equations 5.16 and 5.17, all for a single instance of $\mathbf{x}_s$: namely, $\mathbf{x}_s = \mathbf{x}_{s(q)}$. Then, the matrix $\mathbf{A}$ is defined such that,

$$\mathbf{A}^H = \left[\begin{array}{cccc} \mathbf{A}_1^H & \mathbf{A}_2^H & \cdots & \mathbf{A}_{n_s}^H \end{array} \right],$$

and, as before, $\mathbf{A}$ maps from $\mathbf{m}$ to $\mathbf{d}$. The definition of $\mathbf{d}$ remains unchanged from Section 5.4. In the following sections, we show examples for which we solve for both the pre- and post-stack least-squares representations of α.

5.6 Example: 4 layer acoustic model

For our first example, we consider the synthetic data in Figure 5.3b. The data is generated using an acoustic finite difference code with absorbing boundary conditions (including no free-surface), and the acoustic four layer model in Figure 5.3a. We collect shots spaced, laterally, every 20 metres, and at a depth of 150 metres. In total there are 71 shots, and 281 geophones per shot. The geophones are spaced every 5 metres, and are at a depth of 150 metres. We add Gaussian random noise to the data so that the signal to noise ratio is 3, and, at random, we replace sixty percent of the traces with dead (zero) traces. Due to the missing traces, we let $\mathbf{W}_d$ in equation 5.21 be a diagonal matrix such that its i^{th} diagonal element is,

$$[\mathbf{W}_d]_{ii} = \begin{cases} 1 & , \quad i \notin \mathcal{I}_d \\ 0 & , \quad i \in \mathcal{I}_d, \end{cases} \tag{5.22}$$

where $\mathcal{I}_d$ is the set of indices corresponding to dead traces in $\mathbf{d}$.

We apply the adjoint to each common shot gather, giving common shot image gathers, and then sum over the shot dimension, giving the result in Figure 5.3c. Likewise, we apply least-squares migration to each common shot gather using the pre-stack parameterization of the scattering potential (Section 5.4), and sum to give Figure 5.3d. The differences between Figures 5.3c and d are (in travel-time) small, but there are differences in amplitude that, arguably, make the reflectors more visible in the inverse, as compared to the adjoint. Finally, we apply least-squares migration using the post-stack parameterization of the scattering potential (Section 5.5) to give the result in Figure 5.3e. In this later case, the improvement of the least-squares migration result over the migration (adjoint) result are more obvious. In particular, we see that the quality of the reflectors are enhanced.

In Figure 5.4, we apply the forward (wave-field modelling) operator to the migrated data found by inversion, giving Figure 5.4c for the pre-stack parameterization of the scattering

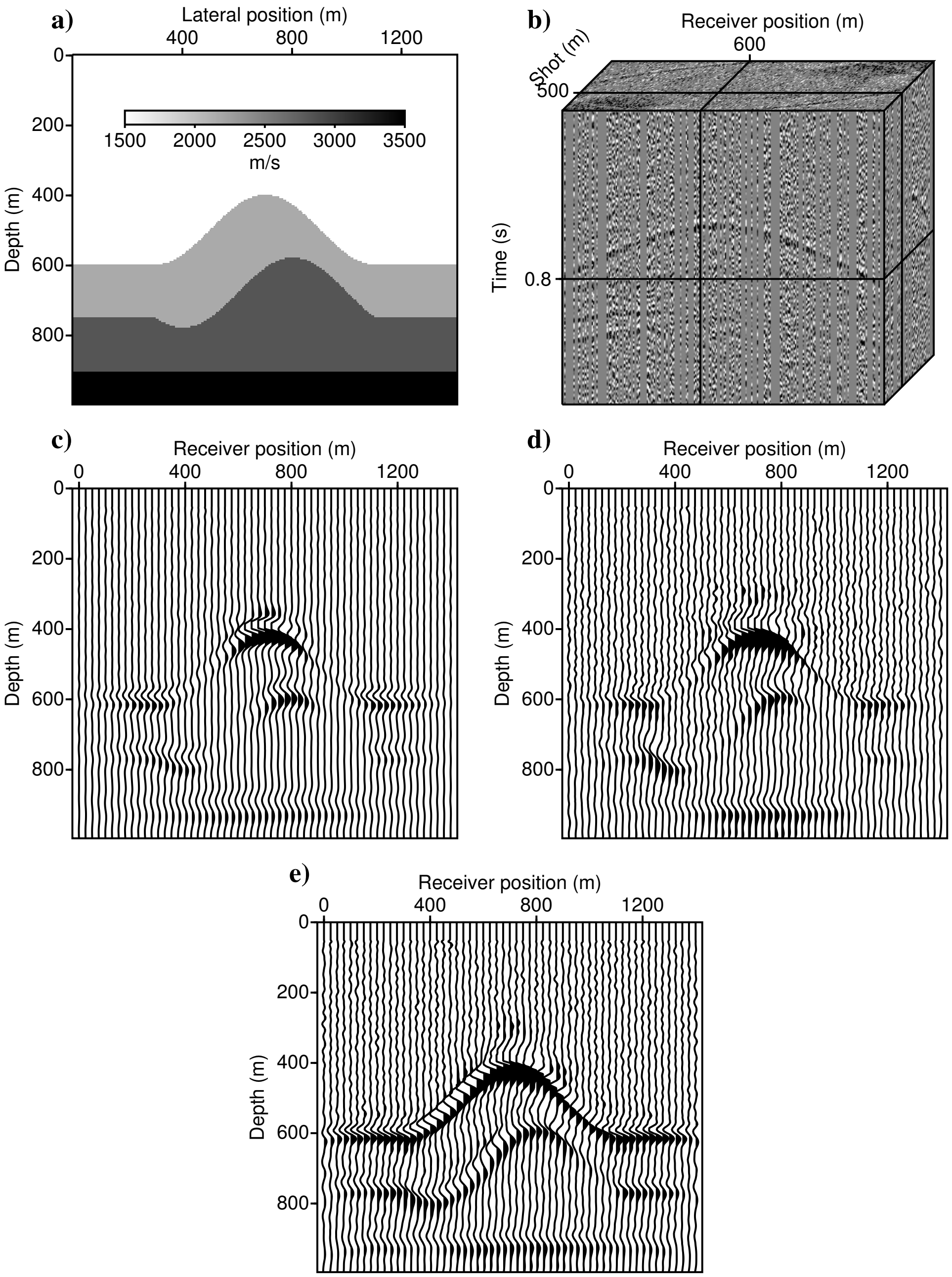

Figure 5.3: Four layer acoustic model example showing, a) the velocity model, b) the data, c) the adjoint, the inverse computed using d) the pre-stack parameterization of α, and e) the post-stack parameterization of α. (both c) and d) are shown after summing over all shots, and c) through e) are all plotted using a 50% clip).

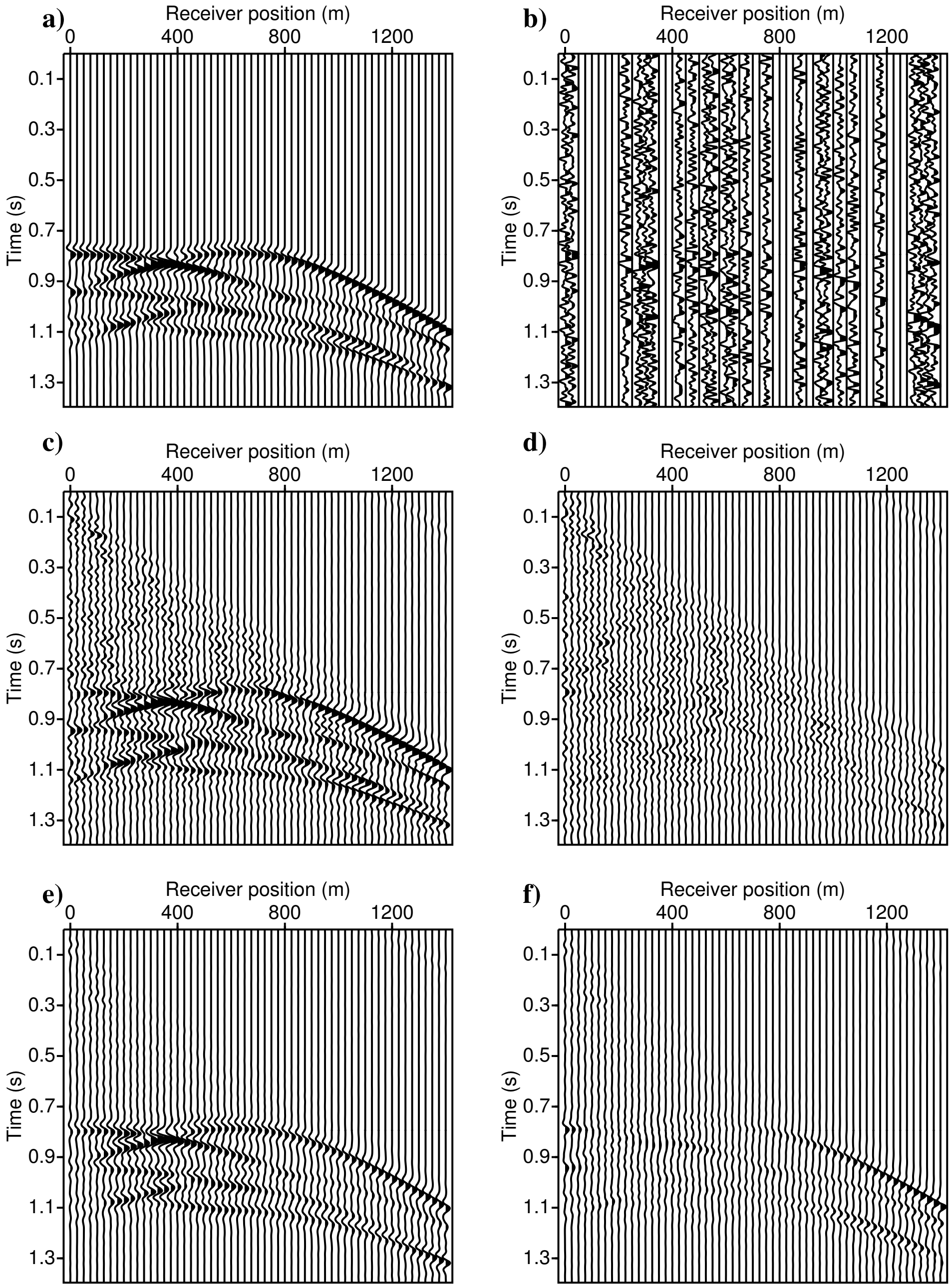

Figure 5.4: Four layer acoustic model example showing for $x_s = 0m$, a) the clean shot gather, b) the shot gather, the interpolated shot gather using least-squares shot-profile migration with c) the pre-stack and e) post-stack parameterization of the scattering potential. In d), we show the difference between a) and c), and in f) we show the difference between e) and a).

potential (Section 5.4) and Figure 5.4e for the post-stack parameterization of the scattering potential (Section 5.5). Figure 5.4b is the corresponding shot gather in the input data, and Figure 5.4a is its noise-free counter-part. Finally, Figure 5.4d is the difference between the reconstructed data and the noise-free data using the pre-stack parameterization of the scattering potential, and Figure 5.4f is the same, but for the post-stack parameterization of the scattering potential. Given that the pre-stack parameterization gives more degrees of freedom to the model then the post-stack parameterization, it is reasonable that the residual in the pre-stack (Figure 5.4d) case is less than the residual in the post-stack case (Figure 5.4f).

5.7 Example: Sigsbee 2a data

For our second example, we consider the Sigsbee 2a model from the SMAART JV project (Bergsma, 2007). We use finite-difference data obtained from the Madagascar project (Irons, 2007). The portion of the model that we consider is shown in Figure 5.5a, and Figure 5.5b is the velocity model used for migration. Figure 5.5c plots the one-and-only shot gather that we use in this example. Since we are using only one shot gather for this example, the pre-stack and post-stack parameterizations of the scattering potential are equivalent. The shot is located at 3.33km, and there are 348 receivers. The near-offset receiver is coincident with the shot, and the far-offset receiver is located at 11.26km (offset 7.93km from the shot). The shots and receivers are placed at a depth of 7.62m. The aperture for the migration runs from the shot location out to 14.3km (offset 10.97km from the shot). The results of applying migration and least-squares migration to the shot gather in Figure 5.5c are shown in Figures 5.6 and 5.7. In Figures 5.6a and 5.6b we show, respectively, migration and least-squares migration, and in Figure 5.6c we show the true reflectivity model, which is representative of the scattering potential α (for display purposes, we apply a low-pass filter to the reflectivity). Figure 5.7 shows the same information as Figure 5.6 but for a shorter time window.

In the inversion, we note that the split-step wave-field propagator makes heavy use of the Fourier transform which tends to cause edge effects. We allow for these by including a taper in the data weights $\mathbf{W}_d$. This means that we do not require the least-squares migration algorithm to fit far offset traces. The effect of $\mathbf{W}_d$ is visible when reconstructing the data gather from the least-squares common shot image gather, and is illustrated in Figure 5.8. Figure 5.8a plots the shot gather (repeated from Figure 5.5c), and Figure 5.8b plots the shot gather constructed using the common shot image gather in Figure 5.6b, and the wave-field modelling operator given by equations 5.12 and 5.13.

Due to the limited data (single shot gather), the migrated images are, in places, of poor quality, especially where the offset from the source is large. We observe that the inverse compared to the adjoint has, in places, improved the image. In particular, artifacts have been reduced, and the point diffractors, especially those at 5km depth and far offset, are better resolved. We include the single-shot example because it illustrates the difference between migration and least-squares migration under limited data. In this case the limitation is, admittedly, extreme (a single shot gather), and is not adequate for illumination of the entire sub-surface. Of course, when more shots are used to generate the image, the resolution improves, and we show this in Figure 5.9 where 241 shot gathers are used. The first shot is the same as the single-shot experiment, located at 3.33km, and with the same geophone configuration. The remaining shots are spaced every 45.72m, with the last shot located at 14.3km. Each shot has 348 receivers, and the survey follows a towed-streamer geometry. Figures 5.9a and 5.9b are the sum over the 241 common shot image gathers which are, in turn, computed using, respectively, migration and least-squares migration using the pre-stack geometry for the scattering potential. In Figure 5.9c, we plot the least-squares migration result for the post-stack parameterization of the scattering potential. We note that Figures 5.9a through 5.9c are plotted using the same percentile clip. While least-squares migration with the pre-stack geometry for the scattering potential shows some uplift in amplitude compared to migration, the differences concerning reflector locations and point diffractor resolution are, perhaps, less tangible than the single shot example. On-the-other-hand, the least-squares migration result using the post-stack geometry for the scattering potential gives some reasonable improvements. For example, we note the improved delineation of the point diffractors located at approximately 7.8km depth in Figure 5.9c as compared to Figure 5.9a and 5.9b.

5.8 Summary

We show, in detail, the derivation of shot-profile wave-equation migration and wave-field modelling operators from the Born approximation, and split-step wave-field modelling. We derived the forward (wave-field modelling) operator, and its adjoint (migration). From this analysis we recognize the usual shot-profile migration algorithm involving the downward continuation of the source and receiver side wave-fields, along with an imaging condition.

Given the forward and adjoint operators for wave-equation shot-profile (de-)migration, we built a shot-profile wave-equation least-squares migration algorithm using pre- and post-stack parameterizations of the scattering potential. This differs from previously published source-receiver least-squares migration algorithms (and those used in Chapters 3 and 4), allowing for the (de-)migration operators to be applied one shot at a time, as well as giving an alternative parameterization of the migrated image gathers.

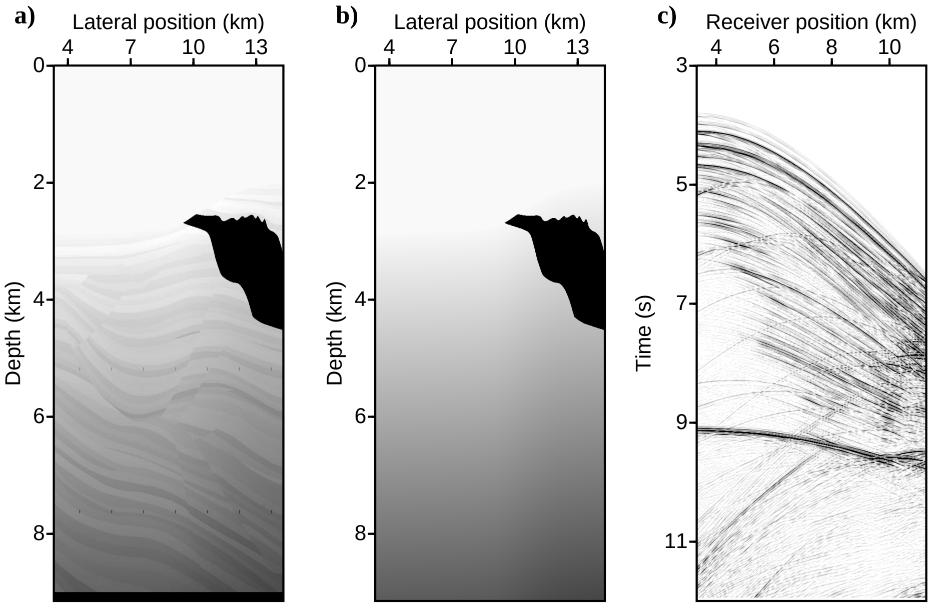

Figure 5.5: Sigsbee 2a example: a) velocity model, b) migration velocity model, and c) data (single shot gather).

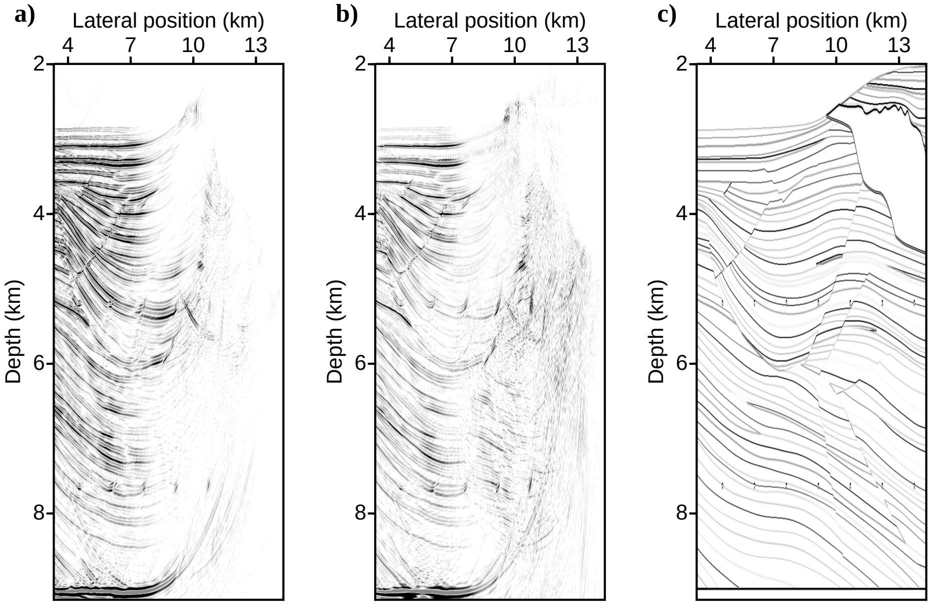

Figure 5.6: Sigsbee 2a example, single shot: a) the migration (adjoint), b) the least-squares migration (inverse), and c) the true reflectivity.

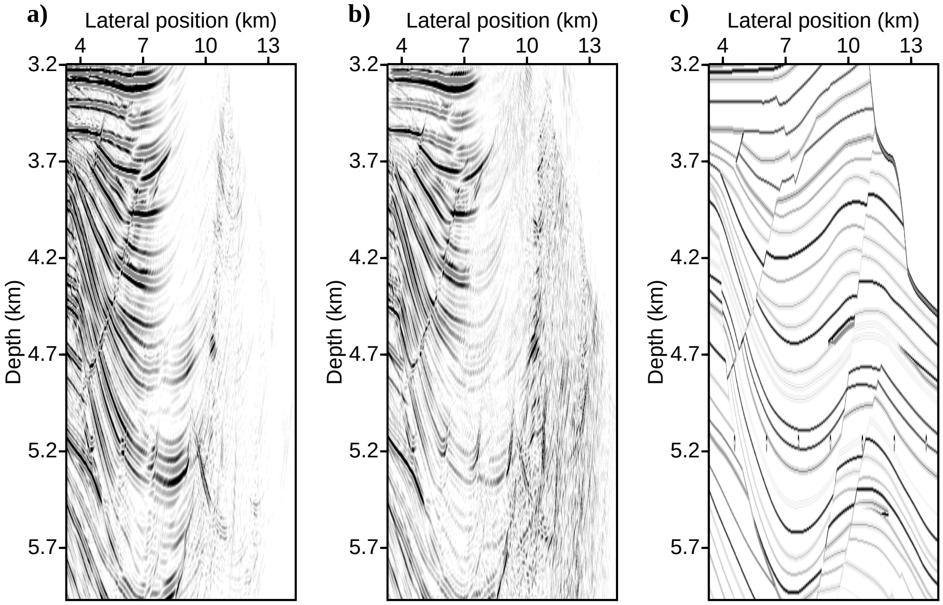

Figure 5.7: Sigsbee 2a example, single shot: for a time window we show, a) the migration (adjoint), b) the least-squares migration (inverse), and c) the true reflectivity.

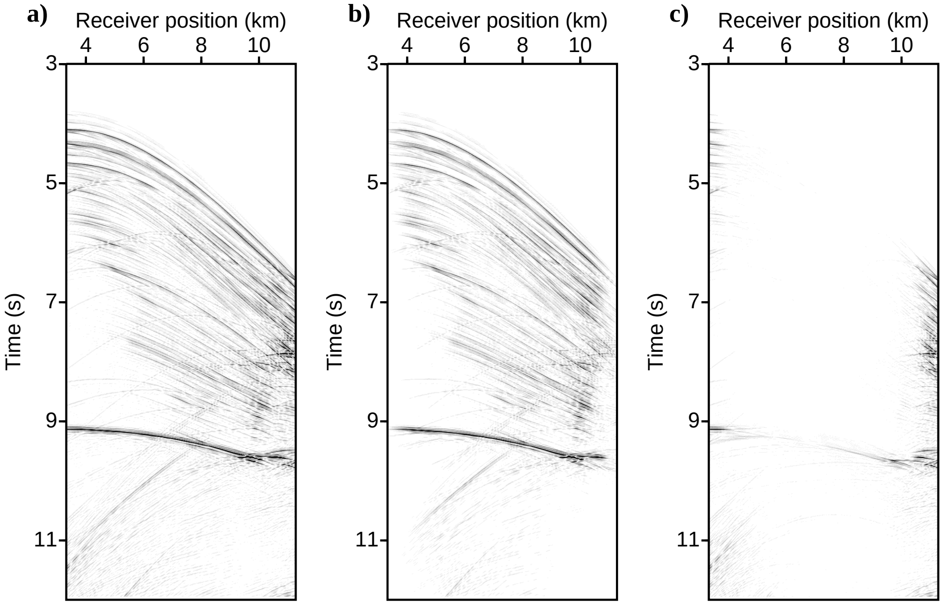

Figure 5.8: Sigsbee 2a example, single shot: a) the shot gather, b) the reconstructed shot gather, c) the difference between a) and b).

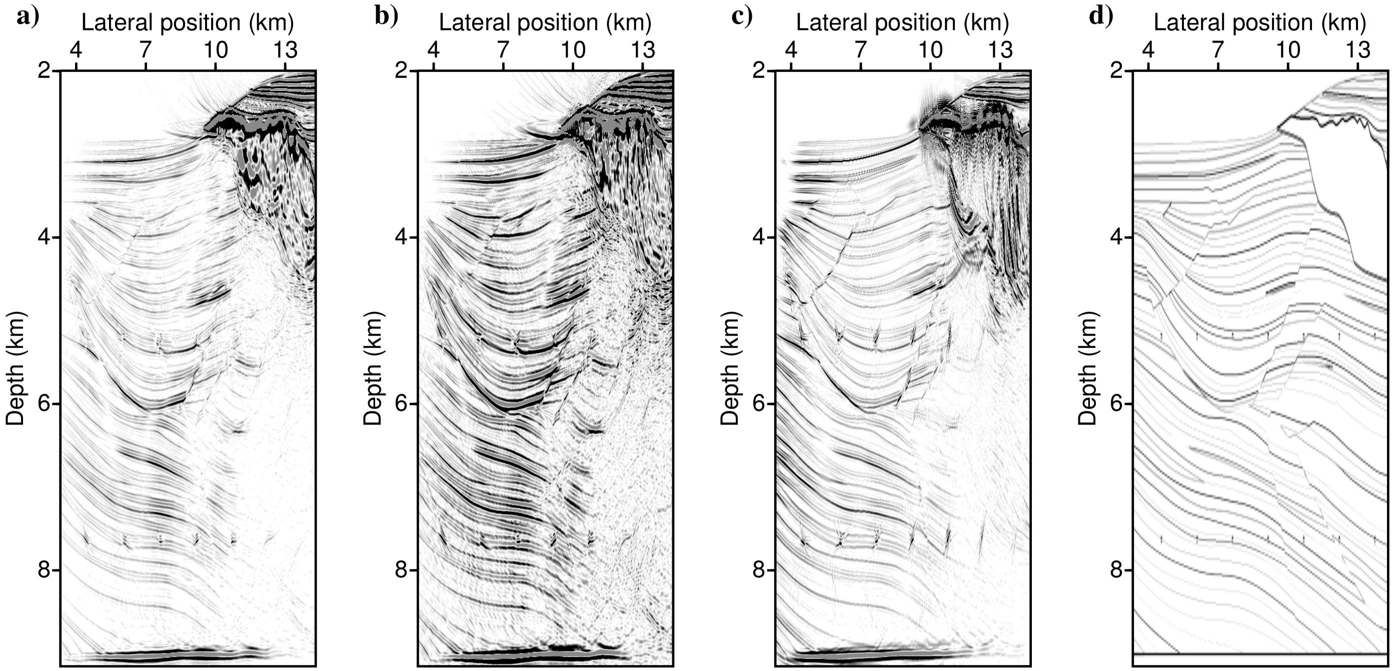

Figure 5.9: Sigsbee 2a example, 241 shots: a) the migration (adjoint), the least-squares migration (inverse) using b) the pre-stack parameterization of the scattering potential and c) the post-stack parameterization of the scattering potential. d) the true reflectivity.

We applied migration and least-squares migration to data generated from the Sigsbee 2a model, showing differences between the two results. For example, point scatterers in the Sigsbee 2a reflectivity were better resolved, especially when the input data consists of a single shot gather. Further, we note that shot-profile least-squares migration is capable of working under noisy input data by making use of prior constraints (model norm and data weights) in the least-squares normal equations. Moreover, we note that results depended on the parameterization of the scattering potential. Namely that the pre-stack parameterization allowed for better fitting of the data due to its greater number of degrees of freedom in the model as compared to the post-stack case. The post-stack parameterization of the scattering potential provided an improved migration image as compared to the post-stack case.

■ CHAPTER 6

Data reconstruction with shot-profile least-squares migration

6.1 Introduction

We introduce a data reconstruction algorithm, *shot-profile migration data reconstruction* (SPDR). SPDR uses the least-squares shot-profile migration algorithm introduced in Chapter 5, but for a constant reference wave-speed c_0. This is done both for efficiency and so that minimal assumptions are made about earth structure. We remind the reader that at the core of least-squares migration are wave-field modelling (de-migration) and migration operators, the former mapping from model space to data space, and the latter mapping from data space to model space. SPDR uses least-squares migration to find an optimal instance of model space which, in turn, is mapped to data space using de-migration, providing the reconstructed shot gather. SPDR is applied to a single shot gather, and in this thesis we assume that the shot gather has a single offset coordinate. This means that SPDR is a one dimensional data reconstruction algorithm. We present an extension of SPDR to multiple shot locations via modifications to the forward and adjoint operators. We call the extension, two dimensional shot profile migration data reconstruction (SPDR2), and use it to interpolate in both one shot and one geophone dimension. We can think of SPDR2 as a multi-dimensional generalization of SPDR.

In seismic data reconstruction, algorithms fall into one of two categories, being rooted in either signal processing or the wave-equation. Examples of the former include $f-x$ trace interpolation (Spitz, 1991), $f-k$ trace interpolation (Gülünay, 2003), minimum weighted norm interpolation (Liu and Sacchi, 2004), reconstruction using the curvelet transform (Herrmann and Hennenfent, 2008), and multistep autoreggressive reconstruction (Naghizadeh and Sacchi, 2007), while examples of the later include the use of the Born approximation and asymptotic methods (Stolt, 2002; Chiu and Stolt, 2002; Ramírez et al., 2006), Green's thoerem (Ramírez and Weglein, 2009), post-stack Stolt migration and data fitting methods (Trad, 2003), and dip move-out operators (e.g. Baumstein and Hadidi, 2006). SPDR belongs to the family of wave-equation based methods for data reconstruction. It differs from previous efforts in its parameterization of model space, being based on shot-profile migration and de-migration operators. Additionally, it relies on data fitting methods such as those used in Trad (2003), rather than direct inversion and asymptotic approximation which are used in, for example, Stolt (2002).

A challenge for data reconstruction algorithms is the aliasing of signal. In particular, when aliased energy is present and interferes with signal, their separation becomes challenging (but, not impossible). A recent example of data reconstruction is Naghizadeh and Sacchi (2007), and uses the non-aliased part of data to aid in the reconstruction of the aliased part of data. An alternative approach is to transform data via some operator that maps from data space to some model space, and such that in that model space, the corresponding representation of signal and alias are separable. This is a common idea used in various signal processing methods (e.g. Trad et al., 2003), and is also the approach that we take in SPDR. In particular, the SPDR model space is the constant velocity shot-profile migration of a shot gather (i.e. a common shot image gather). This means that the model space is a representation of the earth's reflectors parameterized by pseudo-depth (i.e. depth under the assumption of a constant velocity migration velocity model) and lateral position. We will show that under the assumption of limited dips in the earth's reflectors, the SPDR model space allows for the suppression of alias while preserving signal.

We begin with a description of shot-profile migration and de-migration built from the Born approximation to the acoustic wave-field and constant velocity Green's functions. This is similar to the derivation presented in Chapter 5, except that the constant reference wave-speed (i.e. constant velocity Green's functions) leads to more efficient algorithms for both wave-field modelling and migration. We apply shot-profile migration to an analytic example in order to illustrate its mapping of signal and alias from data space (shot gather) to model space (image gather). The mapping will infer that with constrained dip in the earth's reflectors, the signal and alias map to disjoint regions of model space. We note that if the dips in the earth's reflectors are large, then the model space representation of signal and alias are not necessarily disjoint, and the SPDR algorithm will fail unless survey parameters are adjusted to increase Nyquist wave-numbers. We reproduce the analytic result with a similar numerical result. Given the de-migration (forward) and migration (adjoint) operators, we construct a set of weighted least-squares normal equations. The normal equations are built such that 1) to some prescribed noise tolerance, the reconstructed shot gather fits the observed shot gather, and 2) the aliased portion of model space is suppressed. Solving the normal equations gives an optimal common shot image gather, and, in turn, the de-migration of the optimal common shot image gather is the reconstructed shot gather. In total, this is the procedure followed in the SPDR method. We apply SPDR to two data examples. The first is synthetic, and uses data generated by a finite difference implementation of the acoustic wave-equation. The second is a real data example from the Gulf of Mexico.

We follow our description of SPDR with a description of its extension to two dimensions (shot and geophone), SPDR2. In particular, this requires the extension of the forward and adjoint operators to include a variable shot position, further analysis of the signal and alias model space representations, and a slight re-definition of the SPDR least-squares data fitting algorithm. The forward and adjoint operators for SPDR2 are similar to those shown in Section 5.5 in that the model space is post-stack (using a sum over common shot image gathers). We apply SPDR2 to two classes of data reconstruction. The first being data interpolation in both the shot and geophone dimensions, and the second being near-offset data extrapolation. In the first case we show synthetic and real data examples, and in the latter case we restrict ourselves to a real data example.

6.2 Shot-profile migration and de-migration operators

SPDR relies on least-squares shot-profile migration (Chapter 5). We remind the reader that least-squares migration depends on migration and de-migration operators, the former providing a mapping from model space to data space (the forward operator), and the later being its adjoint, mapping from data space to model space. Within the context of the Born approximation, the model space is called the scattering potential (e.g. Weglein et al., 2003), and under the shot-profile parameterization of the operators, the scattering potential is a common shot image gather. In Chapter 5, we described shot-profile wave-equation migration and de-migration operators using split-step Green's functions and a Gazdag like depth marching algorithm. Here, we describe shot-profile wave-equation migration and de-migration operators built with a constant velocity Green's functions, meaning that the reference wave-speed is constant. The use of shot-profile migration rather than source-receiver migration (Biondi, 2003) leads to a shot-profile parameterization of model space which is integral to the SPDR algorithm. Namely, that the model is parameterized by pseudo-depth and lateral position. The term pseudo-depth describes a depth axis that honours the constant reference wave-speed. That is, the pseudo-depth of a reflector does not necessarily reflect its true depth due to the migration velocity model.

The forward operator (de-migration) for SPDR models the scattered seismic wave-field using the Born approximation (equation 2.4) under the assumption of an acoustic and constant velocity Green's function G_0 geared toward the shot-profile parameterization (equation 5.1). We remind ourselves that In equation 2.4, α is the scattering potential. Within the context of least-squares migration it is the model, and for SPDR it is a common shot image gather.

The forward operator in equation 2.4 describes the mapping from the scattering potential α (model space) to the scattered wave-field ψ_s (data space) recorded at geophone positions $(\mathbf{x}_g, z_g)$ where $\mathbf{x}_g = (x_g, y_g)$, and due to the seismic source located at $(\mathbf{x}_s, z_s)$ where $\mathbf{x}_s = (x_s, y_s)$, and with frequency distribution $f(\omega)$. We recall that the vertical wave-number k_{gz} in equation 5.1 is given by the dispersion relation in equation 5.3. The dispersion relation will play a role in understanding the null-space of the de-migration operator which, in turn, will affect our understanding of where alias in the data maps to in the model.

Substitution of equation 5.1 into equation 2.4 with $z_g = z_s = z_0$, and taking the support of α to be below z_0 (see Appendix E.1) gives,

$$\begin{aligned}&\psi_s(\mathbf{x}_g, \omega; \mathbf{x}_s)\\ &\quad= \left(\frac{\omega}{c_0}\right)^2 \left(\frac{1}{2\pi}\right)^4 \mathcal{F}_g^* \int_{z_0}^{\infty} u_p^1(\mathbf{k}_{gx}, z', \omega) \mathcal{F}_g \left[\mathcal{F}_g^* u_p^1(\mathbf{k}_{gx}, z', \omega) g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega)\right] \alpha(\mathbf{x}_g, z') dz',\end{aligned} \tag{6.1}$$

where,

$$u_p^1(\mathbf{k}_{gx}, z', \omega) = -\frac{e^{ik_{gz}(z'-z_0)}}{i4k_{gz}}, \tag{6.2}$$

and $g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega)$ was given in equation 5.7, and is interpreted as being the seismic point source located laterally at $\mathbf{x} = \mathbf{x}_s$. Equations 6.1 and 6.2 describe shot-profile de-migration (i.e. wave-field modelling, or the forward operator). They constitute the constant reference wave-speed de-migration analogue to equations 5.10 and 2.10. In equation 6.1, $\mathcal{F}_g$ is the two-dimensional and unnormalized Fourier transform, mapping from lateral space $\mathbf{x}_g$ to lateral wave-numbers $\mathbf{k}_{gx}$, and $\mathcal{F}_g^*$ is its adjoint so that, as in Chapter 5, the corresponding inverse Fourier transform is $(2\pi)^{-2}\mathcal{F}_g^*$. The function u_p^1 in equation 6.2 is equation 5.5 in Chapter 5 for $l = 0$. It is a phase-shift operator, propagating the wave-field toward, or away from potential scattering points within a constant reference wave-speed. In equation 6.2, the vertical wave-number k_{gz} is given by the dispersion relation defined in equation 2.8. For the sake of notational convenience, we will drop the superscript 1 from u_p^1 for the remainder of this chapter.

The use of the shot-profile parameterization in the construction of wave-field modelling operator in equation 6.1 is integral to SPDR. We recall that it means that in $\alpha(\mathbf{x}_g, z')$, $\mathbf{x}_g$ are the lateral coordinates of the earth, and z' is pseudo-depth (i.e. depth under the assumption of the constant migration velocity model that we use in the Born approximation). This means that the model space is characterized by the sub-surface reflectors that fall within the aperture of the shot. The shot-profile parameterization of the scattering potential has a second, less pleasing, consequence. Namely that, despite the constant reference wave-speed, we cannot use Stolt migration (Stolt, 1978), thus producing a more expensive data reconstruction algorithm than the one derived in Trad (2003).

Not surprisingly, the adjoint of the de-migration operator (equation 6.1) is migration. We find it by, once again, recognizing a Fredholm integral equation of the first kind in the de-migration operator, so that,

$$\psi_s(\mathbf{x}_g,\omega;\mathbf{x}_s) = \int_{z_0}^{\infty} u(\omega,z';\mathbf{x}_g,\mathbf{x}_s)\alpha(\mathbf{x}_g,z')dz',$$

where,

$$u(\omega,z';\mathbf{x}_g,\mathbf{x}_s) = \left(\frac{\omega}{c_0}\right)^2\left(\frac{1}{2\pi}\right)^4 \mathcal{F}_g^* u_p(\mathbf{k}_{gx},z',\omega)\mathcal{F}_g\left[\mathcal{F}_g^* u_p(\mathbf{k}_{gx},z',\omega)g(\mathbf{k}_{gx},\mathbf{x}_s,\omega)\right].$$

Then,

$$\alpha^\dagger(\mathbf{x}_g,z') = \int_{-\infty}^{\infty} u^*(\omega,z';\mathbf{x}_g,\mathbf{x}_s)\psi_s(\mathbf{x}_g,\omega;\mathbf{x}_s)d\omega, \tag{6.3}$$

where,

$$u^*(\omega,z';\mathbf{x}_g,\mathbf{x}_s) = \left(\frac{\omega}{c_0}\right)^2\left(\frac{1}{2\pi}\right)^4 \left[\mathcal{F}_g^* u_p^*(\mathbf{k}_{gx},z',\omega)g^*(\mathbf{k}_{gx},\mathbf{x}_s,\omega)\right]\mathcal{F}_g^* u_p^*(\mathbf{k}_{gx},z',\omega)\mathcal{F}_g. \tag{6.4}$$

In equation 6.4 we have used the relation,

$$\left[\mathcal{F}_g^* u_p(\mathbf{k}_{gx},z',\omega)\mathcal{F}_g g(\mathbf{x}_g,\mathbf{x}_s,\omega)\right]^* = \mathcal{F}_g^* u_p^*(\mathbf{k}_{gx},z',\omega)\mathcal{F}_g g^*(\mathbf{x}_g,\mathbf{x}_s,\omega).$$

We substitute equation 6.4 into equation 6.3, finding,

$$\begin{aligned}&\alpha^\dagger(\mathbf{x}_g,z';\mathbf{x}_s)\\ &\quad= \left(\frac{1}{2\pi}\right)^4 \int \left(\frac{\omega}{c_0}\right)^2 \left[\mathcal{F}_g^* u_p^*(\mathbf{k}_{gx},z',\omega)g^*(\mathbf{k}_{gx},\mathbf{x}_s,\omega)\right]\mathcal{F}_g^* u_p^*(\mathbf{k}_{gx},z',\omega)\mathcal{F}_g\psi_s(\mathbf{x}_g,\omega;\mathbf{x}_s)d\omega,\end{aligned} \tag{6.5}$$

where $(\cdot)^*$ denotes the adjoint when applied to the Fourier operator, and the complex conjugate otherwise. Equation 6.5 is shot-profile migration for an acoustic and constant velocity migration velocity model. With equations 6.1 and 6.5, we have the necessary tools to construct a constant velocity shot-profile least-squares migration algorithm for a constant reference wave-speed.

To produce efficient numerical implementations of the forward and adjoint operators in equations 6.1 and 6.5, we find iterative algorithms that implement the operators. The derivation of the iterations are straight-forward, and the resulting algorithms should be familiar to those found in Chapter 5. However, they differ enough to be interesting. In particular, we will show that the iteration uses an operator that is invariant to depth. This results in more efficient algorithms. In addition, we find that only one iteration over depth is required for the forward operator. This is in contrast to the variable reference wave-speed case in Chapter 5 where an additional and separate recursion over depth was required. We begin with the forward operator (equation 6.1), re-arranging terms and approximating the integral with a Riemann sum so that,

$$\begin{aligned}&\psi_s(\mathbf{x}_g,\omega;\mathbf{x}_s)\\ &\quad= \left(\frac{\omega}{c_0}\right)^2\left(\frac{1}{2\pi}\right)^4\mathcal{F}_g^*\Delta z\sum_{l=1}^{n_z}u_{p(l)}(\mathbf{k}_{gx},\omega)\mathcal{F}_g\left[\mathcal{F}_g^*u_{p(l)}(\mathbf{k}_{gx},\omega)g(\mathbf{k}_{gx},\mathbf{x}_s,\omega)\right]\alpha(\mathbf{x}_g,z_l),\end{aligned} \tag{6.6}$$

where,

$$u_{p(l)}(\mathbf{k}_{gx},\omega) = -\frac{e^{ik_{gz}(z_l-z_0)}}{i4k_{gz}}, \tag{6.7}$$

and the earth model is partitioned into n_z layers of constant thickness Δz. We implement equations 6.6 and 6.7 using two iterations. First, we define $v_{s(l)}$ for $l = 1\dots n_z$ such that,

$$\begin{aligned}v_{s(1)}(\mathbf{k}_{gx},\omega;\mathbf{x}_s) &= u_{p(1)}(\mathbf{k}_{gx},\omega)g(\mathbf{k}_{gx},\mathbf{x}_s,\omega)\\ v_{s(l)}(\mathbf{k}_{gx},\omega;\mathbf{x}_s) &= \Delta u_p(\mathbf{k}_{gx},\omega)v_{s(l-1)}(\mathbf{k}_{gx},\omega;\mathbf{x}_s),\end{aligned} \tag{6.8}$$

where $u_{p(1)} = \exp(ik_{gz}(z_1-z_0))/(i4k_{gz})$ and $\Delta u_p = \exp(ik_{gz}\Delta z)$. Second, we define $v_{r(l)}$ so that,

$$\begin{aligned}v_{r(1)}(\mathbf{k}_{gx},\omega) &= u_{p(1)}(\mathbf{k}_{gx},\omega)\\ v_{r(l)}(\mathbf{k}_{gx},\omega) &= \Delta u_p(\mathbf{k}_{gx},\omega)v_{r(l-1)}(\mathbf{k}_{gx},\omega).\end{aligned} \tag{6.9}$$

Then the forward operator in equation 6.6 becomes,

$$\psi_s(\mathbf{x}_g,\omega;\mathbf{x}_s) = \left(\frac{\omega}{c_0}\right)^2\left(\frac{1}{2\pi}\right)^4\mathcal{F}_g^*\Delta z\sum_{l=1}^{n_z}v_{r(l)}(\mathbf{k}_{gx},\omega)\mathcal{F}_g\left[\mathcal{F}_g^*v_{s(l)}(\mathbf{k}_{gx},\omega;\mathbf{x}_s)\right]\alpha(x_g,z_l;\mathbf{x}_s). \tag{6.10}$$

Equations 6.8-6.10 constitute an algorithm that implements the de-migration operator in equation 6.1, and the bulk of computation is in the two two-dimensional Fourier transforms required per depth and frequency.

Next, we consider the implementation of the adjoint operator (equation 6.5), approximating the integral over angular frequency with a Riemann sum (and using the Fourier representation of the scattered wave-field),

$$\begin{aligned}&\alpha^\dagger(\mathbf{x}_g,z_l;\mathbf{x}_s)\\ &\quad= \left(\frac{1}{2\pi}\right)^4\Delta\omega\sum_{j=1}^{n_\omega}\left(\frac{\omega_j}{c_0}\right)^2\left[\mathcal{F}_g^*u_{p(l)}^*(\mathbf{k}_{gx},\omega)g^*(\mathbf{k}_{gx},\mathbf{x}_s,\omega)\right]\mathcal{F}_g^*u_{p(l)}^*(\mathbf{k}_{gx},\omega)\psi_s(\mathbf{k}_{gx},\omega_j;\mathbf{x}_s).\end{aligned} \tag{6.11}$$

Analogous to the forward operator, we implement equation 6.11 using two iterations. First, we define $v^*_{s(l)}$ for $l = 1 \dots n_z$ so that,

$$\begin{aligned} v^*_{s(1)}(\mathbf{k}_{gx}, \omega; \mathbf{x}_s) &= u^*_{p(1)}(\mathbf{k}_{gx}, \omega) g(\mathbf{k}_{gx}, \mathbf{x}_s, \omega) \\ v^*_{s(l)}(\mathbf{k}_{gx}, \omega; \mathbf{x}_s) &= \Delta u^*_p(\mathbf{k}_{gx}, \omega) v_{s(l-1)}(\mathbf{k}_{gx}, \omega; \mathbf{x}_s). \end{aligned}$$

Second, we define $v^*_{r(l)}$ for $l = 1 \dots n_z$ so that,

$$\begin{aligned} v^*_{r(1)}(\mathbf{k}_{gx}, \omega; \mathbf{x}_s) &= u^*_{p(1)}(\mathbf{k}_{gx}, \omega) \psi_s(\mathbf{k}_{gx}, \omega; \mathbf{x}_s) \\ v^*_{r(l)}(\mathbf{k}_{gx}, \omega; \mathbf{x}_s) &= \Delta u^*_p(\mathbf{k}_{gx}, \omega) v_{r(l-1)}(\mathbf{k}_{gx}, \omega; \mathbf{x}_s). \end{aligned}$$

Then the adjoint operator in equation 6.11 becomes,

$$\alpha^\dagger(\mathbf{x}_g, z_l; \mathbf{x}_s) = \left(\frac{1}{2\pi}\right)^4 \Delta\omega \sum_j \left(\frac{\omega_j}{c_0}\right)^2 \left[\mathcal{F}^*_g v^*_{s(l)}(\mathbf{k}_{gx}, \omega_j; \mathbf{x}_s)\right] \mathcal{F}^*_g v^*_{r(l)}(\mathbf{k}_{gx}, \omega_j; \mathbf{x}_s), \quad (6.12)$$

again requiring two two dimensional Fourier transforms per depth and frequency. Equation 6.12 is shot-profile migration for a constant velocity reference wave-speed. It is important to note that in the iterations for the forward and adjoint operators, Δu_p is constant with respect to the depth index l. Taking advantage of this fact in their software implementation, allows for a constant reference wave-speed (de-)migration that is significantly faster than the equivalent variable reference wave-speed (de-)migration.

For the purposes of SPDR, $\alpha^\dagger$ is called the adjoint, and ψ_s is called data. The adjoint indicates how energy distributes in model space given its computation from some instance of data. Thus, before using the forward and adjoint operators in least-squares migration, we will analyze the adjoint to illustrate how signal and alias are mapped from data space to model space. This analysis will, in turn, motivate the construction of the weights in the weighted least-squares migration algorithm.

6.3 Observed data on a nominal grid

SPDR uses least-squares migration for data reconstruction. The input to the least-squares migration algorithm is observed seismic data (shot gather), and is sampled from the scattered wave-field $\psi_s(\mathbf{x}_g, \omega; \mathbf{x}_s)$. In practise, we have some finite set of geophones to represent ψ_s in its lateral dimensions $\mathbf{x}_g$. In SPDR, we assume $2D$ earth models and a single shot gather so that both data and model spaces are sufficiently described by a single lateral dimension x_g. Then, we define a regular nominal grid x^n_g for that same lateral dimension so that,

$$x^n_g \in \{x_g | x_g = k\Delta x^n_g, k = 1 \dots n^n_g\}.$$

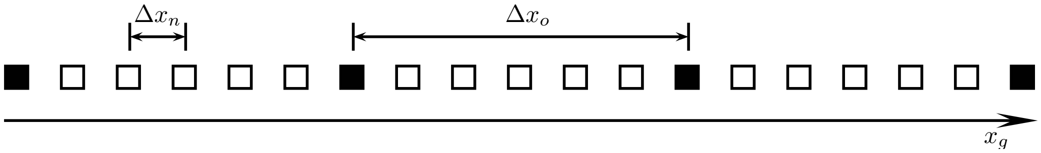

Figure 6.1: Schematic example of nominal and observed grids. The boxes (open and filled) denote the nominal grid, and the observed grid is denoted by the filled boxes. Δx_g^n denotes the grid spacing for the nominal grid, and Δx_g^o denotes the grid spacing for a regular observed grid.

Further, we define an observed grid x_g^o so that $x_g^o \in x_g^n$. This means that each sample on the observed grid is also a sample on the nominal grid. Further, in data space, each sample of the observed grid corresponds to an observation (i.e. a seismic trace). This is a common idea used in data reconstruction algorithms (e.g. Liu and Sacchi, 2004), and is illustrated in Figure 6.1. In this chapter we assume that the observed grid is regular with sample spacing Δx_g^o, although the case of an irregular x_g^o grid is also interesting (but perhaps less challenging since it tends to increase the Nyquist wave-number (e.g. Babu and Stoica, 2010)). SPDR is equally applicable to regular and irregular observed grids.

6.4 Adjoint mapping of signal and alias due to the observed grid

In this section, we analyze the adjoint (shot-profile migration with a constant reference wave-speed) operator with an analytic example. We let data ψ_s be the sampling (Dirac-comb) function corresponding to a regular observed grid x_g^o. Our goal is to provide some understanding of where signal and alias in data space map to in model space via the adjoint operator. We begin with a data space description of the signal and alias using the sampling theorem. Then, we find the adjoint (model space representation) of the sampling function by applying the shot-profile migration operator described in the previous section. This provides a basis for categorizing the energy in model space as corresponding to either data space signal or data space alias; that is, as either model space signal or model space alias.

In the sampling theorem, we represent $\psi_s(x_g, \omega; x_s)$ with $\bar{\psi}_s = III_o(x_g)\psi_s$ where $III_o(x_g)$ is a sampling function such that,

$$III_o(x_g) = \sum_{k=0}^{n_g^o} \delta(x_g + k\Delta x_g^o). \tag{6.13}$$

Then, a well know result of the sampling theorem states that (e.g. Lathi, 1998, p. 319),

$$\bar{\psi}_s(k_{gx},\omega;x_s) = \sum_{p=-\infty}^{\infty} \psi_s(k_{gx} + p2\pi/\Delta x_g^o, \omega; x_s). \tag{6.14}$$

The series in equation 6.14 contains both signal (for $p = 0$), and alias for $p \neq 0$.

We notice that equation 6.14 is signal and alias in data space. We are interested in their manifestation in model space. For this purpose, we consider the migration operator (equation 6.5) applied to $\bar{\psi}_s$ (equation 6.14), so that (using the Fourier representation of the scattered wave-field),

$$\begin{aligned}&\bar{\alpha}^\dagger(x_g,z;x_s)\\ &\quad= \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} \left(\frac{\omega}{c_0}\right)^2 \left[\mathcal{F}_g^* u_p^*(k_{gx},z,\omega) g^*(k_{gx},x_s,\omega)\right] \mathcal{F}_g^* u_p^*(k_{gx},z,\omega) \bar{\psi}_s(k_{gx},\omega;x_s) d\omega.\end{aligned} \tag{6.15}$$

Then, substitution of equation 6.14 into equation 6.15 gives,

$$\begin{aligned}&\bar{\alpha}^\dagger(x_g,z;x_s)\\ &= \left(\frac{1}{2\pi}\right)^2 \Bigg[\int_{-\infty}^{\infty} \left(\frac{\omega}{c_0}\right)^2 \left[\mathcal{F}_g^* u_p^*(k_{gx},z,\omega) g^*(k_{gx},x_s,\omega)\right] \mathcal{F}_g^* u_p^*(k_{gx},z,\omega) \psi_s(k_{gx},\omega;x_s) d\omega\\ &\quad + \int_{-\infty}^{\infty} \left(\frac{\omega}{c_0}\right)^2 \left[\mathcal{F}_g^* u_p^*(k_{gx},z,\omega) g^*(k_{gx},x_s,\omega)\right] \mathcal{F}_g^* u_p^*(k_{gx},z,\omega) \psi_s(k_{gx} + 2\pi/\Delta x_g^o,\omega;x_s) d\omega\\ &\quad + \int_{-\infty}^{\infty} \left(\frac{\omega}{c_0}\right)^2 \left[\mathcal{F}_g^* u_p^*(k_{gx},z,\omega) g^*(k_{gx},x_s,\omega)\right] \mathcal{F}_g^* u_p^*(k_{gx},z,\omega) \psi_s(k_{gx} - 2\pi/\Delta x_g^o,\omega;x_s) d\omega\\ &\quad + \int_{-\infty}^{\infty} \left(\frac{\omega}{c_0}\right)^2 \left[\mathcal{F}_g^* u_p^*(k_{gx},z,\omega) g^*(k_{gx},x_s,\omega)\right] \mathcal{F}_g^* u_p^*(k_{gx},z,\omega) \psi_s(k_{gx} + 4\pi/\Delta x_g^o,\omega;x_s) d\omega\\ &\quad + \cdots \Bigg]\\ &\quad = \int_{-\infty}^{\infty} \alpha_0^\dagger(x_g,z,\omega;x_s) d\omega + \int_{-\infty}^{\infty} \alpha_1^\dagger(x_g,z,\omega;x_s) d\omega + \int_{-\infty}^{\infty} \alpha_{-1}^\dagger(x_g,z,\omega;x_s) d\omega\\ &\qquad + \int_{-\infty}^{\infty} \alpha_2^\dagger(x_g,z,\omega;x_s) d\omega + \cdots,\end{aligned} \tag{6.16}$$

where,

$$\begin{aligned}\alpha_p^\dagger(x_g,z,\omega;x_s) = \left(\frac{1}{2\pi}\right)^2 \left(\frac{\omega}{c_0}\right)^2 &\left[\mathcal{F}_g^* u_p^*(k_{gx},z,\omega) g^*(k_{gx},x_s,\omega)\right]\\ &\times \mathcal{F}_g^* u_p^*(k_{gx},z,\omega) \psi_s(k_{gx} + p2\pi/\Delta x_g^o,\omega;x_s).\end{aligned} \tag{6.17}$$

The term $\int \alpha_0^\dagger(x_g,z,\omega;x_s) d\omega$ from equation 6.16 is the hypothetical image that would be obtained from a continuously sampled spatial dimension x_g in the data (i.e. $\alpha_0^\dagger = \alpha^\dagger$), and $\alpha_p^\dagger$, $p \neq 0$ is the expression of the alias in model space (i.e. the model space alias).

We illustrate the model space alias for a single frequency component of $\bar{\alpha}^\dagger$. In particular, we let $\bar{\psi}_s(x_g, \omega; x_s) = III_o(x_g)\psi_0(\omega)$ which is shown in Figure 6.2a for $k = 0 \ldots 10$. The discrete Fourier transform of $III_o(x_g)$ is (e.g. Lathi, 1998, p. 320),

$$III_o(k_{gx}) = \frac{1}{\Delta x_g^o} \sum_{p=-\infty}^{\infty} \delta\left(k_{gx} + p\frac{2\pi}{\Delta x_g^o}\right),$$

and is shown in Figure 6.2b. Next, we let $\bar{\psi}_s(x_g, \omega; x_s) = III_o(x_g)\psi_0(\omega)$, so that in equation 6.14,

$$\psi_s(k_{gx} + p2\pi/\Delta x_g^o, \omega; x_s) = \frac{1}{\Delta x_g^o}\delta\left(k_{gx} + p\frac{2\pi}{\Delta x_g^o}\right)\psi_0(\omega). \tag{6.18}$$

Upon substitution of equation 6.18 into equation 6.17, we find for any given frequency ω (see Appendix E.2),

$$\begin{aligned}\alpha_p^\dagger(k_{gx}, \kappa_z, &\omega; x_s = 0) \\ &= \frac{f^*(\omega)\psi_0(\omega)}{\Delta x_g^o}\left(\frac{\omega}{c_0}\right)^2 \delta\left(\kappa_z + (k_{gz(p)} + sgn(\omega)\sqrt{\omega^2/c_0^2 - (k_{gx} - k_p)^2})\right),\end{aligned} \tag{6.19}$$

where $k_p = p2\pi/\Delta x_g^o$,

$$k_{gz(p)} = sgn(\omega)\sqrt{\frac{\omega^2}{c_0^2} - k_p^2},$$

κ_z is the Fourier conjugate variable of z, and k_{gz} was given by the dispersion relation in equation 2.8. Equation 6.19 means that $\alpha_p^\dagger$ is non-zero only when,

$$\kappa_z = -k_{gz(p)} - sgn(\omega)\sqrt{\frac{\omega^2}{c_0^2} - (k_{gx} - k_p)^2}. \tag{6.20}$$

Equation 6.20 is shown in Figure 6.2c for $p = 0, \pm 1$.

In SPDR, we force the null space of the migration operator to include the evanescent portion of the wave-field which, in turn, is defined for when k_{gz} in the dispersion relation (equation 2.8) is imaginary. This is common practise in wave-equation migration algorithms, although there are exceptions (Huang et al., 1999). The non-evanescent portion of the wave-field is defined for when,

$$-\frac{\omega}{c_0} \leq k_{gx} \leq \frac{\omega}{c_0}.$$

In Figure 6.2b, we plot the bounds of the non-evanescent region under some assumed frequency ω and migration wave-speed c_0. Notice that energy for $p = 0, \pm 1$ falls within these bounds, and all other energy falls within the evanescent portion of the wave-field. Hence, for this example, the adjoint operator will annihilate energy mapped from data space for where $|p| \geq 2$. This is exactly why in Figure 6.2c, we show $\alpha_p^\dagger$ for only $p = 0, \pm 1$. In model space, these bounds are translated by k_p in equation 6.19 so that,

$$-\frac{\omega}{c_0} + k_p \leq k_{gx} \leq \frac{\omega}{c_0} + k_p.$$

In particular, we show, in Figure 6.2c, the lower and upper bounds for $\alpha_p^\dagger(k_{gx}, \kappa_z, \omega; x_s = 0)$ for $p = 0$, the lower bound for $p = -1$, and the upper bound for $p = 1$.

The synthetic experiment shown in Figure 6.3 confirms the analytic example. In particular, we define a nominal grid x_g^n with $\Delta x_g^n = 5m$. In Figure 6.3a, we plot $\psi_s(x_g, \omega; x_s) = I\hat{I}I_o(x_g)$ for $x_g \in x_g^n$ where,

$$I\hat{I}I_o(x_g) = \begin{cases} 1 & , \quad x_g = k\Delta x_g^o, \quad k = 0 \ldots 6 \\ 0 & , \quad \text{otherwise,} \end{cases}$$

and where $\Delta x_g^o = 300m$. In other words, the observed grid is regular with samples spaced every $300m$. We show the adjoint of $I\hat{I}I_o(x_g)$ evaluated for a temporal frequency of $6Hz$, in Figure 6.3b. The resulting pattern mimics what is predicted by the analytic example.

The examples presented in Figure 6.2 and 6.3 were for a single frequency. In Figure 6.4, we show results for when five frequencies, equally spaced between 6 and $10Hz$, contribute to the integral in the adjoint operator (equation 6.5). In particular, Figure 6.4b shows the model space analytic result, while Figure 6.4a shows a comparable synthetic example.

The analytic example in this section gives us an understanding of where signal and alias in data space map to in model space. This understanding will be useful in the development of SPDR. However, we point out that the application of the adjoint operator to the sampling function is equivalent to propagating a receiver side horizontal plane wave into the earth, and then applying the downward continued source, and the shot-profile migration imaging condition. In practise, the data will contain some continuum of plane-wave components, and each will contribute to the distribution of energy in model space.

6.5 Reflector dip constraints on model space

In Section 6.4, we explored the adjoint mapping of the sampling function. The example showed how the sampling function in data space maps to model space. Moreover, the resulting pictures show obvious overlap between the model space representation of signal and alias (e.g. Figure 6.4). This overlap would make it difficult to construct an algorithm

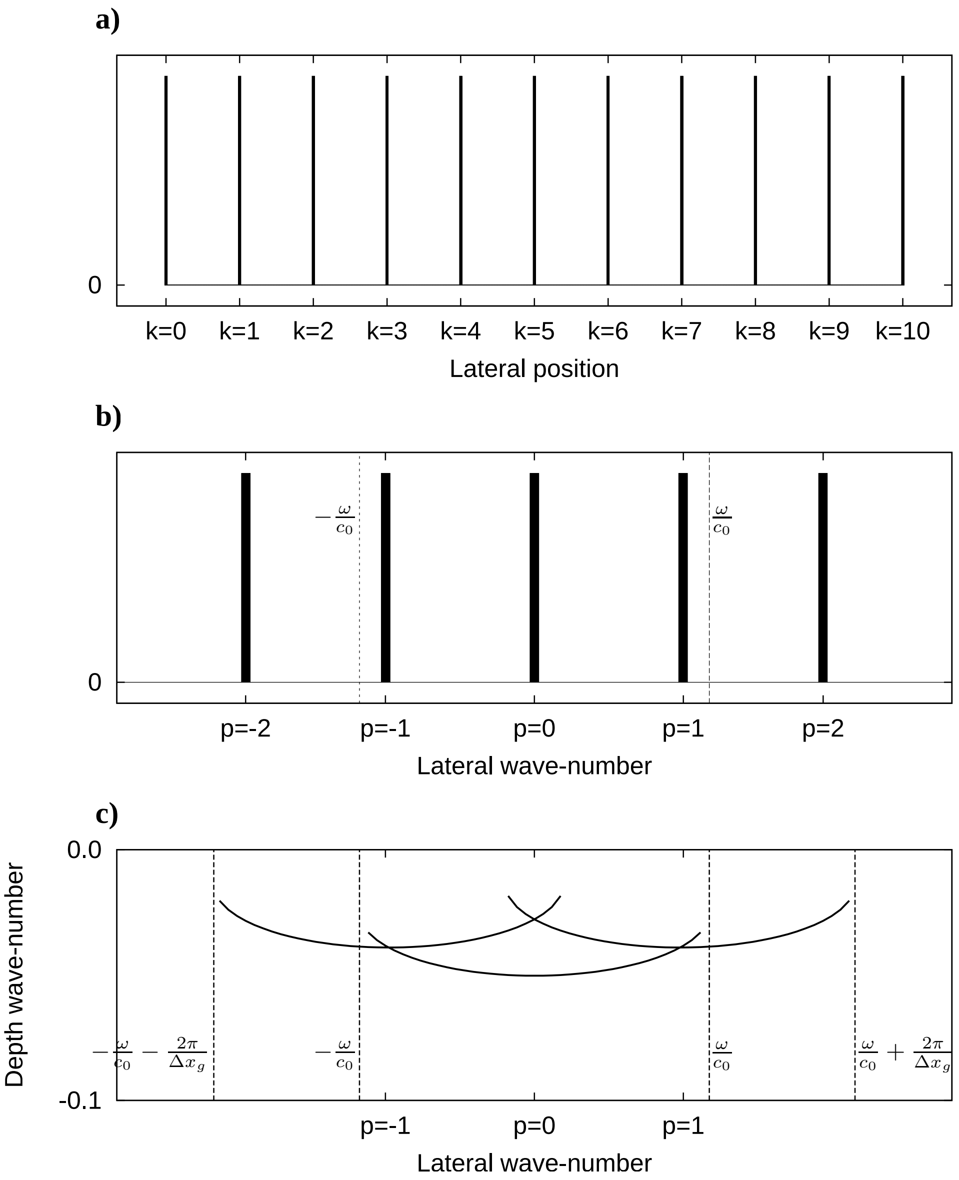

Figure 6.2: Analytic example for a single frequency: a) the sampling function for $k = 0 \ldots 10$, b) the Fourier representation of the sampling function, and c) the adjoint of the sampling function in lateral k_{gx} and depth κ_z wave-number coordinates.

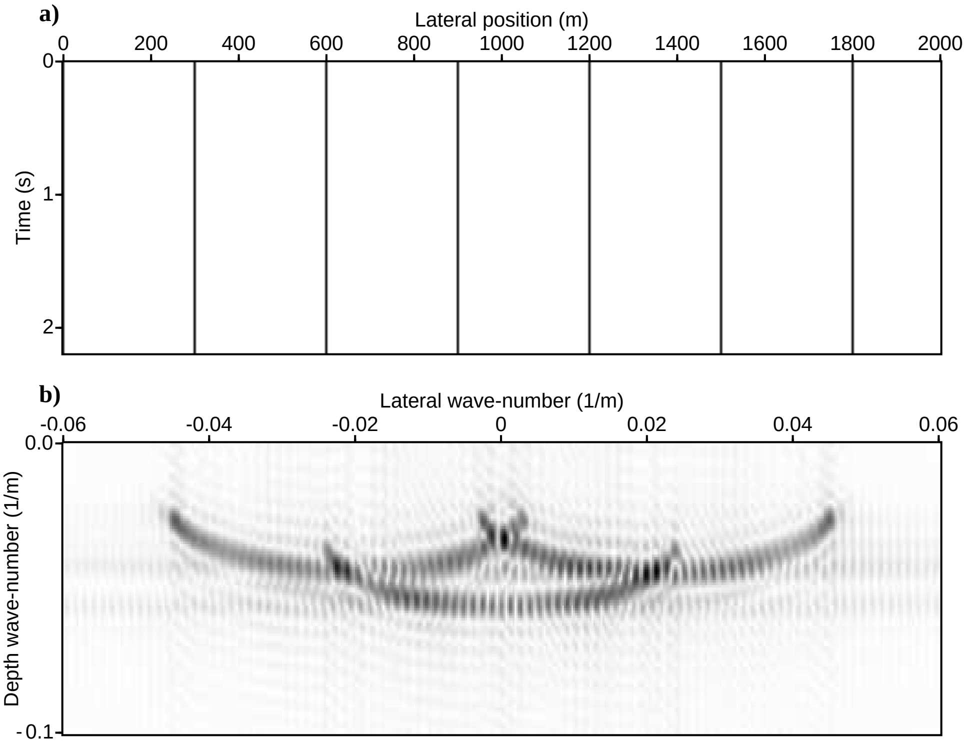

Figure 6.3: Synthetic example for a single frequency: a) the sampling function, b) the adjoint of the sampling function in lateral k_{gx} and depth κ_z wave-number coordinates.

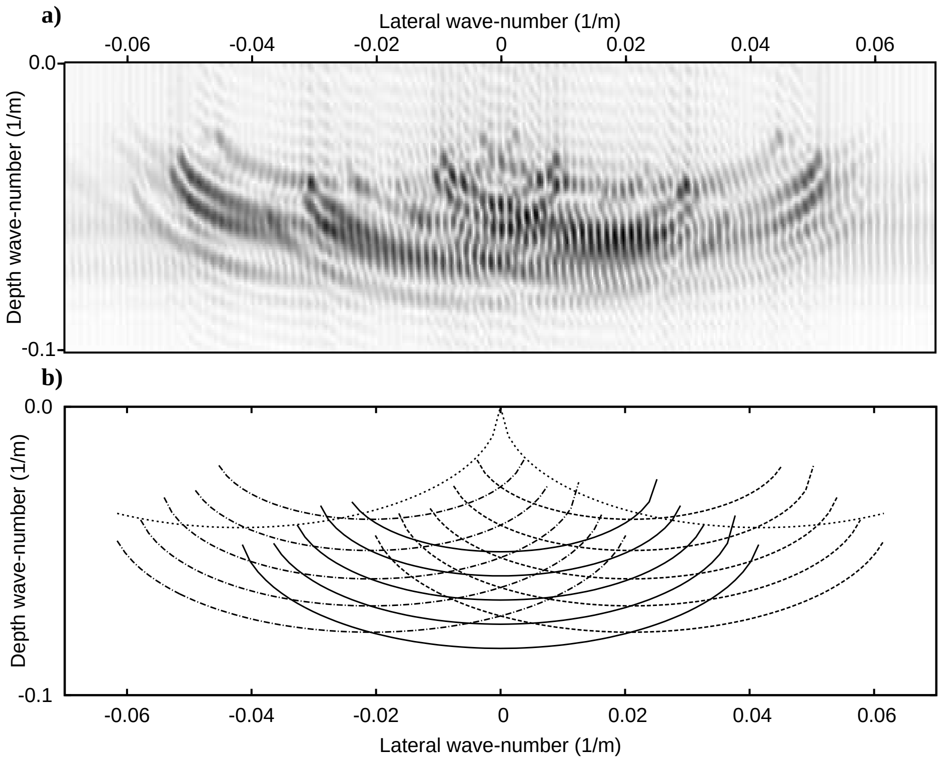

Figure 6.4: Multi-frequency example: the adjoint of the sampling function for a) the synthetic example, and b) the analytic example. Five frequencies spaced evenly between 6 and $10Hz$ are used in the integral over frequency (equation 6.5). In b) we show energy for $p = 0$ (solid line), $p = 1$ (long dashed line), $p = -1$ (dash-dot line), $p = \pm 2$ (short dashed line).

that suppresses the model space alias while preserving the model space signal, and this is exactly how we intend SPDR to function. However, we remind ourselves that the model space provides a representation of the earth's reflectors. This is a direct consequence of using shot-profile migration for the adjoint operator. Thus, it follows that for any given frequency the adjoint will be band-limited in k_{gx} according to the maximum dip of the reflectors (parameterized by lateral position and pseudo-depth). We postulate that this band-limitation allows for a disjoint representation of model space signal and alias. In this section we illustrate this idea with two examples. First, we apply a band-limitation operator to the analytic example studied in the previous section. Second, we use a synthetic example in which the earth model consists of a single reflector.

In the adjoint of the sampling function, we assume that $\alpha_p^\dagger$ is band-limited so that equation 6.19 becomes,

$$\begin{aligned}\alpha_p^\dagger(k_{gx},\kappa_z,&\omega;x_s=0)\\ &= \frac{f^*(\omega)\psi_0(\omega)}{\Delta x_g^o}\left(\frac{\omega}{c_0}\right)^2 h(k_{gx},\kappa_z)\delta(\kappa_z+(k_{gz(p)}+sgn(\omega)\sqrt{\omega^2/c_0^2-(k_{gx}-k_p)^2})),\end{aligned} \tag{6.21}$$

where,

$$h(k_{gx},\kappa_z)=\left\{\begin{array}{ccc}1 & , & |k_{gx}|\le k_b(\kappa_z)\\ 0 & , & |k_{gx}|>k_b(\kappa_z),\end{array}\right.$$

and where $k_b(\kappa_z)$ is built from the expected maximum dip of the earth's reflectors, as represented in pseudo-depth. In particular, if a reflector has dip ξ such that,

$$\alpha(x_g,z)=\delta(z+\xi x_g),$$

then it can be shown that,

$$\alpha(k_{gx},\kappa_z)=2\pi\delta(k_{gx}-\xi\kappa_z). \tag{6.22}$$

Hence, it must be that $k_b(\kappa_z)=\xi\kappa_z$. For example, when the dip of the reflector is null, so is k_b, and, likewise, k_b increases with increasing ξ. Figure 6.5 illustrates equation 6.21 for small dip ($\xi=0.1$). It is a band-limited version of Figure 6.4b. Notice that with the band-limiting function $h(k_{gx},\kappa_z)$ applied, the model space signal and alias of the sampling function are separable. This will make it possible to build a data reconstruction algorithm that filters out the alias, but preserves the signal.

To further illustrate the band-limitation, we construct a synthetic example using the model in Figure 6.6a, and the finite difference approximation to the acoustic wave-equation, constructing data ψ_s in Figure 6.6b. The source ($f(\omega)$ in equation 5.7) is a minimum-phase

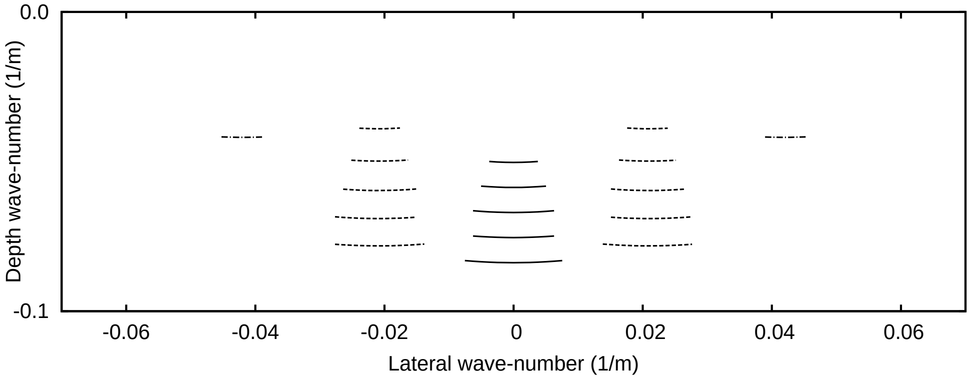

Figure 6.5: Analytic and multi-frequency example: we represent the adjoint of the sampling function under the band-limitation constraint introduced in equation 6.21.

Ricker wavelet with a peak frequency of 15Hz. For the data, we use a nominal grid spacing of $5m$, and a regular observation grid, with observations (geophones) spaced every $300m$. The data are severely aliased as illustrated by its $f-k$ spectrum in Figure 6.6c. Since the model consists of a single reflector with null dip, we expect the adjoint operator to separate signal and alias. In Figure 6.6d, we plot the adjoint of the data using the shot-profile migration operator. A quick comparison to the band-limited analytic example in Figure 6.5 allows us to label the clusters of energy corresponding to model space signal and alias. In particular, we identify energy corresponding to $p = 0, \pm 1, \pm 2$.

We can construct a lower bound for the maximum reflector dip ξ resulting in separable model space signal and artifacts, and allowing for the successful application of SPDR. This dip will be a function of the observed grid spacing (i.e. the lateral Nyquist wave-number). In particular, we note that equation 6.22 gives the relation $k_{gx} = -\xi\kappa_z$ so that $\max|k_{gx}| = \xi \max|\kappa_z|$. At the very least, it must be that $\max|k_{gx}| < k_1$ where $k_1 = 2\pi/\Delta x_g^o$ was defined in the previous section. Hence, it follows that $\xi \max|\kappa_z| < 2\pi/\Delta x_g^o$, or, in other words, $\xi < 2\pi/(\Delta x_g^o \max|\kappa_z|)$. This shows, for example, an inverse relation between the observed grid sampling interval and the maximum allowed reflector dip in model space. In other words it shows a direct relation between the lateral Nyquist wave-number and the maximum allowed dip.

6.6 Least-squares data reconstruction

Thus far, we have identified the adjoint operator (shot-profile migration) as a useful tool for mapping data (common shot data gather) with interfering data space signal and alias to a model space (common shot image gather) where signal and alias are separated. To move

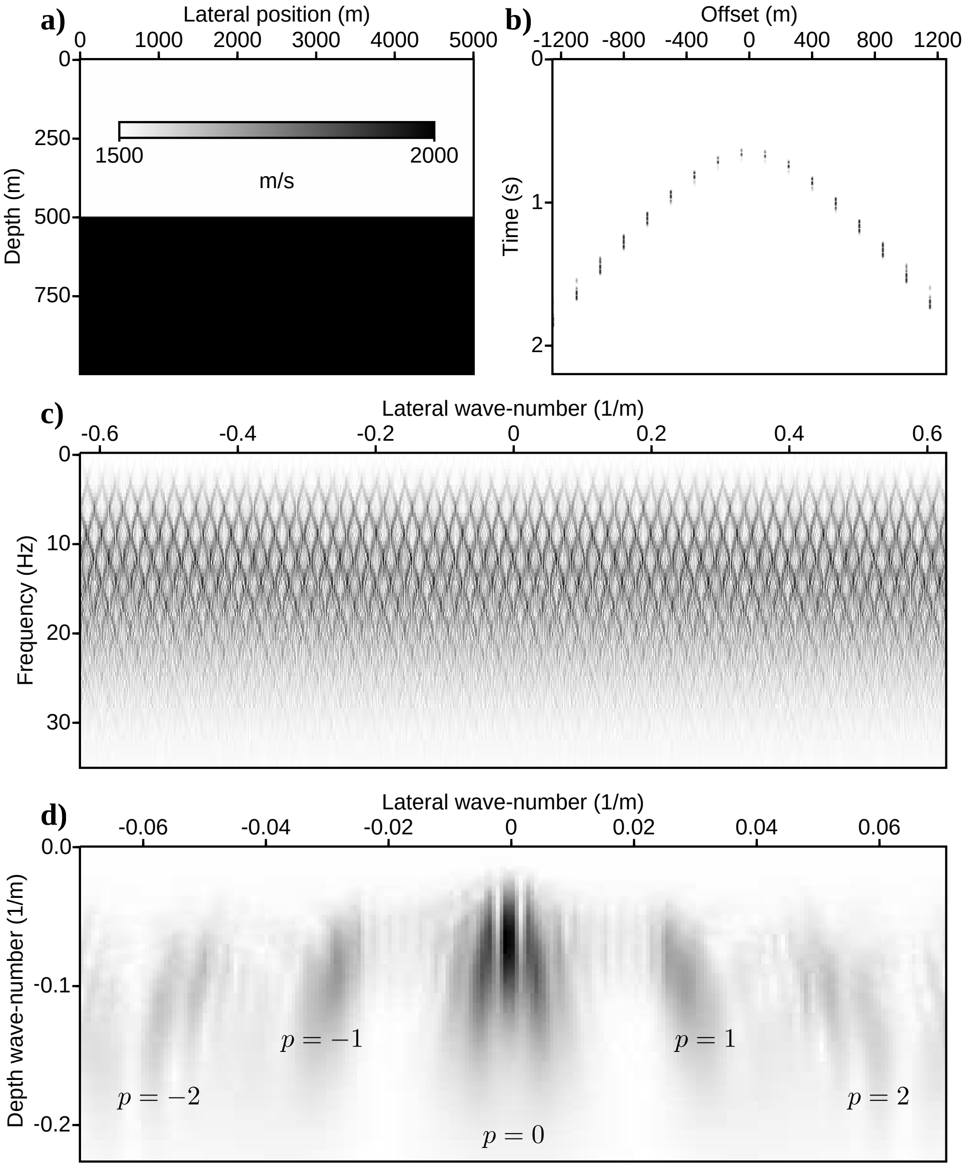

Figure 6.6: Synthetic example for a zero dip reflector: a) the earth model is a half-space over a half-space, b) synthetic data generated by the finite difference method, c) $f - k$ spectrum of the data, and d) $k_{gx} - \kappa_z$ spectrum of the adjoint. We identify energy corresponding to $p = 0, \pm 1, \pm 2$ in equation 6.19.

toward a complete description of SPDR, we are, next, tasked with finding a method that suppresses the alias from model space, while preserving the signal. For this purpose, we use a weighted least-squares inversion, rather than the adjoint, to construct an instance of model space. The consequences are two-fold. First, it will ensure that the forward operator (de-migration) applied to the model produces modelled data that fits the observed data. Second, it allows us to penalize those regions of model space that contain an expression of alias rather than signal (in sections 6.4 and 6.5, we identified disjoint regions of model space that correspond to either model space signal or alias). We begin with a description of the weighted least-squares normal equations. Then, we construct data weights that honour the observed data, and model weights that penalize the aliased energy.

To construct the weighted least-squares inversion, we begin by with a reminder of some previously used definitions. We let $\mathbf{d}$ be a data vector of length M realized from a single shot gather $\psi_s(x_g^n, \omega; x_s)$, and where we recall that x_g^n describes the nominal grid for the lateral dimension x_g. We let $\mathbf{d}$ be the observed data, and a vector of length M. It is sampled by the nominal grid x_g^n, and will be non-zero only for the intersection of the nominal x_g^n and observed x_g^o grids. For this single shot gather, $M = n_\omega n_g^n$ where n_ω is the number of non-negative frequency samples, and, as before, n_g^n is the number of grid points in the nominal grid x_g^n. Likewise, we let $\mathbf{m}$ be a model vector of length N realized from $\alpha(x_g^n, z; x_s)$, where $N = n_z n_g^n$, and n_z are the number of samples in depth for the shot-profile migration image gather. Last, we let $\mathbf{A}$ be the $M \times N$ matrix representation of the forward operator (shot-profile de-migration in equation 6.1), and $\mathbf{A}^H$ its corresponding adjoint operator (shot-profile migration in equation 6.5).

To find an optimal $\mathbf{m}$ that honours the observed data $\mathbf{d}$, we find the minimum of the cost function,

$$\begin{aligned} \phi(\mathbf{m}) &= ||\mathbf{W}_d(\mathbf{d} - \mathbf{A}\mathbf{m})||_2^2 + \mu||\mathbf{W}_m\mathbf{m}||_2^2 \\ &= \phi_d(\mathbf{m}) + \mu\phi_m(\mathbf{m}), \end{aligned} \tag{6.23}$$

where $\mathbf{W}_d$ are data weights, and $\mathbf{W}_m$ are model weights. This is the same as the cost function used in Chapter 5, except for the introduction of $\mathbf{W}_m$ and the change of notation for the data weights ($\mathbf{W}_d$ rather than $\mathbf{W}$, to distinguish it from the model weights). As in previous chapters, we partition the cost function into two components ϕ_d and ϕ_m, with ϕ_d being the data misfit function, and ϕ_m being the model norm function. Finding the minimum of equation 6.23 results in the normal equations,

$$(\mathbf{A}^H\mathbf{W}_d^H\mathbf{W}_d\mathbf{A} + \mu\mathbf{W}_m^H\mathbf{W}_m)\mathbf{m} = \mathbf{A}^H\mathbf{W}_d^H\mathbf{W}_d\mathbf{d}, \tag{6.24}$$

which we solve to find an optimal scattering potential $\mathbf{m} = \mathbf{m}_*$. Then, the data can be reconstructed using the wave-field modelling (forward) operator such that $\mathbf{d}_* = \mathbf{A}\mathbf{m}_*$. That is, $\mathbf{d}_*$ is the SPDR reconstructed data. In practise, equation 6.24 is solved using the least-squares conjugate gradient method, and implicit construction of the matrices.

Key to the construction of $\mathbf{m}_*$, and thus also to the reconstructed data $\mathbf{d}_*$ is the choice of the model and data weight matrices, $\mathbf{W}_m$ and $\mathbf{W}_d$. The data weight matrix $\mathbf{W}_d$ penalizes missing data, and was shown in equation 5.22. In chapter 5 it penalized dead traces. Here, it, more specifically, penalizes any trace that does not fall on the observed grid. That is, it is a diagonal matrix such that its i^{th} diagonal element is,

$$[\mathbf{W}_d]_{ii} = \begin{cases} 1 & , \quad i \in \mathcal{I}_d \\ 0 & , \quad i \notin \mathcal{I}_d, \end{cases} \tag{6.25}$$

where $\mathcal{I}_d$ is the set of indices corresponding to the observed grid x_g^o. This ensures that the cost function in equation 6.23 works to fit only the observed data, and not the missing data. The construction of model weights is slightly more involved, and draws on the discussion in sections 6.4 and 6.5. In particular, we define,

$$\mathbf{W}_m = \mathbf{F}^{-1}\mathbf{W}\mathbf{F}, \tag{6.26}$$

where $\mathbf{F}$ is the two-dimensional Fourier transform over lateral position x_g and depth z. Then, $\mathbf{W}$ penalizes model space aliased energy according to the windowing function,

$$w(k_{gx},\kappa_z) = \begin{cases} 1 + 1/\epsilon & , \quad k_{gx} < -k_2(\kappa_z) \\ 1 + \frac{1}{2\epsilon}\left(1 + \cos(\pi\frac{k_{gx}+k_2(\kappa_z)}{\tau})\right) & , \quad -k_2(\kappa_z) < k_{gx} < -k_1(\kappa_z) \\ 1 & , \quad -k_1(\kappa_z) < k_{gx} < k_1(\kappa_z) \\ 1 + \frac{1}{2\epsilon}\left(1 + \cos(\pi(1 + \frac{k_{gx}-k_1(\kappa_z)}{\tau}))\right) & , \quad k_1(\kappa_z) < k_{gx} < k_2(\kappa_z) \\ 1 + 1/\epsilon & , \quad k_2(\kappa_z) < k_{gx}, \end{cases} \tag{6.27}$$

where $\epsilon \ll 1$ is some small constant. The functions $k_1(\kappa_z)$ and $k_2(\kappa_z)$ are defined according to the model dip constraints in equation 6.22. Namely,

$$k_1(\kappa_z) = \hat{\xi}|\kappa_z| + \eta \tag{6.28}$$

$$k_2(\kappa_z) = \hat{\xi}|\kappa_z| + \eta + \tau, \tag{6.29}$$

so that τ defines the length of the cosine taper, and η is some estimate of the bandwidth of the model space signal. The parameter $\hat{\xi}$ is the estimated maximum dip of the earth's reflectors as parameterized by pseudo-depth and lateral position in the migrated image gather (see equation 6.22). For example, we could set $\hat{\xi} = 0$, and $\eta = 2\pi/\Delta x_g^o$, the former for simplicity and the latter motivated by the sampling theorem (see equation 6.14). The parameter τ

defines the length of the cosine taper. The function $k_b(\kappa_z)$ was defined in equation 6.22, and depends on some user selected maximum dip of the earth's reflectors. We note that equation 6.27 can be generalized to allow for specification of separate maximum positive and negative dips.

To illustrate, we continue with the example shown in Figure 6.6. In particular, we use the decimated data from Figure 6.6b for $\mathbf{d}$ in equation 6.24. Then, we solve equation 6.24 for $\mathbf{m}_*$. We recall that $\mathbf{m}_*$ is a discrete representation of the scattering potential $\alpha = \alpha_*(x_g, z; x_s)$ (common-shot image gather), and in Figure 6.7c we plot its amplitude in the wave-number domain $|\alpha_*(k_{gx}, \kappa_z; x_s)|$. This should be compared to Figure 6.6d, where we plot the adjoint of the decimated data $|\bar{\alpha}^\dagger(k_{gx}, \kappa_z; x_s)|$. Finally, we use $\mathbf{m}_*$ to compute the reconstructed data $\mathbf{d}_* = \mathbf{A}\mathbf{m}_*$. We show the reconstructed data in Figure 6.7b. For comparison, we show the same data computed using the finite difference approximation to the acoustic wave-equation in Figure 6.7a. In Figure 6.7d, we plot the $f - k$ spectrum of the reconstructed data which should be compared to the $f - k$ spectrum of the decimated data which was shown in Figure 6.6c.

We make some comments about the results in Figure 6.7. First, in the recovered data, we notice some missing energy at the far-offsets (indicated by the arrows in Figure 6.7a). This energy corresponds to the seismic head-wave. Second, in Figure 6.7c, we plot the model weights. In this example, we set $\hat{\xi} = 0$ so that the model weights are invariant to the vertical wave-number κ_z. Finally, we plot the data misfit ϕ_d as a function of conjugate gradient iteration in Figure 6.8, showing convergence of the iterative least-squares conjugate gradient algorithm, that we use to solve equation 6.24, after 60 iterations. One can use, for example, a χ^2 statistic built from an estimate of the noise in the data to determine a stopping criterion for the conjugate gradient iterations (e.g. Hansen, 1998). Numerical tests show that the algorithm tends to converge sufficiently after tens of iterations (of course, this will depend on the size of the model space).

Our description of SPDR is now complete. The data weights in the least-squares inversion were built to honour the known data on the observed grid x_g^o, while giving no weight to the remaining samples on the nominal grid x_g^n (i.e. the missing traces in the shot gather). The model weights were built to penalize the portion of model space that corresponds to alias rather than signal. The synthetic example given in this section provided an illustration of SPDR. However, the example in this section is limited to a model with a single zero dip reflector which is, exactly, the least challenging scenario for SPDR. In the following two sections, we present more challenging examples for SPDR, where the reflectors have non-zero dip, and the earth's velocity is not necessarily well represented by the migration velocity model.

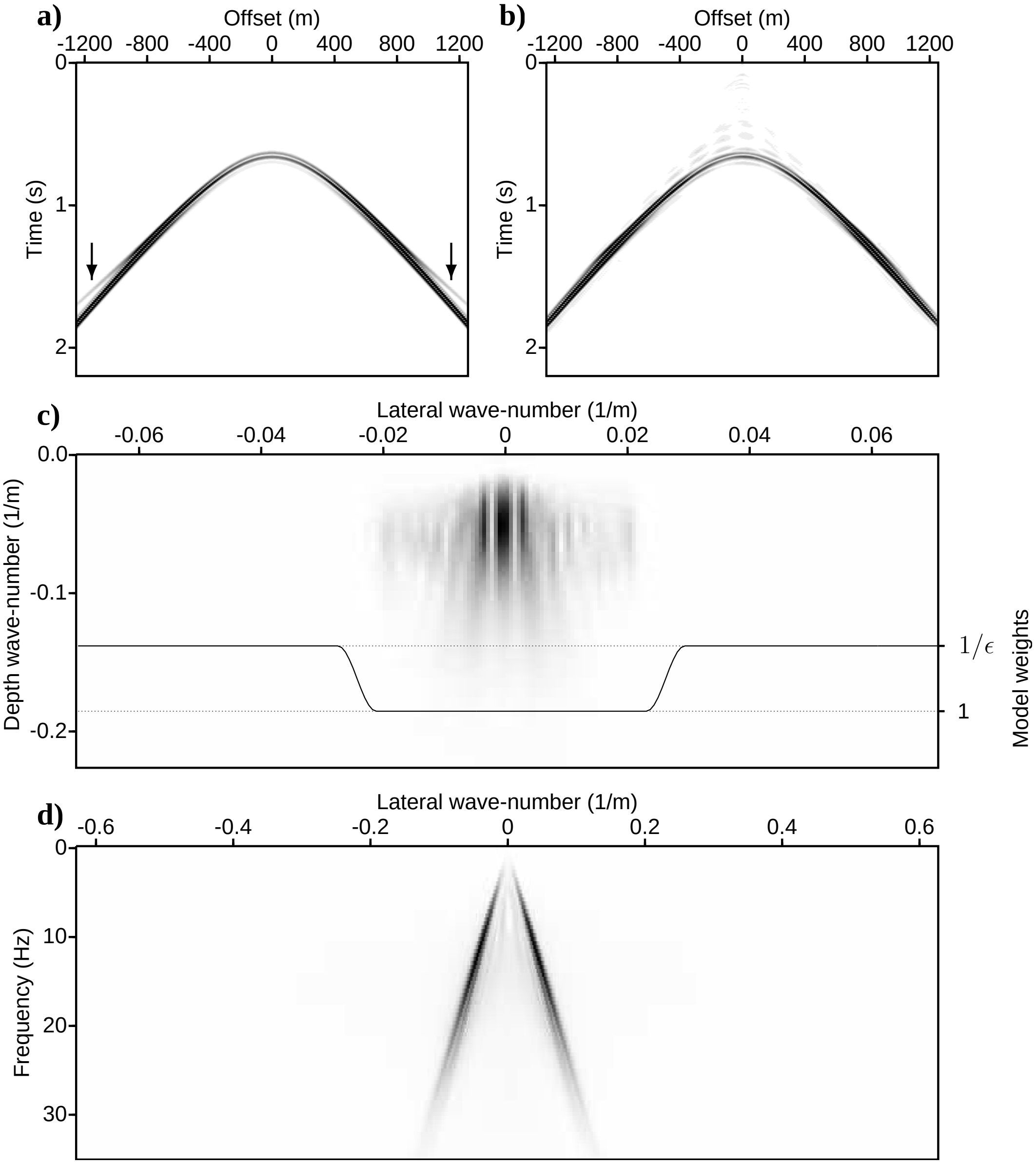

Figure 6.7: Synthetic example for a zero-dip reflector: a) data computed using the acoustic finite difference approximation to the acoustic wave-equation on the nominal grid. b) The reconstructed data computed from the observed data in Figure 6.6b, and using SPDR. c) The wave-number spectrum of the least-squares inverse $\mathbf{m}_*$. The model weights used in the least-squares migration normal equations are also shown in c). d) The $f-k$ spectrum of the SPDR reconstructed data $\mathbf{d}_*$. In a) the head-wave is indicated by two vertical arrows.

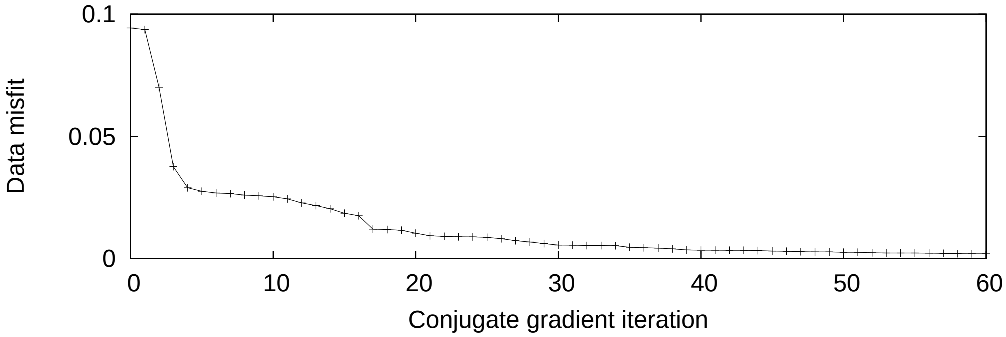

Figure 6.8: The least-squares normal equations are solved using least-squares conjugate gradients. We show the data misfit as a function of conjugate gradient iteration, showing the convergence of the algorithm.

6.7 Synthetic data examples for models with variable dip reflectors

Thus far, we have limited ourselves to a synthetic example where the earth model consists of a single zero-dip reflector. The earth model at and above the reflector consisted of a single velocity of $1500m/s$, precisely what we use for c_0 in the migration and de-migration operators. Moreover, the data set consisted of a single primary event, and neglected, for example, free-surface multiples. In this section, we apply SPDR to more involved synthetic examples, where the models contain a range of reflector dips. In a first example, we use a range of models, each containing a different dip for its interface. This illustrates the dip limitation of the algorithm, and its connection to the Nyquist wave-number. In a second example, we consider a model with multiple reflectors and a range of reflector dips (to maximum absolute dip of approximately $\xi = 0.7$, or a slope of approximately $45°$ as measured from the horizontal). Additionally, we allow the data to contain free-surface multiples, violating the single scattering approximation made in the migration/de-migration operators, and used in our data reconstruction algorithm[1].

For the first example, we aim to illustrate the dip limitation of SPDR, and its relation to the lateral Nyquist wave-number. Figure 6.9 shows the application of SPDR to three velocity models. From left to right, in the figure, we show models with increasing reflector dips. In particular, Figure 6.9a plots the velocity models for these three cases; Figure 6.9b plots the observed data built from the finite difference approximation to the acoustic wave-equation, and a shot located at $(x_s, z_s) = (2500m, 50m)$; Figure 6.9c plots the SPDR reconstructed data; and Figure 6.9d plots the wave-number spectra of the migrated gathers computed from

[1] The data also contains internal multiples, etc.

the data in 6.9b. We notice that in the shallowest dip case, shown in the first column of Figure 6.9, SPDR is successful. This success is due to the clear separation between the adjoint representation of data signal and alias shown in the wave-number spectra (Figure 6.9d, first column). On the other hand, in the second and third columns of Figure 6.9, the reflector dip is increased, and the separation between the signal and artifacts degrades, in turn, reducing the effectiveness of SPDR.

The location of the artifacts is, in part, dictated by the observed grid. In this example, the observed grid is regular with $\Delta x_g^o = 150m$. From the band-limited adjoint representation of data (equation 6.21), we predict that the model space artifacts are band-limited and centred at the lateral wave-numbers $k_{gx} = \pm 2\pi/150 = \pm 0.04.$, and this is observed in Figure 6.9d. Therefore, increasing the Nyquist wave-number by decreasing Δx_g^o will, in turn, improve the separation between the model space signal and artifacts (regardless of reflector dip), and subsequently improve the quality of the SPDR result. This is, of course, not a surprising conclusion, and simply states that data reconstruction becomes less challenging as Δx_g^o decreases.

For the second example, Figures 6.10 through 6.15 show the experiment and the SPDR results. In Figure 6.10a, we plot the acoustic velocity model for the experiment. We use the finite-difference approximation to the acoustic wave-equation to generate the synthetic shot gather in Figure 6.10b. The synthetic shot gather is built with a source located in the acoustic model at $(x_s, z_s) = (2500m, 50m)$, and with, again, a minimum phase Ricker wavelet with a peak frequency of 15Hz. The observations are on the regular grid x_g^o such that geophones are placed every $100m$, and run from $0m$ to $5000m$, and each at a depth of $50m$. The observations are placed on the nominal grid x_g^n with $\Delta x_g^n = 5m$ spacing. In Figure 6.10b, the data are plotted using a subset of the nominal grid. In particular, we plot the recorded data for every 4^{th} nominal grid sample and near offsets. In Figure 6.10c, we plot the $f-k$ spectrum of the recorded data.

We solve equation 6.24 for the optimal model (common shot image gather) $\mathbf{m}_*$, subsequently computing the SPDR reconstructed data $\mathbf{d}_* = \mathbf{A}\mathbf{m}_*$. In finding $\mathbf{m}_*$, we set the model weights such that $\hat{\xi} = 0$, $\eta = .09k_N$, $\tau = .01k_N$ and $\mu = 10^{-2}$, and where $k_N = 0.063$ is the Nyquist wave-number for the observed grid x_g^o. The result, again shown for near offsets and every 4^{th} trace on the nominal grid, is shown in Figure 6.11b. For comparison, we compute the data on the nominal grid using the finite difference approximation to the acoustic wave equation, plotting the resulting data in Figure 6.11a. Figure 6.11c plots the difference between the SPDR reconstructed data in Figure 6.11b, and the finite difference data in Figure 6.11a. Finally, in Figure 6.12, we plot the $f-k$ spectra of the finite-difference and SPDR reconstructed data. In particular, Figure 6.12a plots the $f-k$ spectrum of the finite difference data, and Figure 6.12b plots the $f-k$ spectrum of the SPDR reconstructed

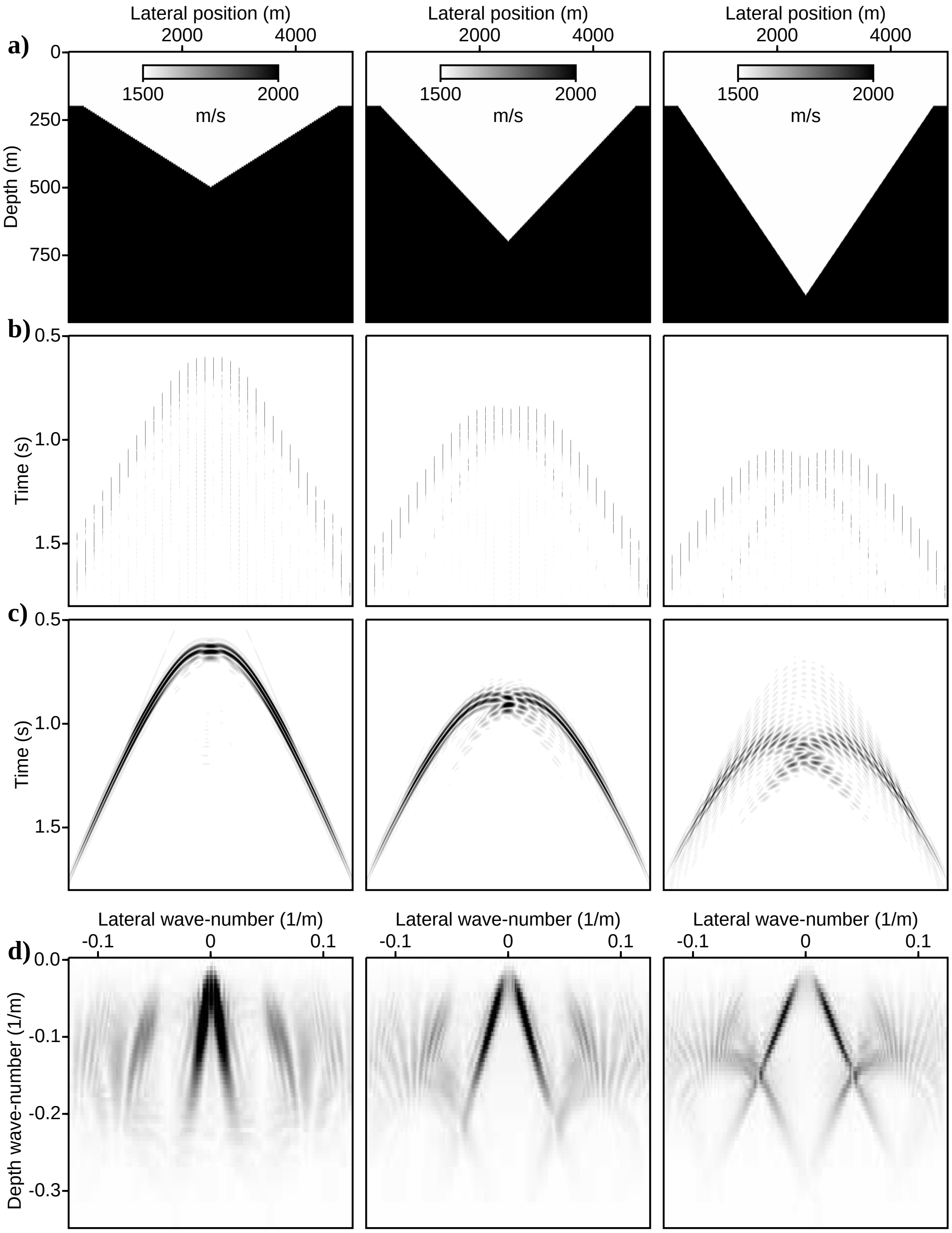

Figure 6.9: Synthetic data example showing the effect of dips: a) the velocity models, b) the observed data, c) the SPDR reconstructed data, and d) the wave-number spectra of the constant wave-speed migrated observed data. Each column plots a different experiment, with the dips of the reflectors increasing from left to right.

data.

We use the results to emphasize two characteristics of SPDR. First, the head-wave is not reconstructed. It falls within the null-space of the forward operator. In Figure 6.13a, we plot far-offset traces from $\mathbf{d}$, the observed data on the nominal grid (every 4^{th} trace in the nominal grid is shown). Figure 6.13b is the data computed for all offsets on the nominal grid using the finite difference approximation to the acoustic wave-equation. In this plot we see both the specular reflection, and the head-wave. The head-wave is indicated by the arrow. Finally, Figure 6.13c plots the data reconstruction $\mathbf{d}_*$ which shows the reconstruction of the specular reflections, and the omission of the head-wave. As we already mentioned, this is a limitation of the SPDR algorithm.

Second, we consider the free-surface multiple events. The single scattering approximation used in the de-migration operator does not, correctly, model free-surface multiples. Indeed, a first order approximation to a free-surface multiple requires a second order scattering interaction (e.g. Weglein et al., 2003). However, the single scattering approximation does see the multiple event as a false primary, and maps it to a copy of the true reflector at an erroneous depth, allowing for the extra travel-time of the wave-field through the water-column. This can be seen in Figure 6.14. In Figure 6.14a, we allow the model space to have a maximum depth of $1000m$. This allows for the model space to contain the pseudo-depth location of the reflectors, but does not allow for the first-order copy that is due to the free-surface multiples. Meanwhile, in Figure 6.14b, we allow the model space to extend to a depth of $2000m$. Then, the copy of the reflector that corresponds to the multiple energy falls within the bounds of the alloted model space. We note that the energy of the multiples mapped to model space is very small. Therefore, to see their energy, we had to apply a severe clip to the image. In turn, this shows various low amplitude artifacts. The effect in the reconstructed data of enlarging the model space to allow for the multiple of the reflector is shown in Figure 6.15. In Figure 6.15a, we plot finite difference data, and in Figures 6.15b and 6.15c, we plot the data reconstructions using, respectively, the larger and smaller model spaces. The corresponding data space effect of the larger model space is shown in Figure 6.15b, where the multiples are reconstructed. In contrast, the smaller model space result in Figure 6.15c excludes the multiples. Note that to show the small multiple energy, we have applied a significant clip to the plots.

6.8 Real data example

For a real data example, we use a shot gather from a marine data set from the gulf of Mexico. The shot gather has receivers spaced every $26.7m$, with the near offset receiver placed at a distance of $20.7m$ from the source, and the far offset receiver placed at a distance of

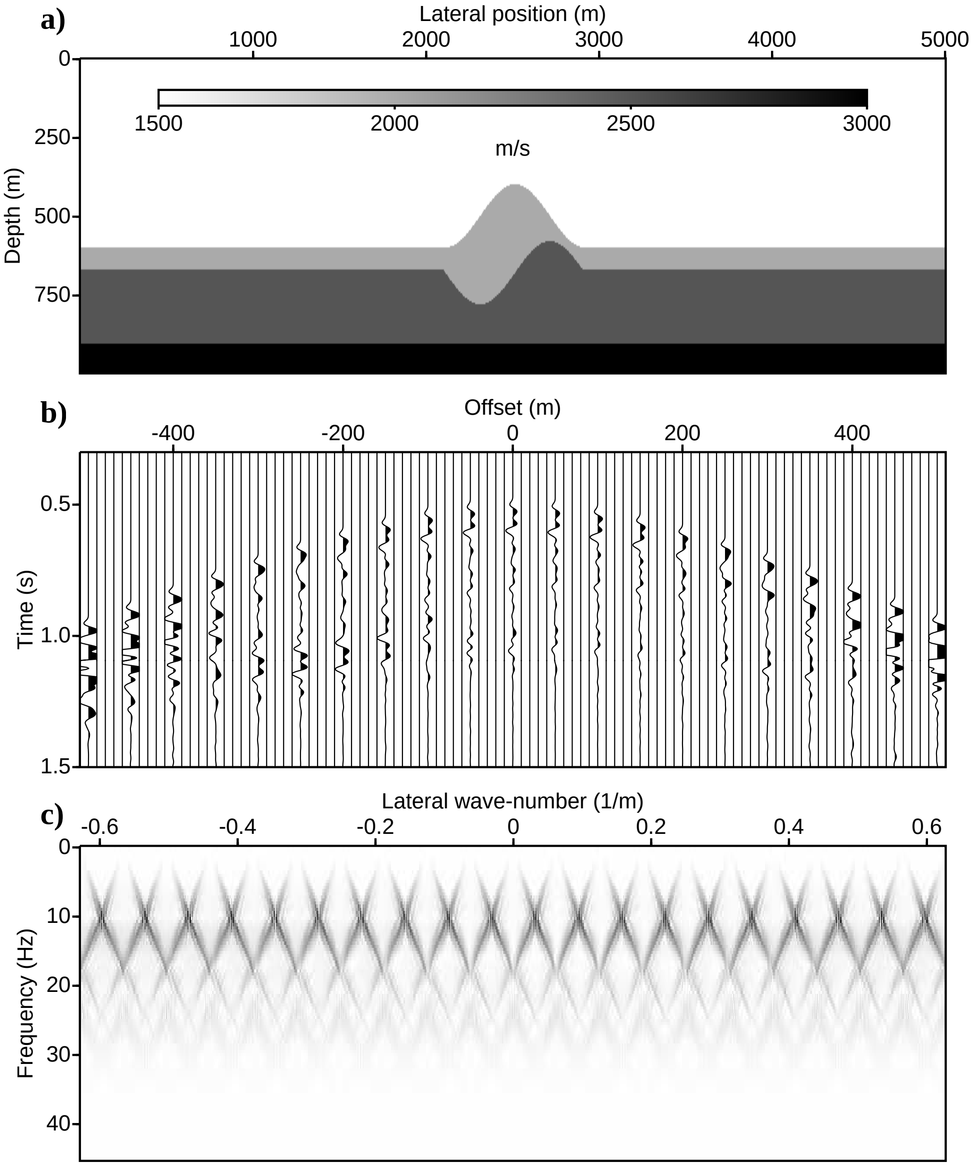

Figure 6.10: Synthetic data example: a) the acoustic velocity model, b) The observed data shown for its near-offset traces, and using every 4^{th} trace in the nominal grid, c) the $f-k$ spectrum of the observed data.

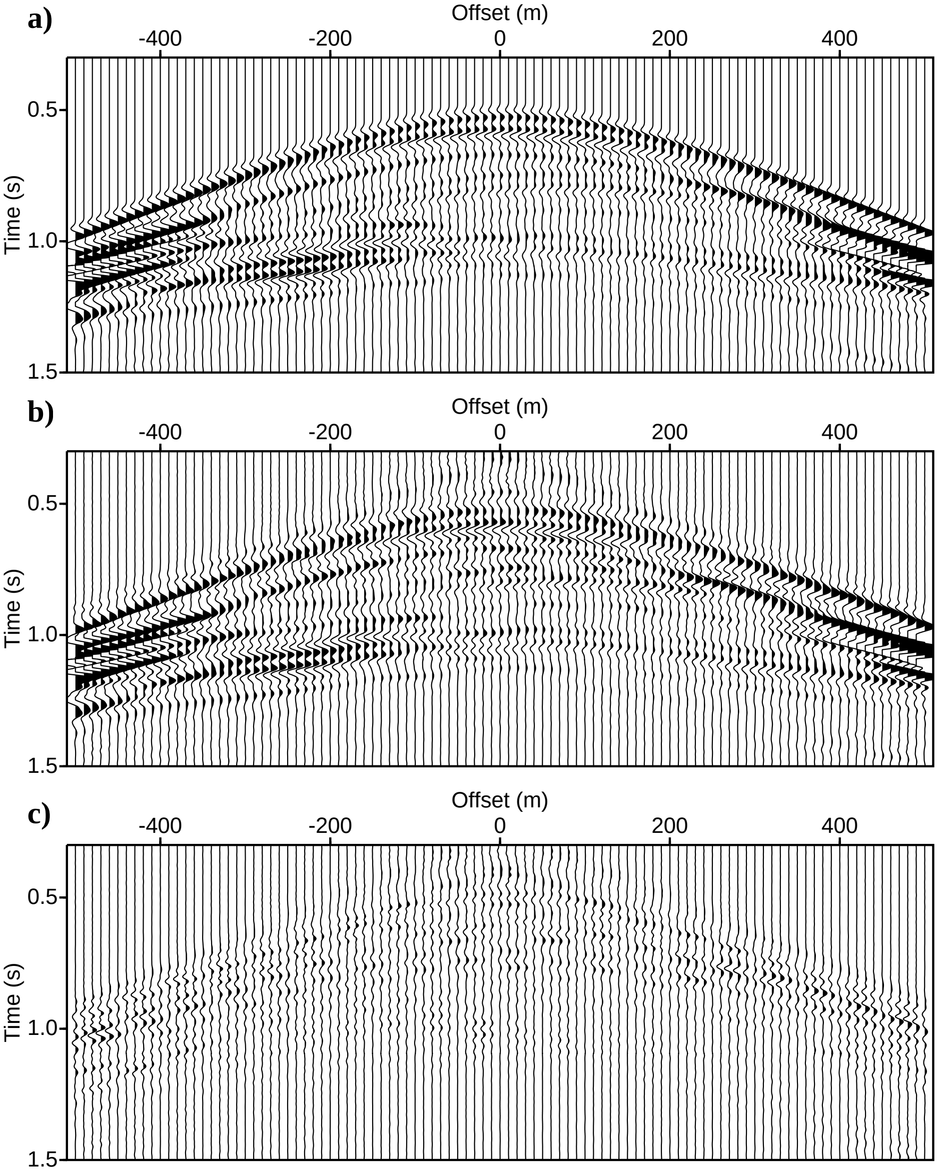

Figure 6.11: Synthetic data example: a) data synthesized using the finite difference approximation and the known velocity model, b) the SPDR reconstructed data computed from the data in Figure 6.10b, c) the difference between the finite difference data in a) and the SPDR reconstructed data in b). In all plots, every 4^{th} trace in the nominal grid is shown for small offsets.

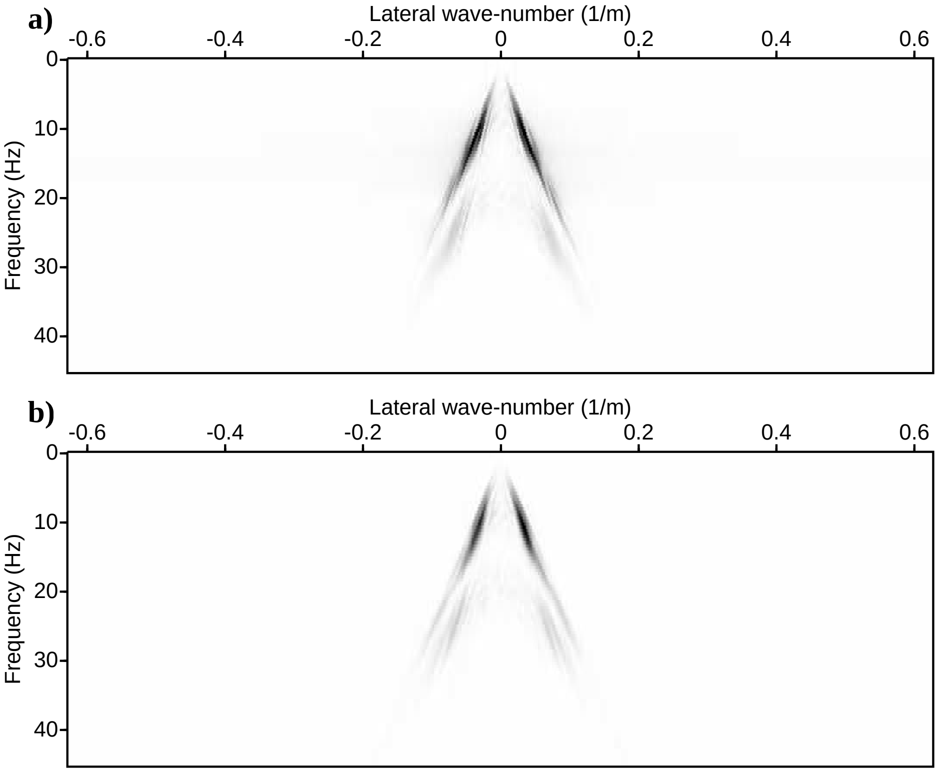

Figure 6.12: Synthetic data example: $f - k$ spectrum of a) the finite difference data in Figure 6.11a, and b) the SPDR reconstructed data in Figure 6.11b. For comparison, consider the $f - k$ spectrum of the observed data in Figure 6.10c.

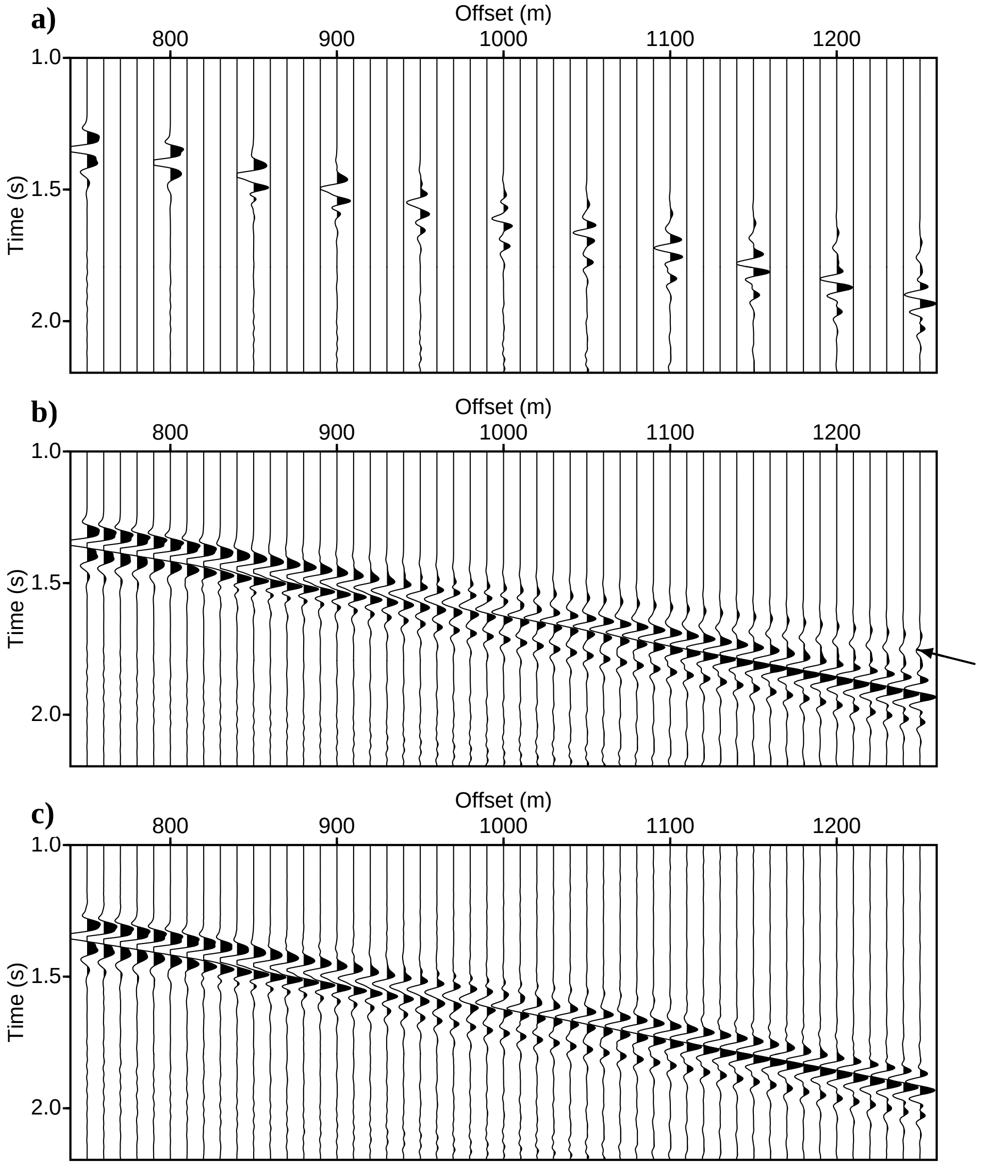

Figure 6.13: Synthetic data example, illustrating that the direct wave is in the null-space of the de-migration operator. We show far offset traces from a) the recorded data, b) data synthesized using the finite difference approximation to the acoustic wave-equation, and c) the SPDR reconstructed data. The direct arrival in the finite difference data is indicated by the arrow in b).

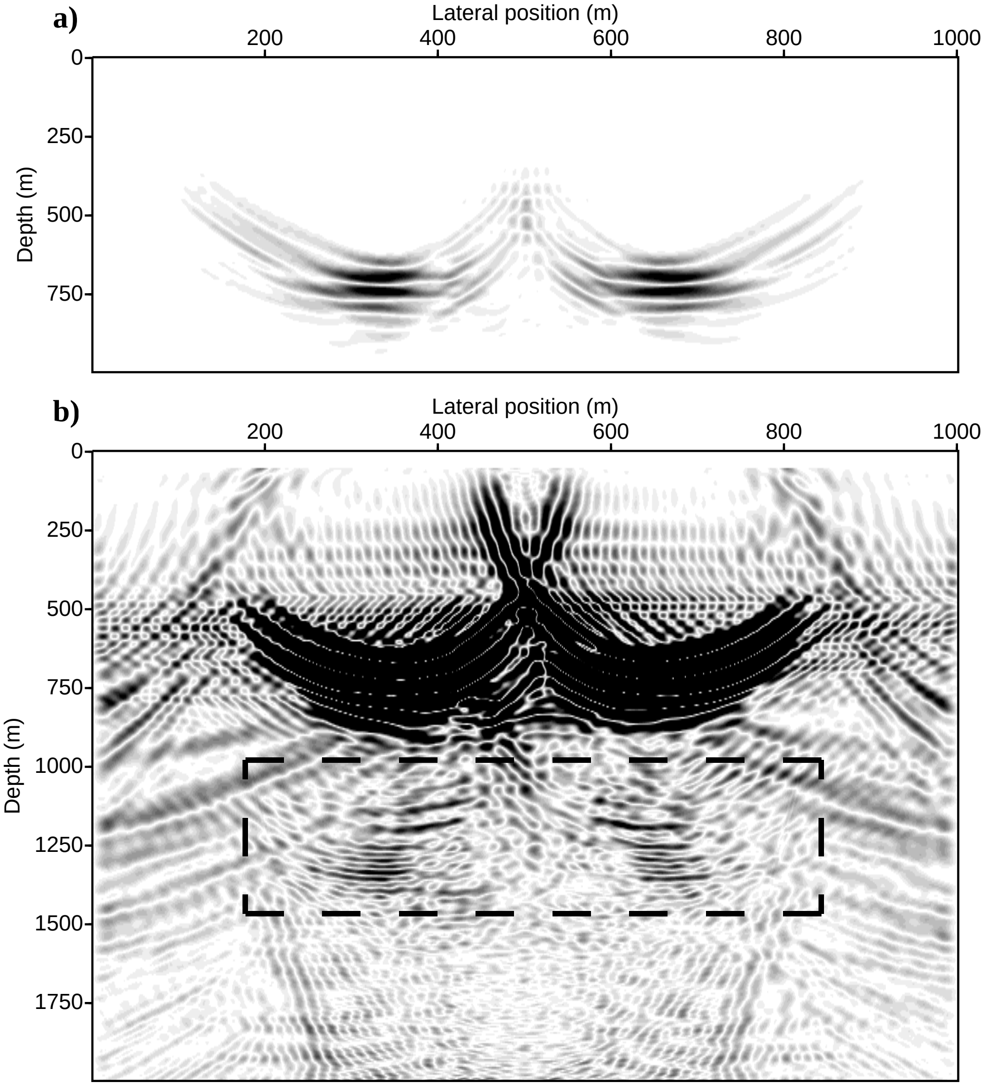

Figure 6.14: Synthetic data example: a) The inverse $\mathbf{m}_*$ computed for when the model space has a maximum depth of $1000m$, b) the inverse $\mathbf{m}_*$ computed for when the model space has a maximum depth of $2000m$. The multiple energy in b) is indicated by the dashed box.

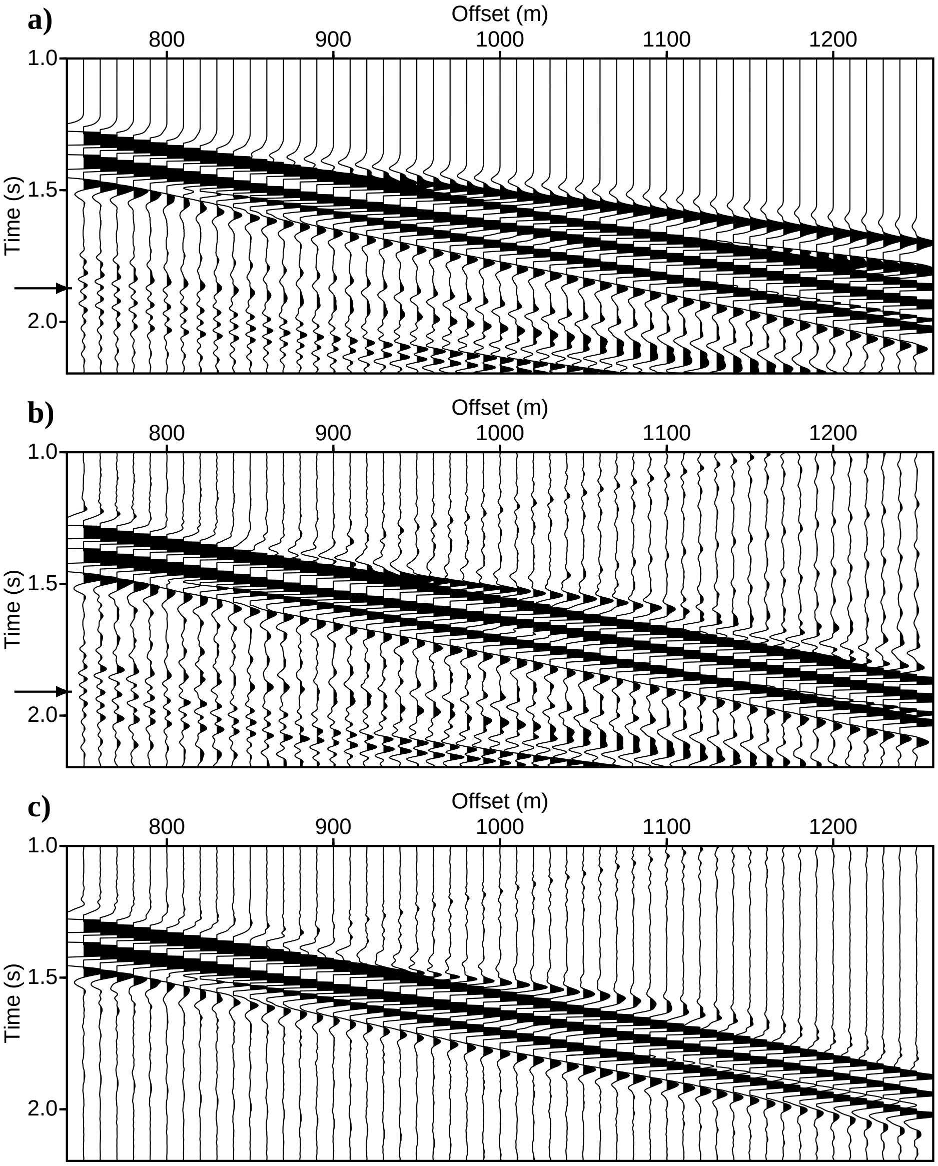

Figure 6.15: Synthetic data example, illustrating the data reconstruction of free-surface multiples: a) data generated for the nominal grid using the finite difference approximation to the wave-equation, b) the SPDR reconstructed data using a model space to 2000m depth, c) the SPDR reconstructed data using a model space to 1000m depth. The multiples are indicated by arrows in a) and b) and are absent from c).

4874.7m from the source. In total there are 183 recorded geophones. In our experiment, we use a nominal grid with 365 grid points spaced every 13.3m. Finally, for our experiment, we decimate the data so that the geophones are spaced every 80.0m. In other words, there are 5 nominal grid points between each point on the observed grid, and that need to be interpolated.

In Figure 6.16, we plot results from the SPDR algorithm. In Figure 6.16a, we plot the original real data on its original observational grid with 26.7m spacing. In Figure 6.16b, we plot the data after decimation, and on its nominal grid with 13.3m spacing. Finally, we show the SPDR result in Figure 6.16c. The result is plotted using the 26.7m grid spacing of the original data. This allows us to compare the SPDR result to the observed data. In particular, Figure 6.16d plots the difference between the data in Figure 6.16a and the SPDR result in Figure 6.16c. Note that before plotting, we apply an automatic gain control algorithm to emphasize the small amplitude events. In addition, the gained data are clipped to 33% of the largest magnitude. In Figure 6.17, we plot the corresponding $f - k$ spectra. In particular, Figure 6.17a plots the $f - k$ spectrum of the original data on the 26.7m grid, Figure 6.17b plots the $f - k$ spectrum of the decimated data (80.0m grid), and Figure 6.17c plots the $f - k$ spectrum of the SPDR reconstructed data.

In Figures 6.18a-c, we plot various realization from model space, and in Figures 6.18d-f, we plot their wave-number spectra. The first column (Figures 6.18a and 6.18d) plots the adjoint of the original data (on its 20.7m grid), while the second column (Figures 6.18b and 6.18e) plots the adjoint of the decimated data (from the 80.0m observational grid). The alias resulting from the decimated data is evident. Finally, the third column (Figures 6.18c and 6.18f) plots the weighted least-squares inverse of the decimated data, and which is used in producing the SPDR result in Figure 6.16c. In applying SPDR, we set $\eta = 0.09$, $\tau = 0.01$ and $\hat{\xi} = 0$, the effect of which is seen in the comparison of Figures 6.18e and 6.18f. Additionally, we used 60 conjugate gradient iterations to find the solution of the least-squares normal equations. We calculate a signal-to-noise ratio (SNR) for the reconstructed data finding,

$$\mathrm{SNR} = 10 \log_{10} \frac{||\mathbf{d}_1||_2^2}{||\mathbf{d}_1 - \mathbf{d}_*||_2^2} = 11\mathrm{dB}. \tag{6.30}$$

where $\mathbf{d}_1$ is the original data before decimation (Figure 6.16a), and $|| \cdot ||_2$ is the $L2$ norm computed for an x_g grid common to both $\mathbf{d}_1$ and $\mathbf{d}_*$. For comparison, the signal to noise ratio for the first synthetic example, presented in Figure 6.7, is 16.0dB.

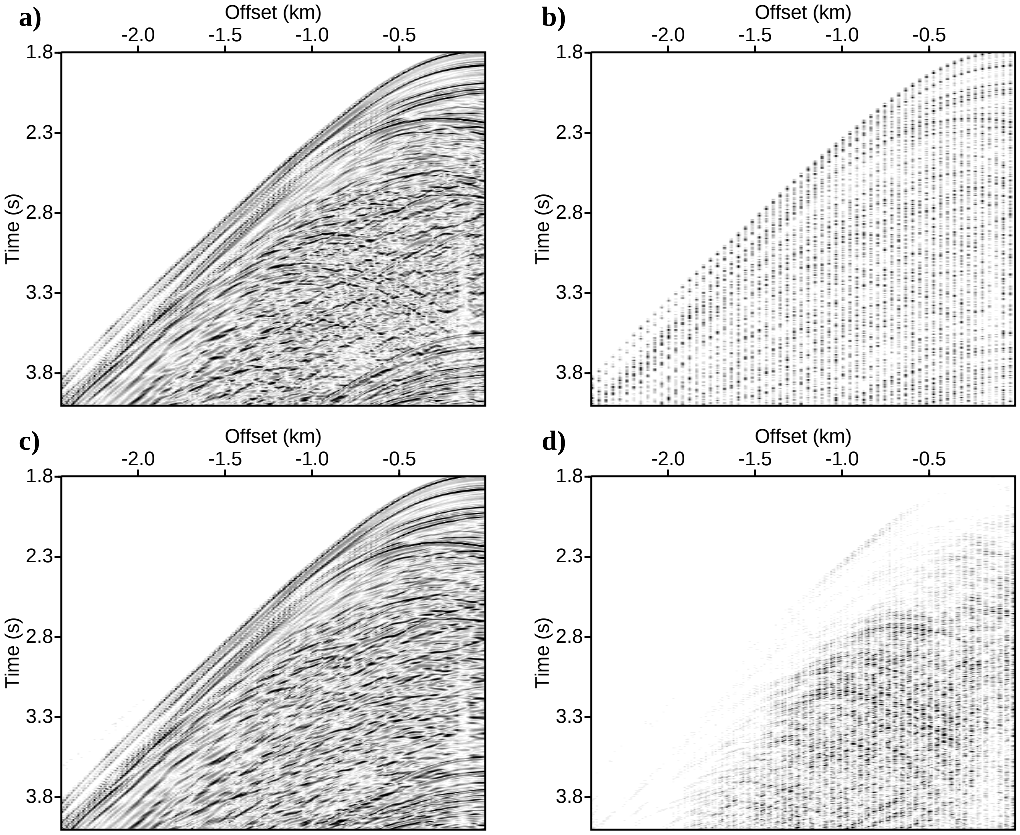

Figure 6.16: Real data example: a) the original data, b) the decimated data, c) the reconstructed data, d) the residual, or difference between c) and a). An automatic gain control algorithm is applied prior to plotting. In addition, we clip the data at 33% of the largest magnitude. All four plots use the same colour scale.

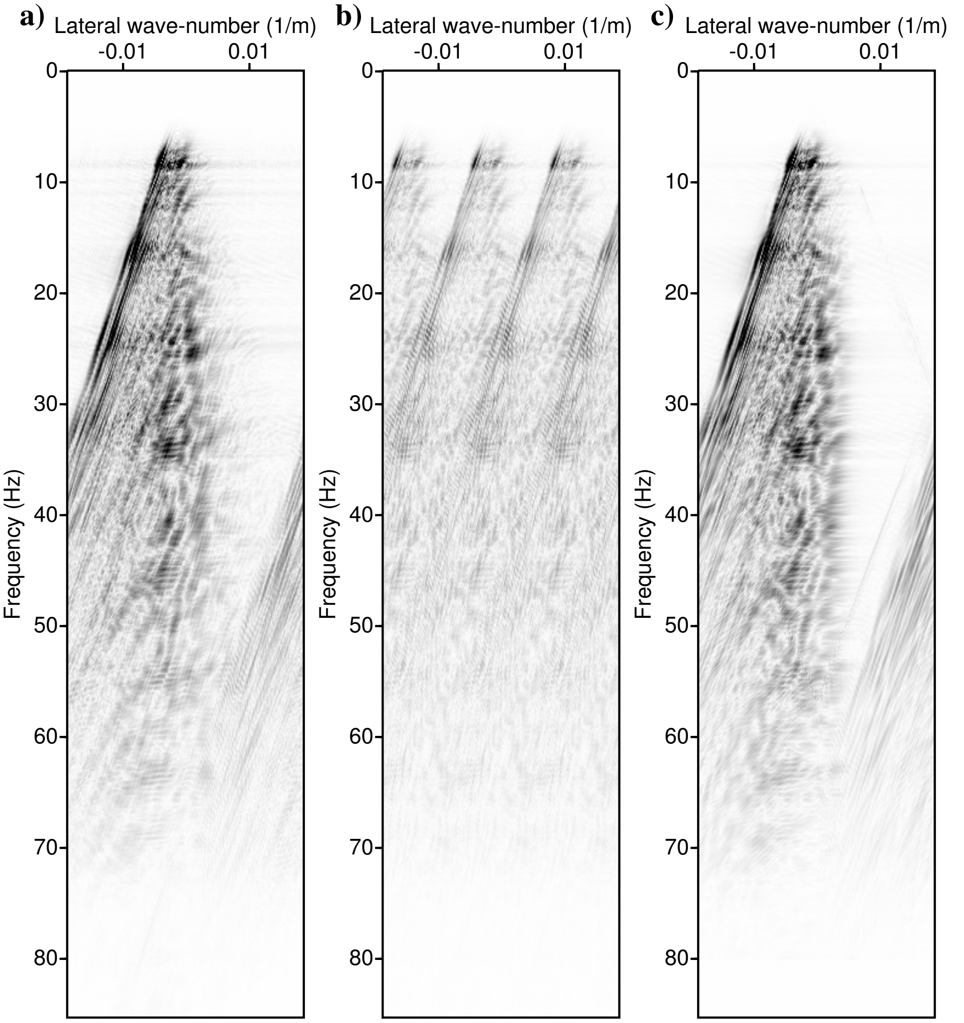

Figure 6.17: Real data example: $f-k$ spectra of a) the original data, b) the decimated data, and c) the reconstructed data.

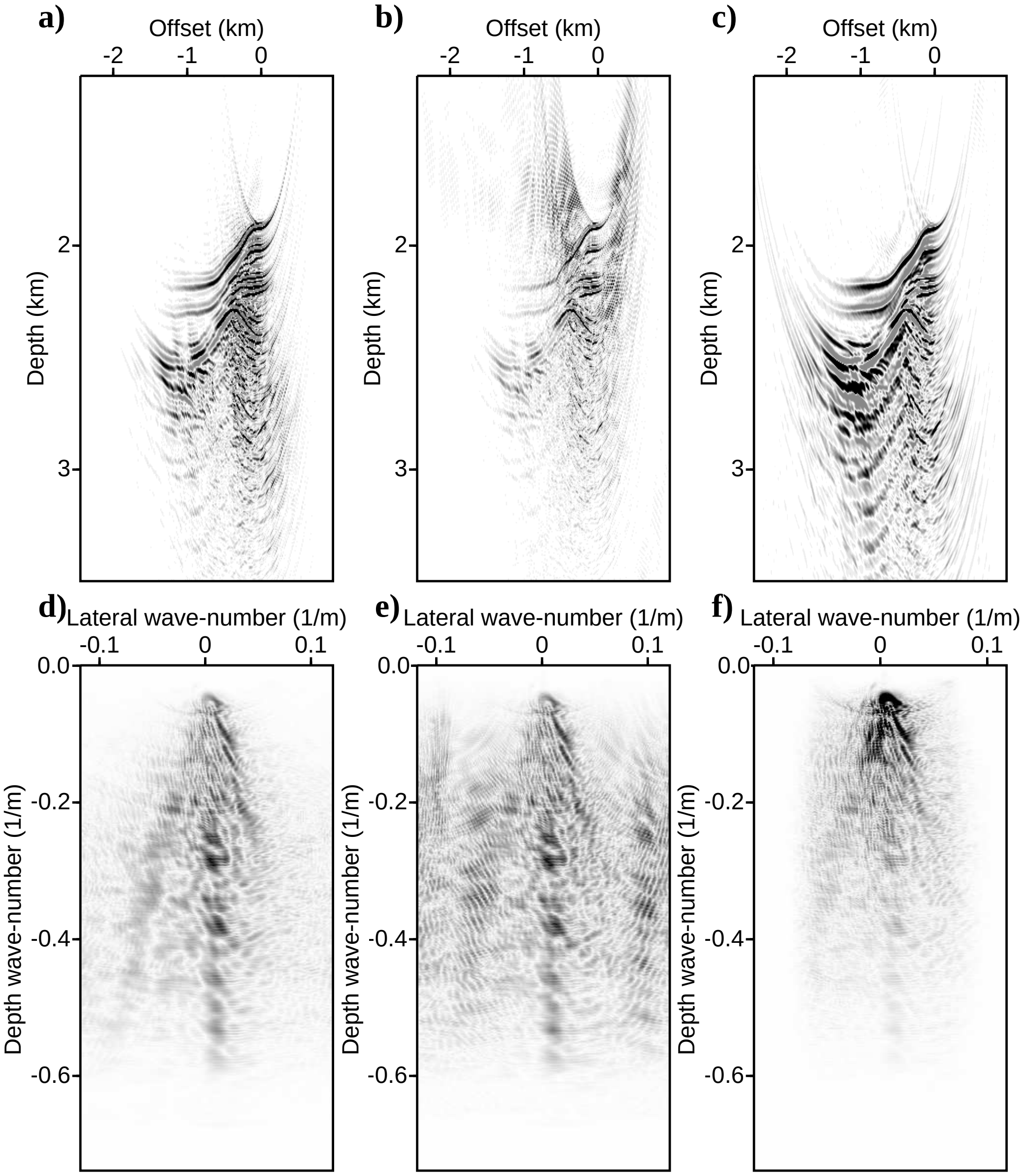

Figure 6.18: Real data example: the model space representation of a) the original data and b) the decimated data, both using the adjoint operator, and c) the decimated data under the weighted least-squares inversion. The corresponding wave-number spectra are shown in d) for the original data and e) for the decimated data, both using the adjoint operator, and f) the decimated data using the weighted least-squares inversion.

6.9 2D shot profile data reconstruction

SPDR is a 1D data reconstruction algorithm meaning that for any given shot gather, it reconstructs data along one geophone dimension. We extend SPDR to a 2D shot-profile data reconstruction algorithm (SPDR2). In particular, SPDR2 reconstructs data in both geophone and shot dimensions. Analogous to SPDR, we restrict ourselves to data reconstruction in one shot and one geophone dimension, namely x_s and x_g, while assuming that data are invariant in the y_g and y_s dimensions. The extension of SPDR to SPDR2 requires the corresponding extensions of the SPDR forward and adjoint operators to their SPDR2 counter-parts, and their use in a least-squares data reconstruction method. In this section, we describe these extensions, and provide an analysis of the adjoint operator analogous to Section 6.4.

6.9.1 Extension of forward and adjoint operators to 5D

In Section 6.2, we considered the de-migration (forward) and migration (adjoint) operators used in SPDR. The forward operator for SPDR maps from a prescribed model space to a prescribed data space. We recall that the data space in SPDR is a single common shot data gather, and is a function of time and lateral dimensions $\mathbf{x}_g$. Likewise, the model space in SPDR is single common shot image gather, and is a function of depth z and the same lateral dimensions $\mathbf{x}_g$. For SPDR2, the operators add two dimensions to data space allowing the shot location $\mathbf{x}_s = (x_s, y_s)$ to vary. In contrast, the model space for SPDR2 is the same as it was for SPDR. In particular, we construct operators so that model space domain remains three-dimensional, described by lateral coordinates $\mathbf{x}_g = (x_g, y_g)$ and depth z.

The forward operator is given by equation 6.1, except with the assertion that ψ_s is a function of $\mathbf{x}_s$, rather than being merely parameterized by it. We assume a finite number of shot locations, and let $\mathbf{x}_{s(q)}$ denote the location of the q^{th} shot, so that,

$$\begin{aligned}\psi_s(\mathbf{x}_g,&\omega, \mathbf{x}_{s(q)})\\ &= \left(\frac{1}{2\pi}\right)^4 \left(\frac{\omega}{c_0}\right)^2 \mathcal{F}_g^* \int_{z_0}^{\infty} u_p(\mathbf{k}_{gx}, z', \omega) \mathcal{F}_g \left[\mathcal{F}_g^* u_p(\mathbf{k}_{gx}, z', \omega) g(\mathbf{k}_{gx}, \mathbf{x}_{s(q)}, \omega)\right] \alpha(\mathbf{x}_g, z') dz'.\end{aligned} \tag{6.31}$$

Equation 6.31 is an extension of equation 6.1, and can be implemented in a similar fashion. We recall that u_p is given by equation 6.2, $g(\mathbf{k}_{gx}, \mathbf{x}_{s(q)}, \omega)$ by equation 5.7, and $\mathcal{F}_g$ and $\mathcal{F}_s$ are the two dimensional unnormalized Fourier transforms over lateral coordinates. To construct the adjoint (shot-profile migration) operator corresponding to the forward operator in equation 6.31, we will need to consider the effect of the extra dimension $\mathbf{x}_{s(q)}$ in ψ_s. In particular, we introduce an operator $\mathcal{H}$, so that the forward operator in equation 6.31 can be expressed using matrix-vector notation,

$$\begin{bmatrix} \psi_s(\mathbf{x}_g,\omega,\mathbf{x}_{s(1)}) \\ \psi_s(\mathbf{x}_g,\omega,\mathbf{x}_{s(2)}) \\ \vdots \\ \psi_s(\mathbf{x}_\mathbf{g},\omega,\mathbf{x}_{s(n_s)}) \end{bmatrix} = \begin{bmatrix} \mathcal{H}(\mathbf{x}_{s(1)}) \\ \mathcal{H}(\mathbf{x}_{s(2)}) \\ \vdots \\ \mathcal{H}(\mathbf{x}_{s(n_s)}) \end{bmatrix} \alpha(\mathbf{x}_g,z'), \tag{6.32}$$

where we have assumed n_s shots. From equation 6.31, we recognize that the operator $\mathcal{H}$ must be defined so that,

$$\begin{aligned} &\mathcal{H}(\mathbf{x}_{s(q)})\alpha(\mathbf{x}_g,z') \\ &\quad= \left(\frac{1}{2\pi}\right)^4 \left(\frac{\omega}{c_0}\right)^2 \mathcal{F}_g^* \int_{z_0}^{\infty} u_p(\mathbf{k}_{gx},z',\omega)\mathcal{F}_g\left[\mathcal{F}_g^* u_p(\mathbf{k}_{gx},z',\omega)g(\mathbf{k}_{gx},\mathbf{x}_{s(q)},\omega)\right]\alpha(\mathbf{x}_g,z')dz'. \end{aligned} \tag{6.33}$$

It follows directly from equations 6.32 and 6.33 that the corresponding adjoint operation is,

$$\alpha^\dagger(\mathbf{x}_g,z') = \begin{bmatrix} \mathcal{H}^*(\mathbf{x}_{s(1)}) & \mathcal{H}^*(\mathbf{x}_{s(2)}) & \cdots & \mathcal{H}^*(\mathbf{x}_{s(n_s)}) \end{bmatrix} \begin{bmatrix} \psi_s(\mathbf{x}_g,\omega,\mathbf{x}_{s(1)}) \\ \psi_s(\mathbf{x}_g,\omega,\mathbf{x}_{s(2)}) \\ \vdots \\ \psi_s(\mathbf{x}_g,\omega,\mathbf{x}_{s(n_s)}) \end{bmatrix}, \tag{6.34}$$

and where,

$$\begin{aligned} &\mathcal{H}^*(\mathbf{x}_{s(q)})\psi_s(\mathbf{k}_{gx},\omega,\mathbf{x}_{s(q)}) = \left(\frac{1}{2\pi}\right)^4 \\ &\quad\times \int_{-\infty}^{\infty} \left(\frac{\omega}{c_0}\right)^2 \left[\mathcal{F}_g^* u_p^*(\mathbf{k}_{gx},z,\omega)g^*(\mathbf{k}_{gx},\mathbf{x}_{s(q)},\omega)\right]\mathcal{F}_g^* u_p^*(\mathbf{k}_{gx},z,\omega)\psi_s(\mathbf{k}_{gx},\omega,\mathbf{x}_{s(q)})d\omega. \end{aligned} \tag{6.35}$$

In short, the adjoint operator includes a sum over shot gathers such that,

$$\begin{aligned} &\alpha^\dagger(\mathbf{x}_g,z') = \left(\frac{1}{2\pi}\right)^4 \\ &\quad\times \sum_{q=1}^{n_s} \int \left(\frac{\omega}{c_0}\right)^2 \left[\mathcal{F}_g^* u_p^*(\mathbf{k}_{gx},z',\omega)g^*(\mathbf{k}_{gx},\mathbf{x}_{s(q)},\omega)\right]\mathcal{F}_g^* u_p^*(\mathbf{k}_{gx},z',\omega)\mathcal{F}_g\psi_s(\mathbf{x}_g,\omega,\mathbf{x}_{s(q)})d\omega. \end{aligned} \tag{6.36}$$

In other words, $\alpha^\dagger(\mathbf{x}_g,z') = \sum_q \alpha^\dagger(\mathbf{x}_g,z';\mathbf{x}_{s(q)})$, and where $\alpha^\dagger(\mathbf{x}_g,z';\mathbf{x}_{s(q)})$ was previously defined in equation 6.5. This formulation is similar to what was shown in Section 5.5.

Equations 6.31 and 6.36 define, respectively, the $5D$ forward and adjoint operators for

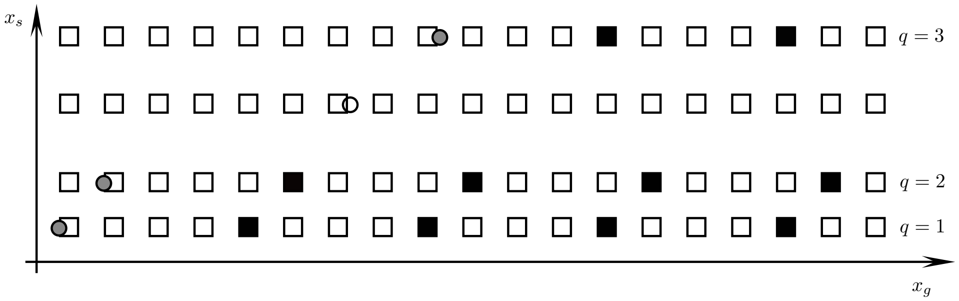

Figure 6.19: 2D schematic of the nominal and observed grids for SPDR2. The squares are the nominal grid x_g^n, the black filled squares are the observed grid $x_g^{o(q)}$ for $i = 1 \dots 3$ (three observed shots). The grey filled circles are the observed shot locations $x_{s(q)}$ for $q = 1 \dots 3$ relative to the geophone axis. The empty circle represents the location of a missing shot gather that will be reconstructed using SPDR2.

SPDR2. We have made an important choice in the construction of the $5D$ forward and adjoint operators. Namely that model space is not a function of shot location. This choice is integral to SPDR2. It allows for the data reconstruction of missing data using information from known and near-by shot gathers.

6.9.2 Observed data on a $2D$ nominal grid

SPDR2 reconstructs missing data in both geophone and shot dimensions. In the geophone dimension, SPDR and SPDR2 share the same requirements. Namely that we define some nominal and observed grid. As stated earlier, we will assume that data and model are invariant to y_g, and that, additionally, the data are invariant to y_s. For all shots, the nominal grid x_g^n is invariant to shot location $x_{s(q)}$. On the other hand, the observed grid necessarily varies with shot location due to the acquisition geometry of the seismic experiment so that for the q^{th} shot, we write $x_g^{o(q)}$. There is no analogous nominal grid for the shot dimension. That is, both the observed and reconstructed shots can be located arbitrarily along the shot dimension. The caveat being that the aperture of the common shot image gather corresponding to the reconstructed shot must be sufficiently illuminated by the observed shots. This caveat will be illustrated when we consider the reconstruction of missing near offset traces in Section 6.9.6. We give a schematic representation of the nominal grid x_g^n for multiple shot locations in Figure 6.19. The empty and filled boxes in Figure 6.19 represent the nominal grid x_g^n, the black filled boxes represent the observed grid $x_g^{o(q)}$ (i.e. data traces), the grey filled circles represent the observed shot locations relative to the x_g axis, and the empty circle represents a shot location for SPDR2 to reconstruct relative to the geophone axis. We note that the shot locations need not fall on the nominal geophone grid.

6.9.3 SPDR2 adjoint mapping of signal and alias due to the observed grid

We analyze the adjoint operator in equation 6.36 with an analytic example. This is an extension of the analysis in Section 6.4. Our goal is to show that the understanding obtained in Section 6.4 for the single shot adjoint operator is roughly equivalent to the understanding of the multi-shot adjoint operator (equation 6.36). In particular, when the adjoint operator is applied to the sampling function, the shape of the sampling function in model space are the same, except that multiple shot gathers introduce an interference pattern along the k_{gx} dimension.

We proceed in analogy to Section 6.4, letting $\bar{\psi}_s(x_g, \omega, x_{s(q)}) = III_o(x_g)\psi_0(\omega)$ for $q = 1 \dots n_s$, so that $\bar{\psi}_s(k_{gx}, \omega, x_{s(q)})$ is defined by equation 6.18. Then, we substitute $\bar{\psi}_s(k_{gx}, \omega, x_{s(q)})$ into the adjoint operator in equation 6.36, finding an expression that is analogous to equation 6.16,

$$\bar{\alpha}^\dagger(x_g, z) = \sum_p \int_{-\infty}^{\infty} \alpha_p^\dagger(x_g, z, \omega) d\omega, \tag{6.37}$$

but where,

$$\begin{aligned}\alpha_p^\dagger(x_g, z, \omega) &= \left(\frac{1}{2\pi}\right)^2 \left(\frac{\omega}{c_0}\right)^2 \\ &\times \sum_{q=1}^{n_s} \left[\mathcal{F}_g^* u_p^*(k_{gx}, z, \omega) g^*(k_{gx}, x_{s(q)}, \omega) \mathcal{F}_g\right] \mathcal{F}_g^* u_p^*(k_{gx}, z, \omega) \psi_s(k_{gx} + p2\pi/\Delta x_g^{o(q)}, \omega, x_{s(q)}).\end{aligned} \tag{6.38}$$

Then, we substitute ψ_s for the Shah function in equation 6.18, finding (see Appendix E.2),

$$\begin{aligned}&\alpha_p^\dagger(k_{gx}, \kappa_z, \omega) \\ &= \frac{f^*(\omega)\psi_0(\omega)}{\Delta x_g^o} \left(\frac{\omega}{c_0}\right)^2 \delta\left(\kappa_z + (k_{gz(p)} + sgn(\omega)\sqrt{\omega^2/c_0^2 - (k_{gx} - k_p)^2})\right) \sum_{q=1}^{n_s} e^{i(k_{gx} - k_p)x_{s(q)}},\end{aligned} \tag{6.39}$$

where, again, κ_z is the Fourier conjugate variable of z. Equation 6.39 can be thought of as a generalization of equation 6.19. That is, if in equation 6.39 we use a single shot gather such that $n_s = 1$ and $x_{s(1)} = 0$, then we find equation 6.19. The generalization in equation 6.39 introduces the sum over shot location, and an interference pattern into the wave-number spectrum of $\alpha_p^\dagger$ in the k_{gx} dimension.

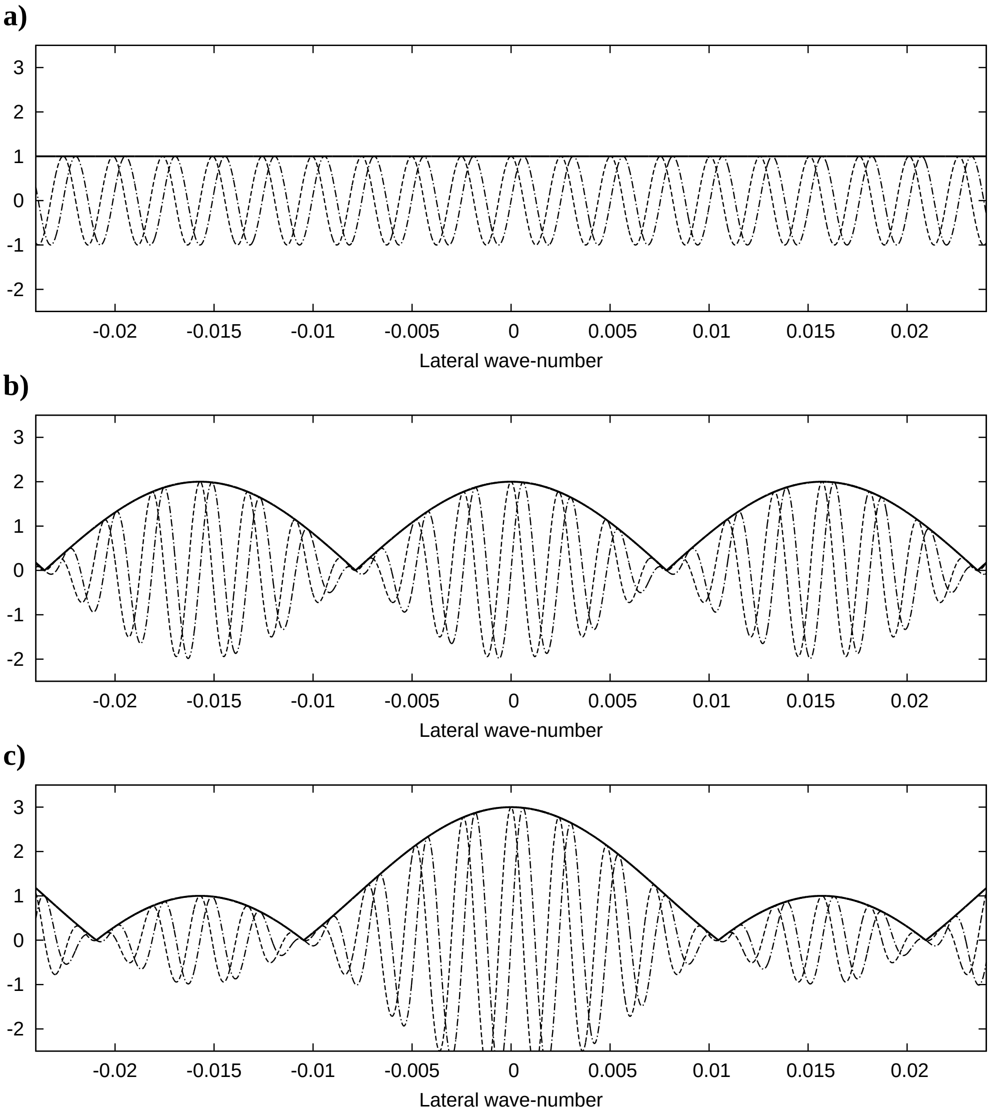

Figure 6.20: Interference patterns $I(k_{gx})$ for a) one ($n_s = 1$), b) two ($n_s = 2$) and c) three ($n_s = 3$) terms in equation 6.40. We plot the real (dashed line) and imaginary (dash-dot line) components of $I(k_{gx})$, along with the amplitude $|I(k_{gx})|$ (solid line).

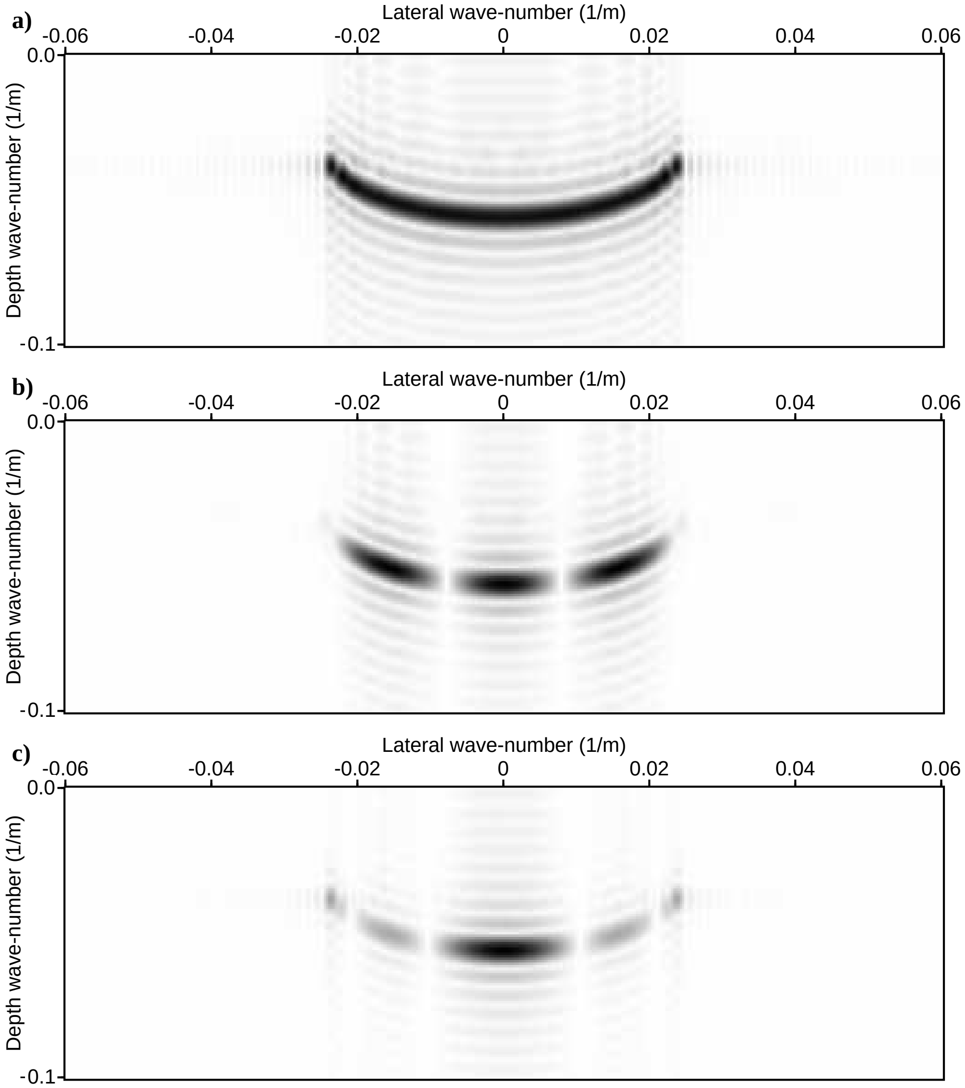

Figure 6.21: Synthetic example for a single frequency using a) one shot gather ($n_s = 1$), b) two shot gathers ($n_s = 2$) and c) three shot gathers ($n_s = 3$). The interference patterns in the lateral wave-number dimension correspond to the analytic predictions shown in Figure 6.20.

We illustrate the interference pattern in Figures 6.20 and 6.21. In Figure 6.20a, 6.20b and 6.20c, we plot the interference pattern for, respectively, $n_s = 1$, $n_s = 2$ and $n_s = 3$. Namely we plot,

$$I(k_{gx}) = \sum_{q=1}^{n_s} e^{i(k_{gx}-k_p)x_{s(q)}}, \tag{6.40}$$

for $p = 0$. In each plot, the dashed line is the real component of $I(k_{gx})$, the dot-dashed line is the imaginary component, and the solid line is the amplitude, $|I(k_{gx})|$. In Figure 6.21, we show results for the equivalent synthetic experiment. In particular, we compute $\alpha^\dagger$ for a single frequency $6Hz$. We plot the adjoint for when the data consists of one (Figure 6.21a), two (Figure 6.21b) and three (Figure 6.21c) shot gathers. Notice that the interference pattern in the numerical results are predicted by the analytic example.

6.9.4 SPDR2 least-squares data reconstruction

Applying the SPDR2 adjoint operator (equation 6.36) to the sampling function produced a similar pattern to the SPDR adjoint operator (equation 6.5), with the exception of the interference pattern in the k_{gx} dimension. Therefore, we choose to assume that the least-squares data reconstruction algorithm shown for SPDR in Section 6.6 is, with some modification, also applicable to SPDR2. The modifications include a straight-forward redefinition of the data vector $\mathbf{d}$, and the corresponding mappings given by the matrices $\mathbf{A}$ and $\mathbf{A}^H$. In addition the data weights $\mathbf{W}_d$ will be trivially extended to accommodate multiple shot gathers; but, the model $\mathbf{m}$ and model weights $\mathbf{W}_m$ will remain unchanged. The interference pattern introduced by the SPRD2 adjoint operator seems to imply that a different model regularization is required. However, we have found in practise that the SPDR model regularization sufficiently accounts for the interference pattern introduced into the SPDR2 model space. We note that the model space dip limitation ($h(k_{gx})$ in equation 6.21) will help mitigate the effect of the interference pattern. To avoid notational clutter, we will reuse the notation of SPDR least-squares data reconstruction in our description of SPDR2 least-squares data reconstruction.

Like SPDR, SPDR2 finds the solution to equation 6.24. However, in SPDR2 $\mathbf{d}$ is a data vector of length M realized from n_s observed shot gathers $\psi_s(x_g^n, \omega, x_{s(q)})$ so that $M = n_s n_\omega n_g^n$, and as before there are n_g^n geophones on the nominal grid. The forward and adjoint operators are evaluated for n_ω realization of frequency. The model vector $\mathbf{m}$ is as it was in SPDR algorithm so that $N = n_z n_g^n$ where n_z are the number of depth points in $\alpha(x_g, z)$. Then, we let $\mathbf{A}$ be the $M \times N$ matrix representation of the forward operator (shot-profile de-migration in equation 6.31), and $\mathbf{A}^H$ its corresponding adjoint operator (shot-profile migration in equation 6.36). Then, as before, $\mathbf{A}$ maps from $\mathbf{m}$ to $\mathbf{d}$.

The extension to the data weights $\mathbf{W}_d$ for SPDR2 is straight forward so that in equation 6.25, $\mathcal{I}_d$ is the set of indices corresponding to the observed grid $x_g^{o(q)}$ for $q = 1 \dots n_s$. The model weights $\mathbf{W}_m$ remain unchanged and are given by equations 6.26-6.29.

Once equation 6.24 is solved in the context of SPDR2, the forward (de-migration) operator in equation 6.31 can be applied to reconstruct common shot data gathers for arbitrary shot locations $x_{s(q)}$, and geophones that fall on the nominal grid x_g^n. Again, the caveat is that any reconstructed shot must have its corresponding common shot image gather sufficiently illuminated by the observed shots. We illustrate the method with one synthetic example and two real data examples.

6.9.5 Synthetic data example

For a synthetic example, we consider the acoustic velocity model in Figure 6.10a. Using the finite difference approximation to the acoustic wave-equation, we generate the 13 shot gathers shown in Figure 6.22a. The shot gather locations are given by,

$$x_{s(q)} = 2300m + (q-1)50m \quad , \quad q = 1 \dots 13.$$

For all shots, 1001 geophones are spaced every $5m$ between 0 and $5000m$. We decimate the data to give Figure 6.22b. In particular, we decimate the data by removing all but 4 shots, so that the distance between each shot is $200m$. Additionally, we remove geophones so that there are 51 geophones spaced every $100m$. We use SPDR2 to reconstruct the missing data in Figure 6.22b, showing the result in Figure 6.22c. The input to the SPDR2 algorithm is the four observed shot gathers on the nominal geophone grid. We regularize the model space by setting $\mathbf{W}_m$ such that in equations 6.28 and 6.29, $\eta = 0.06k_n$, $\tau = 0.01k_n$, and $\xi = 0.2$, and k_n is the Nyquist wave-number for the nominal geophone grid. Finally, we set the trade-off parameter in the least-squares normal equations (equation 6.24) to $\mu = 0.1$. Note that, η and τ were chosen from an interpretation of the adjoint model, and in which the location of signal and alias can be interpreted according the analysis in Sections 6.4 and 6.9.3. In Figure 6.23, we plot the $f-k$ spectra of the data, decimated data and reconstructed data. The $f-k$ spectra illustrates the presence of aliased events in the decimated data, and the lack-there-of in the original and reconstructed data.

6.9.6 Real data examples

For real data examples, we revisit the Gulf of Mexico data from Section 6.8. For our first example, we consider five shot gathers from this data, and that are shown in Figure 6.24b. The central shot gather was also used in Section 6.8 for the SPDR example. The shots are

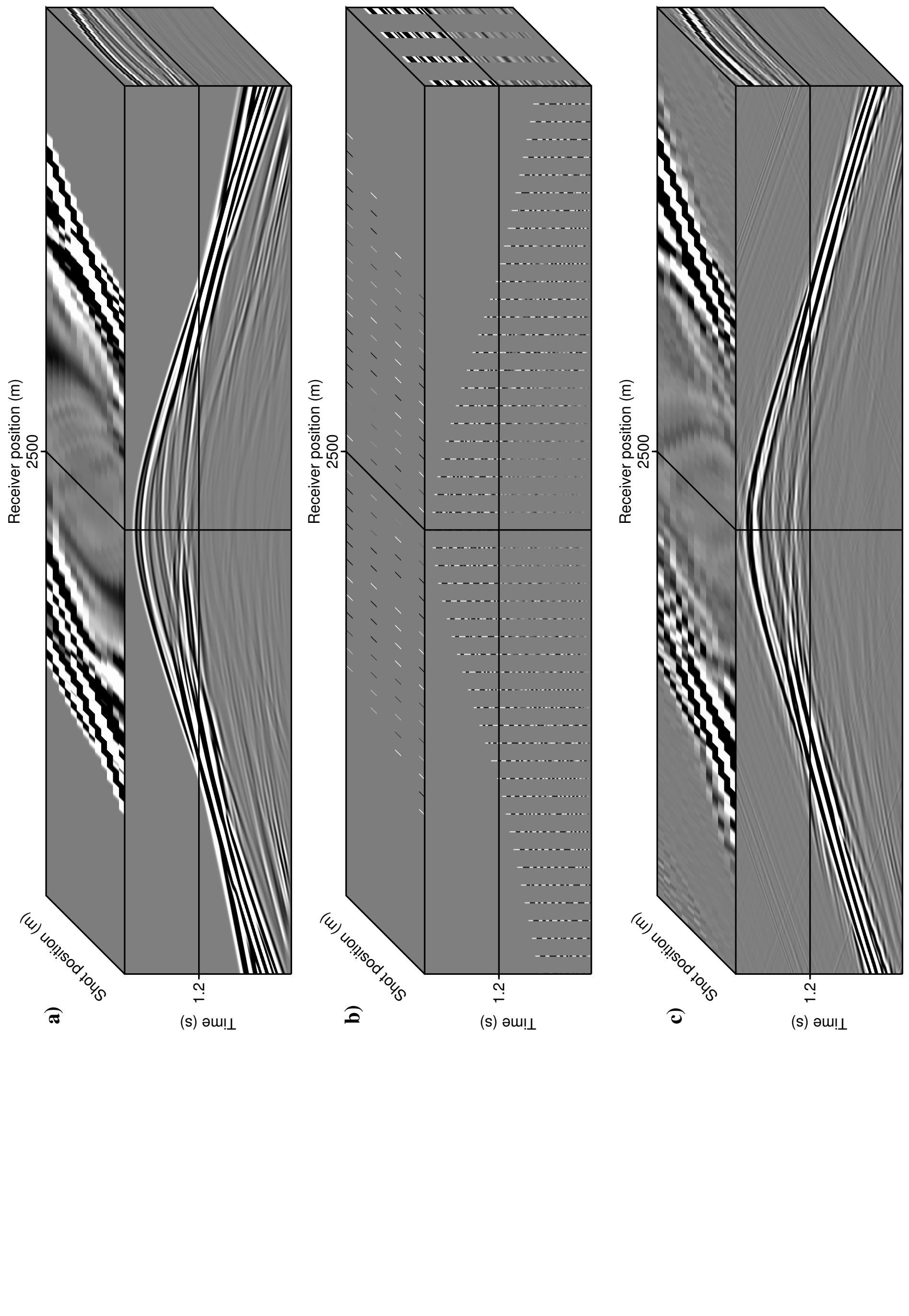

Figure 6.22: 2D synthetic data example: a) the finite difference data, b) the decimated finite difference data, c) the SPDR2 reconstructed data using the data from b).

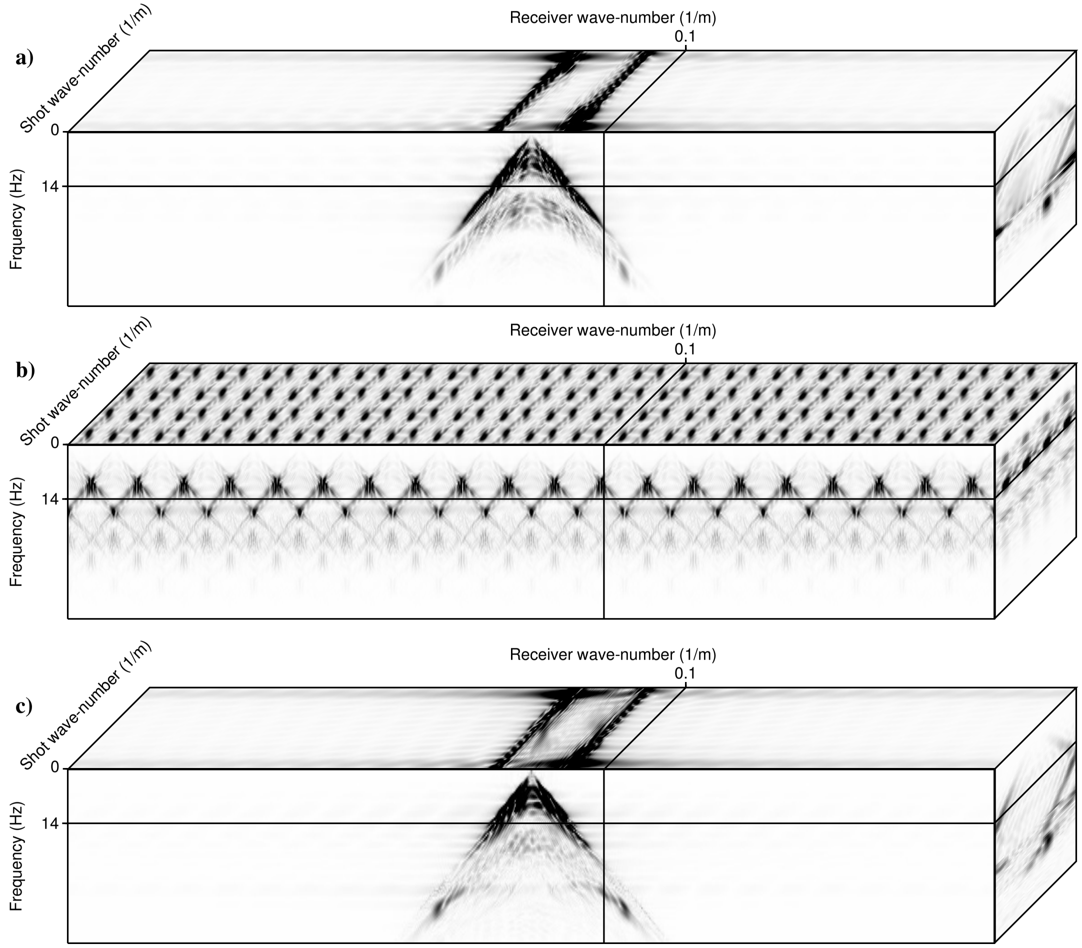

Figure 6.23: 2D synthetic data example: wave-number spectra of a) the finite difference data, b) the decimated finite difference data, c) the SPDR2 reconstructed data using the data from b).

spaced every $53.34m$, and the geophones are spaced every $26.67m$. We decimate the data to give Figure 6.24a. In particular, we remove every other shot gather so that the distance between observed shots is $106.68m$. We remove geophones so that the distance between observed geophones is $80.01m$. The observed shot gathers in Figure 6.24a are input to SPDR2, and the SPDR2 reconstructed data are shown in Figure 6.24b. The SNR of the reconstruction is 6dB (see equation 6.30). For comparison, we plot the original data, before decimation, in Figure 6.24c.

For our second example, we consider the reconstruction of near-offset traces. The successful reconstruction of near-offset traces is important to seismic data processing techniques such as surface related multiple elimination (e.g. Ramírez et al., 2006). We use the 6 shot gathers shown in Figure 6.25 to illustrate the effectiveness of SPDR2 for the data reconstruction of near-offset traces. Note that in Figure 6.25, we show a subset of these offsets. The shot gathers have geophones spaced every $26.67m$ with offsets running from $-2.43km$ to $-10.37m$. The shots are spaced every $53.34m$ so that from left to right in Figure 6.25, the shot location is increasing (see Figure 6.26a for a schematic representation of the geometry of the first four shot gathers). This is typical of a towed-streamer acquisition geometry, and is an important point in constructing an understanding of why SPDR2 can be used to reconstruct near-offset traces. The data in Figure 6.25 is used by SPDR2 with the goal of reconstructing the near-offset data for the first (left-most) shot gather, but using information from all six shot gathers. The argument for this is as follows. If SPDR2 is applied to just the first ($q = 1$) shot gather in Figure 6.25a, then that part of the model allowing for the reconstruction of near offset traces is not illuminated. If, on the other hand, SPDR2 is applied to the first two shot gathers ($q = 1, 2$), then traces from the second $q = 2$ shot gather will begin to illuminate that part of the model required to reconstruct the near offset traces in the first shot gather. As the third through sixth ($q = 3 \dots 6$) shot gathers are added to the input data for SPDR2, the illumination of the pertinent portion of model space is refined. This is illustrated with the schematic in Figure 6.26. In Figure 6.26a, the grey filled boxes are the shot locations, and the black filled boxes are the geophone locations on the observed grid $x_g^{o(q)}$. The near-offset traces that we want to reconstruct fall between the shot location and its nearest observed geophone location. In SPDR2, all shot gathers contribute to the construction of the model according to the adjoint operator in equation 6.36. Hence, in Figure 6.26b, we plot the union of the observed geophone positions from Figure 6.26a. This provides some representation of the lateral locations illuminated in model space, and allows us to suppose that the near offsets for the $q = 1$ shot gather can be recovered via SPDR2. For our example, we illustrate this illumination of model space in Figure 6.27 where from left to right, we plot $\alpha^\dagger(x, z)$ computed using the SPDR2 adjoint and the shot gathers in Figure 6.25a for $q = 1 \dots n_s$ with, respectively, $n_s = 1 \dots 6$. The SPDR2 reconstructed shot gathers are shown in Figure 6.25c. We expect a good reconstruction result for the first (left-most) shot gather, and results that degrade as we increase the shot location (i.e. as the shot gathers in Figure 6.25c progress to the right). In Figure 6.28 we plot traces from the first ($q = 1$) shot gather. In particular, Figure 6.28a plots the traces from original data, and Figure 6.28b plots the reconstructed data, for the 14 nearest offset traces.

We note that with this particular acquisition geometry, and within the context of the schematic in Figure 6.26b, it is always shot gathers that falls to the right of the shot gather of interest that contribute to the correct data reconstruction. In addition, we note that in running SPDR2 on this example, we set $\mathbf{W}_m$ to the identity matrix. This is because the sampling in the geophone dimension is such that no signature of alias is present in model space. Finally, we note that for the data reconstruction of the first ($q = 1$) shot gather, $SNR = 4$dB.

6.10 Summary

We introduce a new data reconstruction algorithm, called shot-profile migration data reconstruction (SPDR), along with its extension (SPDR2) to a two-dimensional data reconstruction algorithm. The algorithms share similarities with other wave-equation based data reconstruction methods. They differ in their choice of mode-space for which we use a shot-profile migration image gather parameterized by lateral position and pseudo-depth. An analysis of SPDR and SPDR2 shows that they are applicable to earth models consisting of reflectors with limited dip. We illustrated the effectiveness of the algorithm using both synthetic and real data examples.

While the single scattering approximation used does not explicitly model multiple reflections, a careful parameterization of model space maps data multiples to false copies of the model space reflectors. In turn, this allows SPDR to interpolate multiple events, as well as primary events. On the other hand, portions of data (such as the head-wave) corresponding to the null-space of the forward operator are not reconstructed by the SPDR algorithm.

In this thesis the SPDR algorithm is limited to a single $2D$ shot gather. We imagine that the extension to a $3D$ shot gather would be straight forward. The same comment can be made regarding the SPDR2 algorithm which, in this thesis, is applied for one geophone and one shot dimension.

While this chapter explored the model space separation of signal and alias that overlap in data space, it did not give definite conditions for when model space signal and alias will be disjoint. Rather, choosing to make broader statements about the need for limited dip reflectors. This presents a possible topic for future research, to find rigorous bounds on model space that would ensure the success of the algorithm.

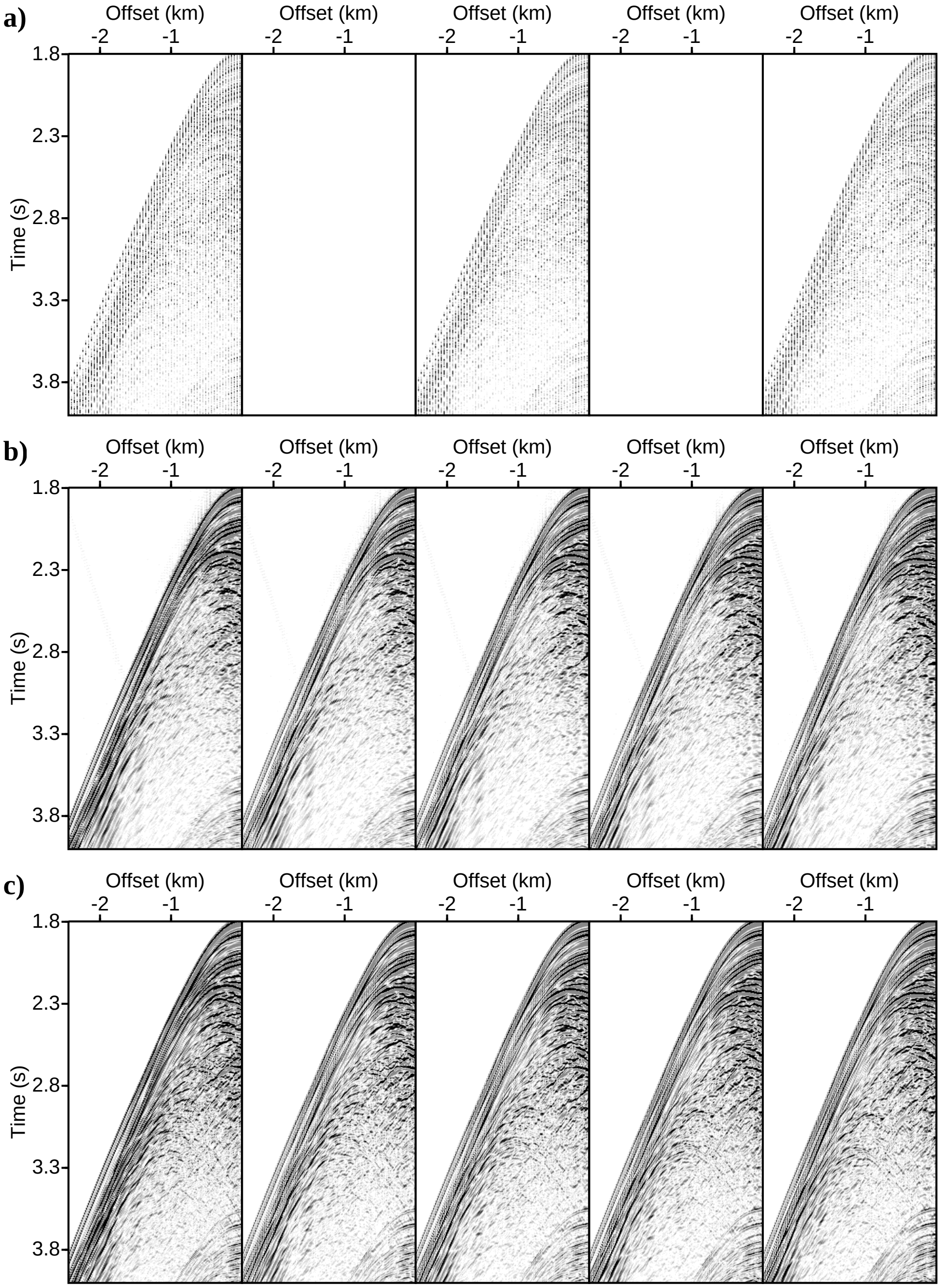

Figure 6.24: 2D real data example: a) the decimated real data, b) the reconstructed data using SPDR2, and the decimated data in a), and c) the original real data, before decimation. For plotting, the data are clipped at ten percent of the maximum absolute value.

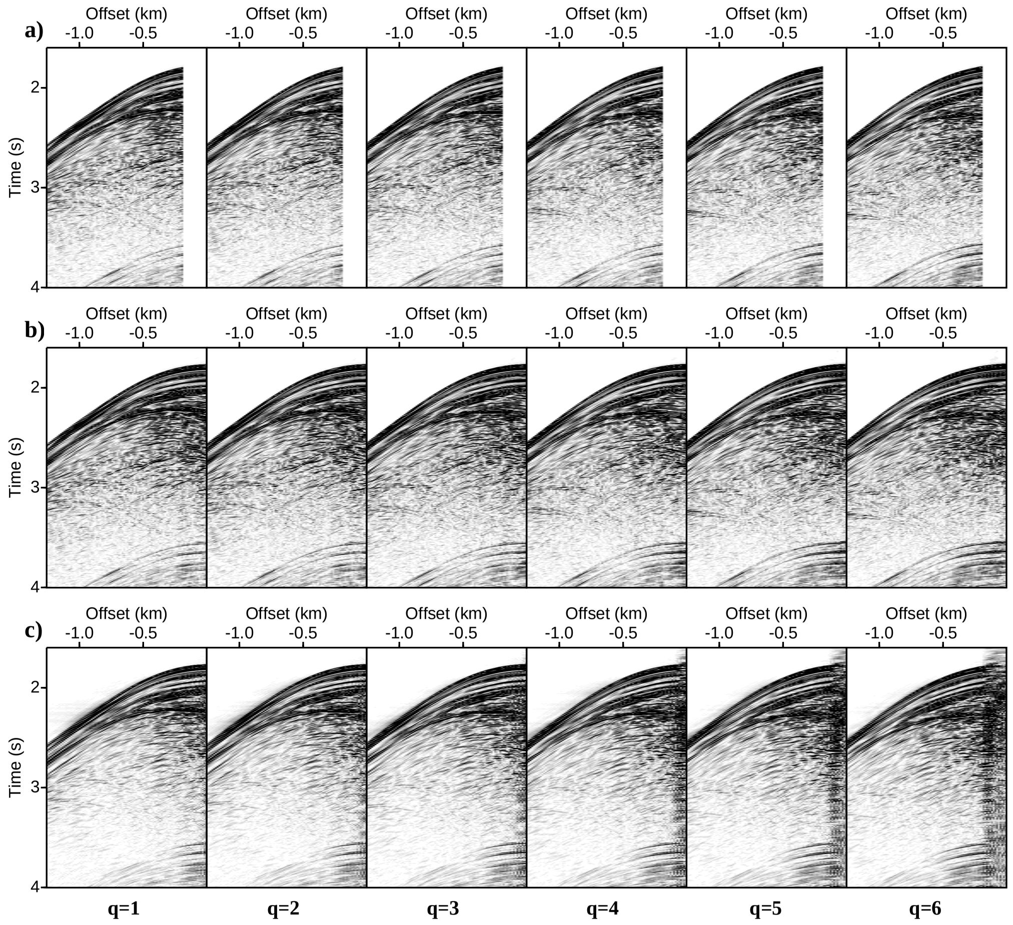

Figure 6.25: 2D real data example, reconstruction of near offsets: a) the decimated data, b) the original data, c) the reconstructed data computed using SPDR2 and the decimated data from a). For plotting, the data are clipped at ten percent of the maximum absolute value.

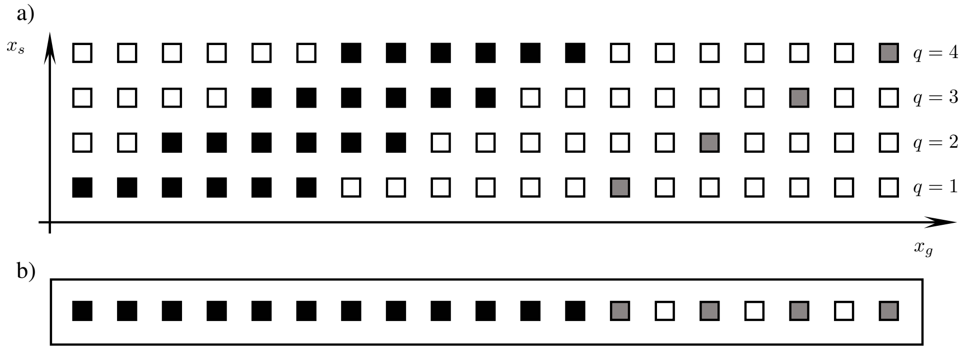

Figure 6.26: 2D real data example, reconstruction of near offsets: a) we give a schematic representation of the data acquisition for the first four shots. b) represents the union of the shot and geophone positions of the four shot gathers, and is representative of the model. Empty and filled boxes represent the nominal geophone grid x_g^n, while black filled boxes represent the observed geophone grid $x_g^{o(q)}$, and the grey filled boxes represent the shot locations relative to the geophone dimension.

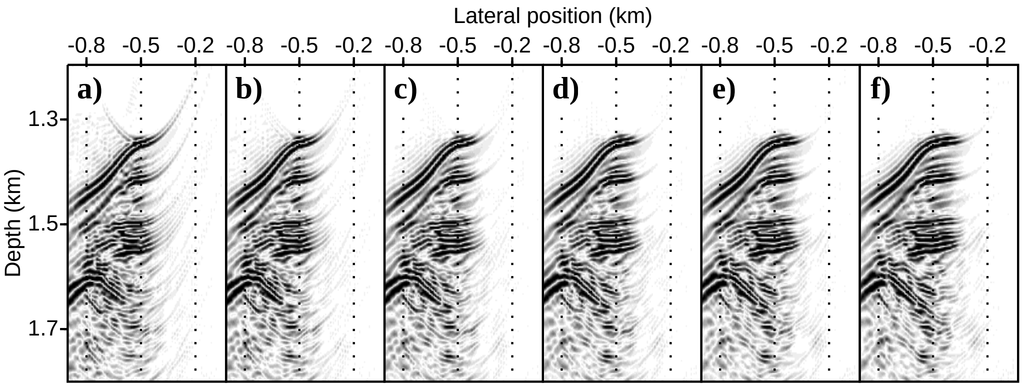

Figure 6.27: 2D real data example, reconstruction of near offsets: we plot the adjoint computed using the decimated shot gathers in Figure 6.24a using $n_s = 1 \ldots 6$ shot gathers, so that in a) we show the result for $n_s = 1$ using the $q = 1$ shot gather, and in b) the result for $n_s = 2$ using the $q = 1, 2$ shot gathers, etc.

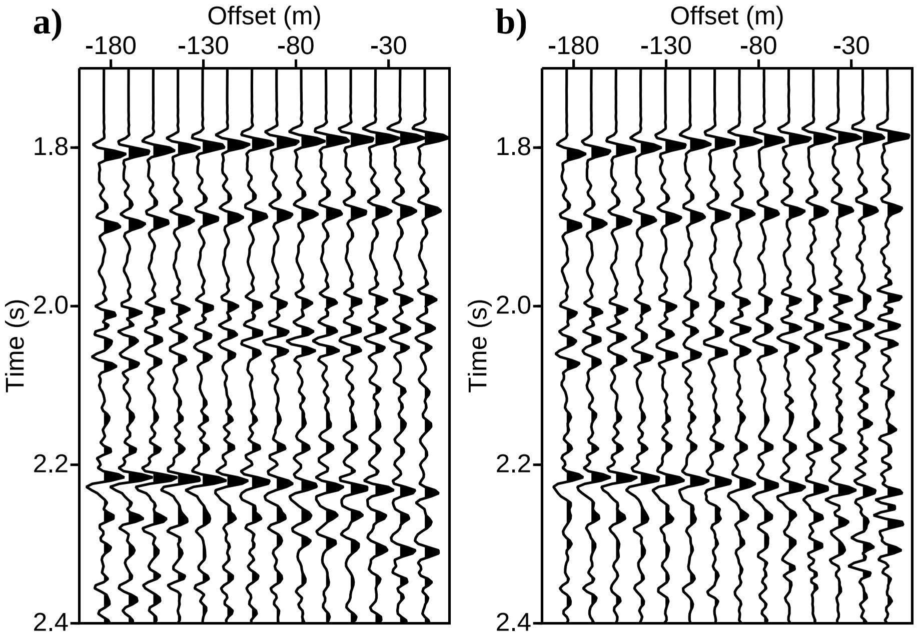

Figure 6.28: 2D real data example, reconstruction of near offsets. For a small time window, we show a) the reconstructed near-offset traces using SPDR2 and the decimated data from Figure 6.24a, and b) the observed near-offset traces.

As with many signal processing algorithms, SPDR and SPDR2 include a parameter selection problem. While there are many methods for parameter selection in inverse theory (e.g. L-curve fitting, Chapter 4), we do not explore them here. We note that for the examples presented, parameter selection was not onerous. We found that the most important parameter to select is, not surprisingly, η. It is used to determine the windowing function used to construct the model weights. In general, we choose it to be a value slightly smaller than the location of the alias in k_{gx}, and which corresponds to k_p for $p = \pm 1$, a property of the observed grid.

■ CHAPTER 7

Parallel implementation of least-squares migration for distributed systems

7.1 Introduction

In Chapters 2–4, we discuss wave-field modelling operators for the source-receiver parameterization of the scattering potential and their application in regularized migration algorithms. In this chapter, we consider the parallel implementation of split-step wave-field modelling and its adjoint, as well as least-squares migration, all for source-receiver migration. The algorithmic complexity (expense) of these source-receiver migration algorithms encourages their parallel implementation. The expense of shot-profile migration (Chapters 5 and 6) also warrants its parallel implementation. However, shot-profile migration is an embarrassingly simple algorithm to make parallel; in particular, because the migration of each shot gather is independent (see Chapter 5). In addition, we note that the least-squares migration implementation is integral to the iterative re-weighted least-squares (IRLS) algorithm used in Chapter 4. Hence, by simple extension, the parallel implementations described in this chapter are also applicable to the IRLS examples shown in Chapter 4. Previous efforts have been made to parallelize source-receiver migration operators (split-step and PSPI) using shared memory computer systems and OpenMP (Kuehl, 2002; Wang, 2005). In this thesis we consider a parallel implementation for distributed computing systems using MPI. The advantage being that distributed systems tend to be larger than shared memory systems in both their number of central processing units (CPU's) and their amount of memory. The disadvantage being in the complexity of the resulting algorithm in needing to cope with a distributed (rather than shared) memory space. Regardless, I thought it would be a useful exercise to write an MPI implementation of least-squares migration with split-step wave-equation operators.

We begin with the equations of interest. Then, we present our parallelization strategy and provide scaling results, showing the behaviour of the implementation for up to, and including, thirty-two MPI processes. Our tests are run on a Linux Opteron cluster with InfiniBand interconnect, and hosted by the Western Canada Research Grid (www.westgrid.ca). The results show an algorithm that scales well, and is an important contribution to this thesis, enabling the examples that were presented in previous chapters.

7.2 Operators

The forward and adjoint operators that we are charged with evaluating were shown in Chapter 2. To review, the split-step forward wave-field modelling operator is,

$$\psi_s = \psi_{s(1)} + \psi_{s(2)} + \cdots + \psi_{s(n_z)}, \tag{7.1}$$

where,

$$\begin{aligned}\psi_{s(l)}(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega) = \left(\frac{1}{2\pi}\right)^{4l} f(\omega)(u_{s(1)}\mathcal{F}^*_{gs}u_{p(1)}\mathcal{F}_{gs})\\ \cdots(u_{s(l-1)}\mathcal{F}^*_{gs}u_{p(l-1)}\mathcal{F}_{gs})(u_{s(l)}\mathcal{F}^*_{gs}u_{p(l)}\mathcal{F}_{gs})\frac{\omega^2}{c^2_{1(l)}}\alpha(\mathbf{x}_g, \mathbf{x}_s, z_l),\end{aligned} \tag{7.2}$$

$$u_{p(l)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}, \omega) = -\frac{e^{i(k_{gz(l)}+k_{sz(l)})(z_l - z_{l-1})}}{16k_{gz(l)}k_{sz(l)}}, \tag{7.3}$$

and,

$$u_{s(l)}(\mathbf{x}_g, \mathbf{x}_s, \omega) = e^{i\omega(c^{-1}_{0(l)}(\mathbf{x}_g)+c^{-1}_{0(l)}(\mathbf{x}_s)-2c^{-1}_{1(l)})(z_l - z_{l-1})}. \tag{7.4}$$

Likewise, the adjoint operator is,

$$\alpha^\dagger(\mathbf{x}_g, \mathbf{x}_s; z_l) = \sum_j u^*(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_l)\psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega_j), \tag{7.5}$$

where u^* is computed recursively in depth from,

$$\begin{aligned}&u^*(\mathbf{x}_g, \mathbf{x}_s; \omega_j; z_l)\\ &= \left(\frac{1}{2\pi}\right)^{4l} f^*(\omega)(\frac{\omega^2}{c^2_{1(l)}}u^*_{s(l)}\mathcal{F}^*_{gs}u^*_{p(l)}\mathcal{F}_{gs})\cdots(u^*_{s(2)}\mathcal{F}^*_{gs}u^*_{p(2)}\mathcal{F}_{gs})(u^*_{s(1)}\mathcal{F}^*_{gs}u^*_{p(1)}\mathcal{F}_{gs}),\end{aligned} \tag{7.6}$$

and the recursion starts at $z_k = z_0$ where z_0 is the measurement surface. The functions, $u^*_{p(l)}$ and $u^*_{s(l)}$ are, respectively, the complex conjugate of $u_{p(l)}$ and $u_{s(l)}$.

The vertical wave-numbers (k_{gz} and k_{sz}) play an important role in load balancing a parallel implementation of the forward and adjoint split-step operators. In particular, we recall the dispersion relations,

$$k_{gz(l)} = sgn(\omega)\sqrt{\omega^2/c_{1(l)}^2 - \mathbf{k}_{gx}\cdot\mathbf{k}_{gx}} \qquad k_{sz(l)} = sgn(\omega)\sqrt{\omega^2/c_{1(l)}^2 - \mathbf{k}_{sx}\cdot\mathbf{k}_{sx}}. \tag{7.7}$$

The evanescent wave-field is where $k_{gz(l)}$ or $k_{sz(l)}$ are imaginary, and we assume that the evanescent contribution to the wave-field is small, and can be discarded; thus, shrinking the domain of computation for u_p to where, at each depth/frequency pair,

$$-\omega/c_{1(l)} \leq |\mathbf{k}_{gx}| \leq \omega/c_{1(l)}. \tag{7.8}$$

In the next section, the algorithm is described such that computation is made parallel by assigning groups of frequencies ω to each MPI process. Equation 7.8 means that care must be taken when distributing the frequencies across the MPI processes in order to maintain a load-balanced algorithm.

7.3 Algorithms

Section 7.2 reviewed the equations for the forward and adjoint operators used in least-squares migration, and using the source-receiver parameterization of the scattering potential. In our parallel implementation of these equations each MPI process evaluates the operators for some subset of frequencies. Thus, we divide the processing work amongst MPI processes. In addition, the storage of $\alpha^\dagger$ and ψ_s is shared amongst the distinct memory spaces of the MPI processes. In particular, $\alpha^\dagger$ is partitioned along its depth axis, and ψ_s is partitioned along its frequency axis. This allows us to take advantage of the full storage capacity of a distributed system. Finally, the least-squares system of equations are solved by the method of conjugate gradients (Chapter 3) which is, somewhat trivially, also given a corresponding parallel implementation.

7.3.1 Adjoint operator

In short, the algorithm for computing the adjoint operator in equation 7.5 consists of the following steps, most of which we will discuss in greater detail. 1) We read data (the recorded wave-field ψ_s) from disk, so that each MPI process stores some subset of common midpoint gathers (or common shot gathers). 2) We Fourier transform the data from time t to frequency ω. 3) We re-distribute the data in memory so that the memory space for each MPI process stores a subset of frequencies, but for all space $(\mathbf{x}_g, \mathbf{x}_s)$. 4) For each frequency, we evaluate the adjoint operator in equation 7.5. 5) We undo the re-distribution in step 3

$r_1 \cap r_2 \cap r_3$	1	2	3	4	5	6	7	8	9	10	11	12
r_1	1			4			7			10		
r_2		2			5			8			11	
r_3			3			6			9			12

Table 7.1: We illustrate the distribution of 12 frequencies across 3 MPI processes using the round-robin strategy in equation 7.9.

so that each MPI process stores some subset of common midpoint (or common shot) image gathers which can then be written to disk.

We make some comments about the algorithm when there are n_r MPI processes, n_m common midpoint gathers, n_w realizations of frequency, and n_z realizations of depth. In particular, we emphasize how each step makes the processing and storage tasks parallel. In step 1, the data are stored in parallel, so that each MPI process stores n_m/n_r common midpoint gathers. In step 2, the Fourier transforms from $\psi(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; t)$ to $\psi(\mathbf{x}_g, z_s|\mathbf{x}_s, z_s; \omega)$ in each MPI process are mutually independent, making the implementation parallel.

Step 3 requires communication between the memory spaces of all MPI processes. Before this communication, each MPI process stores all frequencies for some subset of common midpoint gathers; and, after the communication each MPI process stores all common midpoint gathers for some subset of frequencies, with each memory space holding n_w/n_r frequencies. The memory requirements before and after communication are the same. This redistribution of frequencies must take into account the evanescent portion of the wave-field to maintain a load-balanced algorithm. In particular, we re-distribute data using a round-robin strategy so that the j^{th} frequency in the data is the i^{th} frequency to be stored in the memory space of the k^{th} MPI process, so that,

$$j = k + (i-1)n_r, \tag{7.9}$$

$k = 1, 2 \ldots n_r$, and $i = 1, 2 \ldots n_w/n_r$. To illustrate, we consider a simple example with $n_w = 12$, and $n_r = 3$, and show the resulting distribution of frequencies in Table 7.1, where r_k denotes the k^{th} MPI process.

Step 4 is the evaluation of the adjoint operator in equation 7.5 and, in the context of a parallel MPI implementation, has two complications. First the sum over frequency is divided into parts. In particular,

$$\alpha^\dagger(\mathbf{x}_g, \mathbf{x}_s; z_l) = \sum_{k=1}^{n_r} \left[\sum_{j \in \mathcal{W}_k} u^*(\mathbf{x}_g, \mathbf{x}_s; \omega_j, z_l) \psi_s(\mathbf{x}_g, z_0|\mathbf{x}_s, z_o; \omega_j) \right], \tag{7.10}$$

where $\mathcal{W}_k$ is the set of frequency indices stored in the memory space of the k^{th} MPI process. The inner sum in equation 7.10 is performed independent of other MPI processes; whereas, the outer sum requires communication between all MPI processes. Second, the storage of $\alpha^\dagger$ is distributed amongst the MPI processes. In particular, we let the k^{th} MPI process store the set of depth indices $\mathcal{Z}_k$ such that,

$$\mathcal{Z}_k = \{(k-1)n_z/n_r \leq l < kn_z/n_r\}\,. \tag{7.11}$$

Remembering the iteration used in evaluating equation 7.5, we evaluate equation 7.10 for $\mathcal{Z}_k$ on all MPI processes, and store its result on the k^{th} MPI process. In Algorithm 7.1, we illustrate step 4.

Algorithm 7.1 Step 4 in computing the adjoint operator

```
Running on the $i^{th}$ MPI process.
for $k = 1$ to $n_r$ do
    for all $l \in \mathcal{Z}_k$ and $j \in \mathcal{W}_i$ do
        Evaluate the inner sum in equation 7.10.
    end for
    for all $l \in \mathcal{Z}_k$ do
        Evaluate the outer sum in equation 7.10 by communicating across MPI processes,
        finding,
            $\alpha^\dagger(\mathbf{x}_g, \mathbf{x}_s; z_l) \quad , \quad \forall l \in \mathcal{Z}_k$
    end for
    Store $\alpha^\dagger(\mathbf{x}_g, \mathbf{x}_s; z_l)$ for all $l \in \mathcal{Z}_k$ on the $k^{th}$ MPI process.
end for
```

7.3.2 Forward operator

The parallel organization of the forward operator is similar to that of the adjoint operator. Here, we assume that $\alpha(\mathbf{x}_g, \mathbf{x}_s; z_l)$ is already stored in the memory spaces of the MPI processes where, as before, the memory space of the k^{th} MPI process stores α for depths z_l where $l \in \mathcal{Z}_k$. However, an analysis of equations 7.1 and 7.2 show that all depths are required to compute ψ_s for a single frequency. Therefore, a strategy for computing the forward operator must include communication of α between the memory spaces of the MPI processes. Fortunately, a recursion is evident in equation 7.2 makes this communication feasible. We should emphasize that this recursion is not new to this chapter, but is the accepted method of computing the forward (de-migration) operator (e.g. Kühl and Sacchi, 2003), regardless of whether the implementation is parallel or serial, and which we reviewed in Chapter 2.

We recall that the recursion for computing the forward operator was show in equation 2.23.

For convenience, we re-write equation 2.23 here, so that for $l = n_z \dots 2$,

$$\begin{aligned}\psi_s(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega_j) &= \Delta z \left(\frac{1}{2\pi}\right)^{4l} f(\omega_j) v_1 \\ v_{l-1} &= u_{s(l)} \mathcal{F}^*_{gs} u_{p(l)} \mathcal{F}_{gs} \left(v_l + \frac{\omega^2}{c^2_{1(l)}} \alpha(\mathbf{x}_g, \mathbf{x}_s, z_l) \right),\end{aligned} \tag{7.12}$$

and where $v_{n_z} = 0$. Due to the recursion in equation 7.12, we use the following steps in our parallel implementation: 1) we communicate α from the memory space of the $k = n_r$ MPI process to all memory spaces. 2) For all $l \in \mathcal{Z}_{n_r}$, we compute the recursion in equation 7.12 for all frequencies ω_j and lateral space $(\mathbf{x}_g, \mathbf{x}_s)$. In doing so, we remember that the memory space of the i^{th} MPI process stores frequencies in the set $\mathcal{W}_i$. 3) We repeat steps 1 and 2 for α in the memory space of the $k = n_r - p$ MPI process for $p = 1 \dots (n_r - 1)$. These steps are illustrated in Algorithm 7.2.

Algorithm 7.2 Computation of the forward operator

```
Running on the $i^{th}$ MPI process.
for $k = n_r$ to 1 do
    Communicate $\alpha(\mathbf{x}_g, \mathbf{x}_s; z_l)$ for $l \in \mathcal{Z}_k$ to all MPI processes.
    for all $l \in \mathcal{Z}_k$ and $j \in \mathcal{W}_i$ do
        Evaluate the recursion in equation 7.12.
    end for
end for
```

7.3.3 Parallelization of the conjugate gradient algorithm for the normal equations

We review the use of the forward (Section 7.3.2) and adjoint operators (Section 7.3.1) in the least-squares normal equations,

$$(\mathbf{A}^H \mathbf{A} + \mu \mathbf{I})\mathbf{m} = \mathbf{A}^H \mathbf{d}, \tag{7.13}$$

where $\mathbf{A}$ is the forward operator corresponding to equations 7.1 and 7.2, and $\mathbf{A}^H$ is the adjoint operator corresponding to equation 7.5. We call the vector $\mathbf{m}$ the model, made from realizations of $\alpha(\mathbf{x}_s, \mathbf{x}_g, z)$, and the vector $\mathbf{d}$ the data, made from realizations of $\psi_s(\mathbf{x}_g, \mathbf{x}_s, \omega)$. We solve equation 7.13 for $\mathbf{m}$ using a parallel implementation of conjugate gradients. In doing so, we assume that the forward $\mathbf{A}$ and adjoint $\mathbf{A}^H$ are implemented according to Sections 7.3.2 and 7.3.1. We point out that this does not constitute a novel method (see for example Gupta et al. (1995)). This is not surprising giving how readily the conjugate gradient method is made parallel. Regardless, we give an overview of our implementation.

Please note that we assume, in this section, that the reader is knowledgeable of the conjugate gradient method (see Chapter 3).

Algorithm 7.3 Conjugate gradient algorithm for distributed parallel systems

```
Running on the $l^{th}$ MPI process, and assuming that $\mathbf{m}$ is of size $n_m$.
$\mathbf{r}_{0,l} = \mathbf{A}_l^H(\mathbf{d}_l - \mathbf{A}\mathbf{m}_{0,l}) - \mu\mathbf{m}_{0,l}$
$\mathbf{p}_{0,l} = \mathbf{r}_{0,l}$
$\delta_{0,l} = \mathbf{r}_{0,l}^T\mathbf{r}_{0,l}$
(MPI) $\delta_0 = \sum_l \delta_{0,l}$
for $i = 0$ to $i = n_m$ and $\delta_i > \epsilon\delta_0$ do
    $a_l = \mathbf{p}_{i,l}^T(\mathbf{A}_l^H\mathbf{A}_l + \mu\mathbf{I})\mathbf{b}_{i,l}$
    (MPI) $a = \sum_l a_l$
    $\alpha_i = \delta_i/a$
    $\mathbf{m}_{i+1,l} = \mathbf{m}_{i,l} + \alpha_i\mathbf{p}_{i,l}$
    $\mathbf{r}_{i+1,l} = \mathbf{A}_l^H(\mathbf{d}_l - \mathbf{A}\mathbf{m}_{i,l}) - \mu\mathbf{m}_{i,l}$
    $\delta_{i+1,l} = \mathbf{r}_{i+1,l}^T\mathbf{r}_{i+1,l}$
    (MPI) $\delta_{i+1} = \sum_l \delta_{i+1,l}$
    $\beta_i = \delta_{i+1}/\delta_i$
    $\mathbf{p}_{i+1,l} = \mathbf{r}_{i+1,l} + \beta_i\mathbf{p}_{i,l}$
end for
```

To proceed, recall that conjugate gradients is an iterative method, and let $\mathbf{m}_i$ be the result of the i^{th} iteration. Then, we partition the model $\mathbf{m}_i$ and data $\mathbf{d}$ according to the number of MPI processes so that,

$$\mathbf{d}^T = \begin{bmatrix} \mathbf{d}_1^T & \mathbf{d}_2^T \cdots \mathbf{d}_{n_r}^T \end{bmatrix} \tag{7.14}$$

$$\mathbf{m}_i^T = \begin{bmatrix} \mathbf{m}_{i,1}^T & \mathbf{m}_{i,2}^T \cdots \mathbf{m}_{i,n_r}^T \end{bmatrix}, \tag{7.15}$$

where $\mathbf{d}_l$, $l = 1 \ldots n_r$ contain the frequency indices in the set $\mathcal{W}_l$. Likewise, $\mathbf{m}_{i,l}$, $l = 1 \ldots n_r$ contain the depth indices in the set $\mathcal{Z}_l$. Then, the forward and adjoint operators applied on the l^{th} MPI process are given by $\mathbf{A}_l$ and $\mathbf{A}_l^H$, respectively. With this in mind, we define a residual vector $\mathbf{r}_{i,l}$ on the l^{th} MPI process and at the i^{th} conjugate gradient iteration such that,

$$\mathbf{r}_{i,l} = \mathbf{A}_l^H(\mathbf{d}_l - \mathbf{A}_l\mathbf{m}_{i,l}) - \mu\mathbf{m}_{i,l}. \tag{7.16}$$

Likewise, we let the conjugate directions be $\mathbf{p}_{i,l}$ for, again, the i^{th} conjugate gradient iteration and l^{th} MPI process. With these definitions in mind, we present our implementation of the parallel conjugate gradient method in Algorithm 7.3 where lines that are prefixed with (MPI) require communication across all n_r MPI processes. Please note that Algorithm 7.3 is not indicative of the number of times that the forward and adjoint operator must be evaluated for each iteration of the conjugate gradient algorithm. It is well known that only one forward and one adjoint operation need be evaluated for each conjugate gradient iteration.

MPI processes	Forward operator		Adjoint operator		Least-squares inverse	
	Run-time	Speed-up	Run-time	Speed-up	Run-time	Speed-up
1	176.5	1.0	174.5	1.0	1402.7	1.0
2	89.4	2.0	85.6	2.0	695.7	2.0
4	44.7	4.0	43.3	4.0	345.3	3.9
8	22.4	7.9	22.5	7.8	177.5	7.9
16	11.6	15.2	11.9	14.7	89.9	15.6
32	6.4	27.4	6.9	25.3	49.5	28.4

Table 7.2: Run-time and speed-up values for evaluating the forward, adjoint and least-squares inverse (using ten iterations of the conjugate gradient method). All run-times are given in minutes.

We leave it to the interested reader to understand these redundancies, and which were also discussed in Chapter 3.

7.4 Results

We present scaling results of the proposed algorithm, showing an almost ideal speed-up as a function of the number of MPI processes. All tests were performed using the Marmousi data (Versteeg, 1994, e.g). The results are presented in Table 7.2. The code was run on a Linux Opteron cluster with InfiniBand interconnect hosted by the Western Canada Research Grid (www.westgrid.ca). If the algorithm scaled perfectly as a function of the number of MPI processes, then we would expect the number of MPI processes to match the Speed-up.

To give some confidence that the code is working, we compare it to the serial code in Figures 7.1-7.3. In particular, we show results from applying the adjoint operator (Figure 7.1), the least-squares inverse (Figure 7.2) and the forward operator (Figure 7.3). In each figure, we plot the result from the serial program, and the results from the parallel code using 2, 4, 8, 16 and 32 MPI processes. We note that the purpose of the least-squares result is to illustrate the consistency of the algorithm, rather than for comparison with the adjoint result in Figure 7.1. The least-squares result is computed using ten conjugate gradient iterations.

7.5 Summary

We implement split-step wave-field modelling and migration operators for parallel distributed computing systems. In addition, we present a parallel implementation of the conjugate gradient algorithm. The primary purpose of this work was to accelerate the

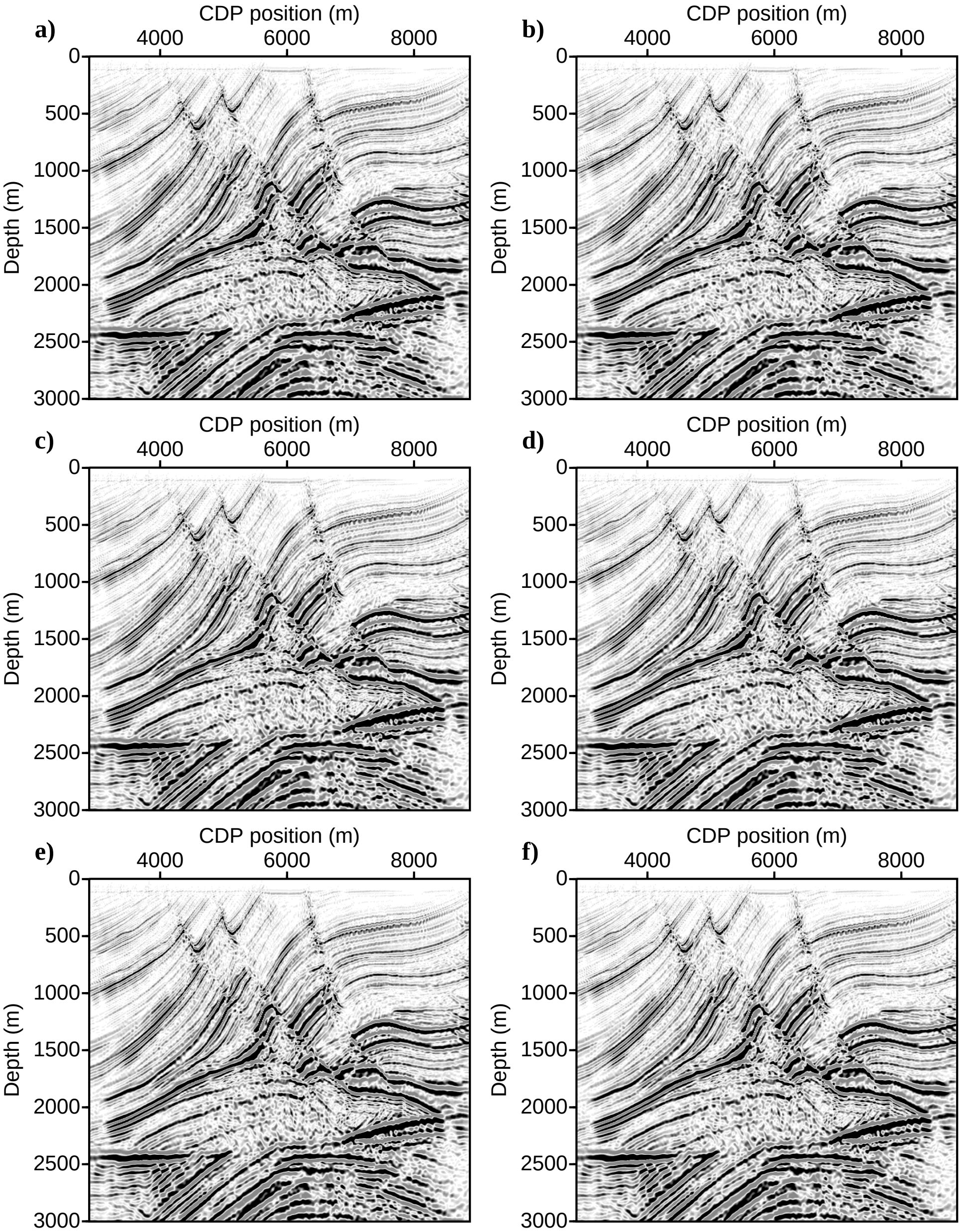

Figure 7.1: Parallel and serial codes comparison: the adjoint operator, applied to the finite difference data using a) the serial code, and b)-f) using the MPI code with, respectively, 2, 4, 8, 16 and 32 MPI processes.

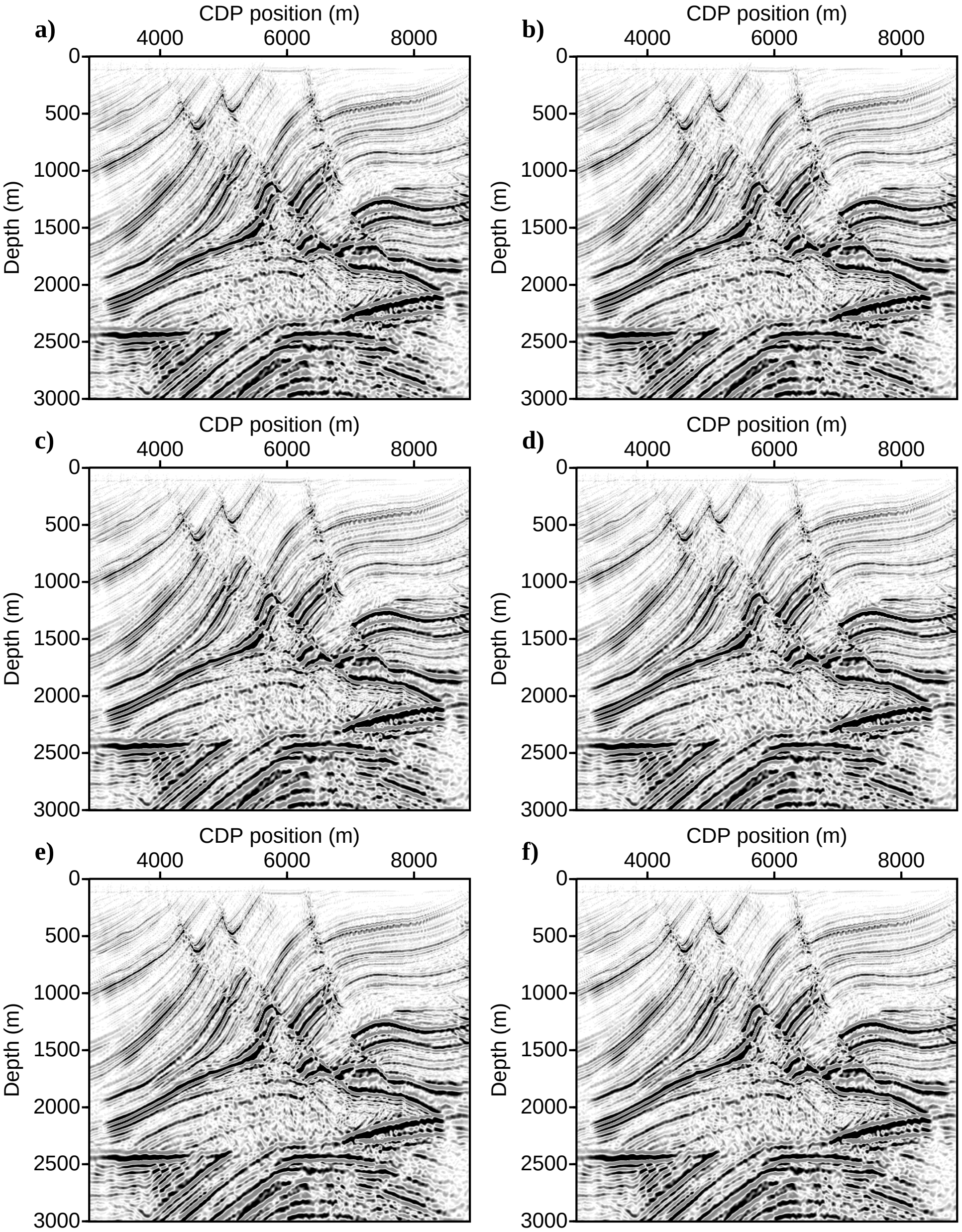

Figure 7.2: Parallel and serial code comparison: the least-squares inverse, applied to the finite difference data using a) the serial code, and b)-f) using the MPI code with, respectively, 2, 4, 8, 16 and 32 MPI processes.

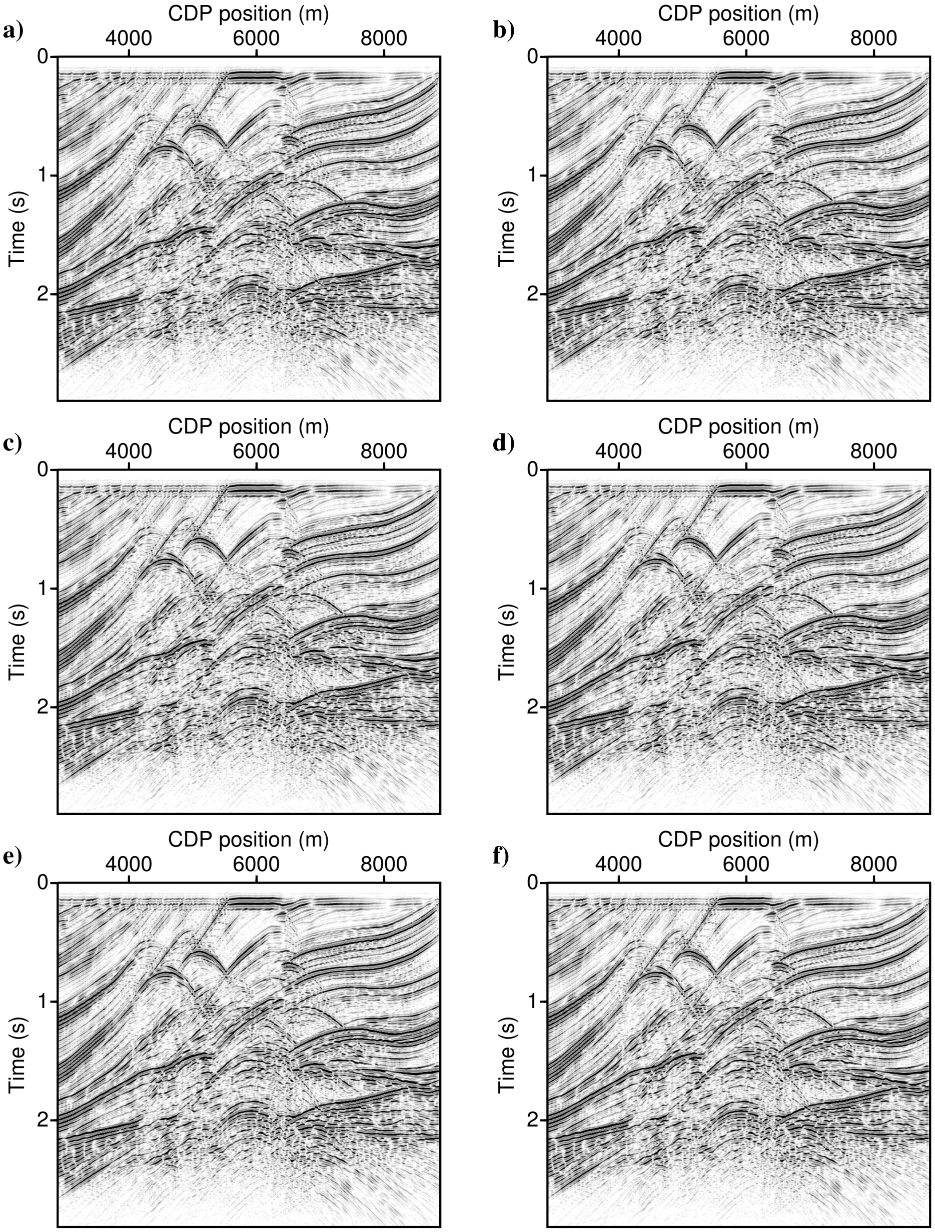

Figure 7.3: Parallel and serial codes comparison: the forward operator, applied to the least-squares inverse (show in Figure 7.2) using a) the serial code, and b)-f) using the MPI code with, respectively, 2, 4, 8, 16 and 32 processes.

pace of research, reducing the turn-around time when running regularized migration tests used throughout this thesis.

■ CHAPTER 8

Data pre-processing: noise attenuation

8.1 Introduction

In this thesis, we have primarily been concerned with wave-field modelling and migration operators for least-squares migration. It is important to remember that migration is one step in a larger seismic processing work-flow. Data collected in the field may undergo any number of processing steps before migration is applied. In some cases these processing steps are necessary because of assumptions made in the construction of the migration operator. For example, the migration operator does not account for either the direct wave-field or *seismic multiples*. In seismic exploration, multiples (previously discussed in Chapter 6) are data events from that portion of the wave-field that has experienced at-least one downward reflection during its propagation from source to receiver (e.g. Verschuur et al., 1992; Weglein et al., 1997).[1] To model these events, we would require an operation that is nonlinear in the scattering potential. We recall that migration is linear in the scattering potential. An additional, and important processing step is the removal of random noise from seismic data. Random noise is another feature of data that can hinder migration, although as we have seeing in previous chapters, least-squares migration can cope with significant levels of random noise.

In this chapter, we use sparse coding (an established method in information theory) to attenuate both random Gaussian and coherent (multiples) noise in seismic data. Sparse coding finds some linear expansion of data in which both the coefficients and basis of the expansion are found from the data. That is, the basis of the expansion is data-driven. The coefficients of the expansion are the *sparse code*, and we attenuate noise in the data by filtering the sparse code.

The sparse coding representation of data is found by making some assumption about its coefficients, namely that they are sparse. This is similar to principal component analysis

[1] to the exclusion of source- and receiver-side ghosts which are defined separately.

(PCA). In PCA, a series expansion of data is found such that its coefficients are small (in a $L2$ sense), and the basis of the expansion follows from this criterion. PCA has been applied to the problem of noise attenuation in seismic data (Freire and Ulrych, 1988; Kaplan and Ulrych, 2002). More recently, empirical mode decomposition (EMD) has been used to represent seismic data (Bekara and van der Baan, 2009). Like sparse coding and PCA, EMD derives its basis functions from data; but does not specify a criteria for the coefficients of the expansion. Instead, it finds the basis by applying successive operations to the data (Huang et al., 1998). In contrast to all of these method, the curvelet transform (Candès and Donoho, 2004) and local wave-field decomposition (Sacchi et al., 2004) are among those data representations where the basis are prescribed analytically, and the coefficients are found by projection. Interestingly, the set of basis functions studied in the local wave-field decomposition and curvelet transforms often resemble those found by sparse coding.

Sparse coding stems from the field of cognitive neuroscience and is an effort to explain the function of the primary visual cortex (Vinje and Gallant, 2000). In-fact, the underlying mathematical model for sparse coding can be described using a neural network (Olshausen and Field, 1997). It is, then, not surprising that it has been used for feature extraction and de-noising of images (e.g. Oja et al., 1999; Hoyer, 1999), something that the average human is quite proficient at. In the development of its practical applications to image feature extraction and de-noising, sparse coding has gained algorithms that were originally developed for independent component analysis (ICA) (Common, 1994). For our work, we employ the widely used *FastICA* algorithm (Hyvärinen, 1999).

Here, we use sparse coding for two specific tasks in seismic data processing: 1) random Gaussian noise attenuation, and 2) multiple attenuation. We accomplish both tasks by applying a filter to the sparse code. In random noise attenuation, we follow the work of, for example, Hyvärinen et al. (1999), and shrink the sparse code (Kaplan and Ulrych, 2005). For multiple attenuation, we apply sparse coding to a common midpoint gather after normal move-out correction, filtering the sparse code based on the dominant ray-parameter in the corresponding basis vectors. This allows us to filter out the primary events from the gather, leaving only multiples which are, subsequently, subtracted from the original data. This is similar to Hampson (1986) and Foster and Mosher (1992), who both use the Radon transform in a similar manner.

We start with a brief review of sparse coding, referring the interested reader to relevant articles. Then, we describe the filters applied to the sparse code. For random de-noising, we give a brief review of the traditional sparse code shrinkage method. For multiple attenuation, we describe a novel filter that effectively sorts the basis by its dominant ray-parameters, and then truncates the series expansion. The sorting separates the data into flat (signal) and curved (noise) components. Next, we show a synthetic data example for random noise attenuation, making comparisons to results obtained by $f-x$ deconvolution (Canales, 1984; Gülünay, 1986; Galbraith, 1991) and the curvelet transform (Starck et al., 2002; Herrmann and Hennenfent, 2008). Lastly, we show a real data example from the Gulf of Mexico for multiple attenuation, making a comparison to a method using the parabolic Radon transform.

We do not presume to recommend sparse coding in-place of established methods such as $f-x$ deconvolution for Gaussian noise attenuation, and the Radon transform for multiple attenuation. However, we think that the application of data-driven data representation to geophysical signal processing is interesting.

8.2 Sparse coding

Consider the expansion,

$$\mathbf{x} = \sum_{i=1}^{n} y_i \mathbf{p}_i = \mathbf{P}\mathbf{y}, \tag{8.1}$$

where $\mathbf{x} \in \mathcal{R}^m$ and $\mathbf{y}^T = \left[\begin{array}{cccc} y_1 & y_2 & \cdots & y_n \end{array} \right]$ are random vectors (y_i are random variables), and $\mathbf{p}_i \in \mathcal{R}^m$ is the i^{th} column of $\mathbf{P} \in \mathcal{R}^{m\times n}$. Here, $\mathbf{p}_i$ are the basis vectors of the expansion, and y_i are the corresponding coefficients. In other words, the vectors $\mathbf{p}_i$ span a subspace containing $\mathbf{x}$. In this chapter, both $\mathbf{p}_i$ and $\mathbf{y}$ are computed using ICA, providing a sparse representation of $\mathbf{x}$. The resulting expansion is called sparse coding, and $\mathbf{y}$ is the sparse code.

8.2.1 Input data

We consider data $d(t,x)$ in time t and space x, and work to construct, from it, equation 8.1. To construct the realizations of $\mathbf{x}$, we define matrices $\mathbf{W}_j$, $j = 1 \ldots J$ and $\mathbf{V}_i$, $i = 1 \ldots n$, and call $\mathbf{W}_j$ *data patches*, and $\mathbf{V}_i$ *basis patches*. We apply a windowing algorithm to $d(t,x)$ where the j^{th} window is the data patch $\mathbf{W}_j \in \mathcal{R}^{M\times N}$ with M time samples and N samples (i.e. seismic traces) in space. Presently, we let,

$$\mathbf{W}_j = \sum_{i=1}^{n} y_{ij} \mathbf{V}_i, \tag{8.2}$$

be the expansion of the j^{th} data patch $\mathbf{W}_j$ onto the basis patches $\mathbf{V}_i$, and coax equation 8.2 into the form of equation 8.1 by letting each data patch $\mathbf{W}_j$ be a realization of $\mathbf{x}$. In particular,

- The j^{th} data patch $\mathbf{W}_j$ is mapped, via lexicographic reordering, to the j^{th} realization of $\mathbf{x}$.
- The basis patches $\mathbf{V}_i$ are mapped, via lexicographic reordering, to the basis vectors $\mathbf{p}_i$.
- The coefficient y_{ij} is the j^{th} realization of y_i, the i^{th} random variable in the sparse code $\mathbf{y}$.

For example, consider data patches such that $\mathbf{W}_j \in \mathcal{R}^{32\times 16}$ (i.e. 32 time samples by 16 traces). In this case, both $\mathbf{x}$ and $\mathbf{y}$ would have length $m = 512$. If several such windows are collected from $d(t, x)$, then both $\mathbf{x}$ and $\mathbf{y}$ will have several realizations, one for each data patch.

8.2.2 The sparse code and entropy

The sparse coding representation of $\mathbf{x}$ is achieved using ICA, a method commonly used for source separation (e.g. Common, 1994). ICA finds the components of the expansion in equation 8.1 through the computation of independent components,

$$y_i = \mathbf{b}_i^T \mathbf{x} \qquad\qquad \mathbf{y} = \mathbf{B}\mathbf{x}, \tag{8.3}$$

where $\mathbf{b}_i^T$ is the i^{th} row of $\mathbf{B}$, and is chosen such that the independent components y_i have zero-mean, unit-variance, minimum entropy, and are statistically uncorrelated. Here, we use differential entropy $h(p_Y)$ which for the i^{th} independent component is,

$$h(p_Y) = -\int_{-\infty}^{\infty} p_Y(y_i) \ln p_Y(y_i) dy_i, \tag{8.4}$$

where $y_i \sim p_Y$, and in the context of ICA, y_i is an independent component when equation 8.4 has its minimum. The random variable y_i is distributed as the probability density function p_Y. It follows that distributions can be found which maximize entropy. In particular, given moment constraints defined by the functions r_k and,

$$\int_{-\infty}^{\infty} r_k(y_i) p_Y(y_i) dy_i = c_k \quad , \quad k = 1 \ldots l, \tag{8.5}$$

Appendix F.1 finds the classical result that the maximum entropy distribution of y_i is,

$$p_Y(y_i) = \exp\left(-1 + \lambda_0 + \sum_{k=1}^{l} \lambda_k r_k(y_i)\right), \tag{8.6}$$

and shows that for a prescribed mean and variance it is Gaussian. In equation 8.6, λ_k are Lagrange multipliers. Since the maximum entropy distribution is Gaussian, minimizing entropy maximizes its distance from a Gaussian distributed random variable. One way to manifest this distance is to make p_Y a sparse distribution, in turn, making y_i, in addition to an independent component, one component of the sparse code.

As is evident from equation 8.4, computing entropy creates the rather difficult task of estimating integrals from probability density functions. Here, we use an estimate introduced by Hyvärinen (1998) that expands the probability density function p_Y onto a basis of non-polynomial functions ($r_k(y)$ in equation 8.5) such that it satisfies the maximum entropy distribution in equation 8.6 and the moment constraints in equation 8.5. This estimate leads to an approximation for a measure related to entropy, called negentropy $J(p_Y(y_i))$, such that,

$$J(p_Y(y_i)) = h(p_\xi(\xi)) - h(p_Y(y_i)) \approx \frac{1}{2}\sum_{k=3}^{l}[\mathrm{E}(r_k(y_i))]^2, \tag{8.7}$$

where E is the expectation operator, $p_\xi(\xi)$ is a Gaussian distribution with the same mean and variance as $p_Y(y_i)$. Hence, negentropy measures the distance from a Gaussian random variable, and minimizing the entropy of y_i is equivalent to maximizing its negentropy. Therefore, independent components correspond to maxima of equation 8.7. Note that in the derivation of equation 8.7, $k = 1, 2$ are used for the prescribed constraints of zero mean and unit variance.

8.2.3 The FastICA algorithm

Due to the relation between negentropy and independent components, the ICA problem is reduced to one in optimization with an associated cost function measuring negentropy. We use the FastICA algorithm with symmetric orthogonalization (Hyvärinen, 1999) for the optimization, the details of which are given in Appendix F.2. Here, we show the utility of PCA for ICA, write down the cost function that the algorithm optimizes, and show how its solution relates to the sparse coding basis $\mathbf{P}$.

PCA, used as a pre-processor, allows for the derivation of much needed constraints for the optimization problem. Given zero mean data $\mathbf{x}$, we define a vector $\mathbf{z}^T = \left[\begin{array}{cccc} z_1 & z_2 & \cdots & z_n \end{array}\right]$, and a whitening matrix $\mathbf{\Gamma} \in \mathcal{R}^{n\times m}$ so that $\mathbf{z} = \mathbf{\Gamma}\mathbf{x}$ where $\mathbf{z}$ is white ($\mathrm{E}(\mathbf{z}) = \mathbf{0}$, $\mathrm{E}(\mathbf{z}\mathbf{z}^T) = \mathbf{I}$ and $\mathbf{I}$ is the identity matrix). That is, the random variables z_i, $i = 1 \ldots n$ are mutually uncorrelated. An appropriate choice for $\mathbf{\Gamma}$ is found such that,

$$\mathbf{\Gamma} = \mathbf{\Sigma}^{-1}\mathbf{U}^T \quad , \quad \mathbf{\Sigma}^{-1} = \left[\begin{array}{cc} (\mathbf{\Sigma}')^{-1} & \mathbf{0} \end{array}\right],$$

where the columns of $\mathbf{U}$ are the principal components of $\mathbf{x}$, $\mathbf{\Sigma}' = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$, and

σ_i is the standard deviation of the i^{th} principal component (e.g. Kaplan and Ulrych, 2005). Due to $\Sigma^{-1} \in \mathcal{R}^{n\times m}$, the whitening matrix reduces the dimension of $\mathbf{x}$ by removing the principal components with the smallest σ_i's.

Next, define a matrix $\mathbf{Q}$ such that $\mathbf{y} = \mathbf{Qz}$, $\mathbf{q}_i^T$ is the i^{th} row of $\mathbf{Q}$ and $y_i = \mathbf{q}_i^T\mathbf{z}$ is an independent component exactly when $\mathbf{q}_i$ is chosen such that y_i has maximum negentropy. Hence, an appropriate cost function (for minimization) is,

$$\phi(\mathbf{q}_i) = -J(p_Y(y_i)) = -J(p_Y(\mathbf{q}_i^T\mathbf{z})). \tag{8.8}$$

As already mentioned, whitening the data further constrains the cost function. In particular, recalling that $\mathrm{var}(y_i) = 1$, $\mathrm{E}(y_i) = 0$ and that the independent components are uncorrelated such that $\mathrm{E}(y_iy_j) = 0$, $i \neq j$ gives,

$$\mathrm{E}(y_iy_j) = \mathrm{E}\left[(\mathbf{q}_i^T\mathbf{z})(\mathbf{q}_j^T\mathbf{z})^T\right] = \mathbf{q}_i^T\mathrm{E}(\mathbf{z}\mathbf{z}^T)\mathbf{q}_j = \mathbf{q}_i^T\mathbf{q}_j = \begin{cases} 0 & , \quad i \neq j \\ 1 & , \quad i = j. \end{cases}$$

Thus, the cost function need only be considered on the surface defined by $\mathbf{q}_i^T\mathbf{q}_i = 1$, and multiple local minima may be found using orthogonality.

Once the optimal $\mathbf{Q}$ is found by FastICA (Appendix F.2), the basis $\mathbf{P}$ for sparse coding is readily computed. In particular, noting that $\mathbf{Q}$ is orthogonal, $\mathbf{y} = \mathbf{Qz}$ and $\mathbf{z} = \mathbf{\Gamma x}$, we find $\mathbf{x} = \mathbf{\Gamma}^{-1}\mathbf{Q}^T\mathbf{y}$, where $\mathbf{\Gamma}^{-1}$ is the left inverse of $\mathbf{\Gamma}$. Hence, it follows from equation 8.1 that $\mathbf{P} = \mathbf{\Gamma}^{-1}\mathbf{Q}^T$. The sparse code is given by $\mathbf{y} = \mathbf{Qz}$, so that the sparse coding representation of the data $\mathbf{x}$ is complete.

8.3 Filtering the sparse code

In the preceding section, we reviewed the sparse coding representation of data $d(x,t)$, building the random vector $\mathbf{x}$, and its representation with the sparse code $\mathbf{y}$ and basis $\mathbf{P}$. When $\mathbf{x}$ is corrupted with noise, equation 8.1 becomes,

$$\hat{\mathbf{x}} = \sum_{i=1}^{n} \hat{y}_i\hat{\mathbf{p}}_i = \hat{\mathbf{P}}\hat{\mathbf{y}}, \tag{8.9}$$

where $\hat{\mathbf{x}} = \mathbf{x} + \mathbf{n}$, and $\mathbf{n}$ is a noise vector.

8.3.1 Sparse code shrinkage for Gaussian noise

If the noise is Gaussian, we follow Hoyer (1999), and shrink the coefficients so that for the i^{th} random variable in the sparse code $\mathbf{y}$ (see Appendix F.3),

$$y_i = \frac{1}{1+\sigma^2 a}\text{sign}(\hat{y}_i)\max(0, |\hat{y}_i| - \sigma^2 b), \tag{8.10}$$

where σ is the estimate of the standard deviation of the noise, and a and b control the sparseness of y_i. For example, if $a = 1$ and $b = 0$, then y_i is expected to follow a Laplacian distribution, and if $a = 0$ and $b = 1$, then y_i is expected to follow a Gaussian distribution. The filtered data are $\mathbf{x} \approx \hat{\mathbf{P}}\mathbf{y}$. In the construction of equation 8.10, it is assumed that the noise random variables are independent.

8.3.2 $\tau - p$ truncation for multiple suppression

If $d(x,t)$ is a common midpoint gather after normal move-out correction, then the data consist of flat events (primaries), and the noise consists of curved events (multiples). We assume that some subset of the basis $\mathbf{p}_i$ represent the curved signal, say $\mathbf{p}_i$ for $i \in \mathcal{I}_n$, while the compliment set represents the flat primaries. Then the multiple attenuated data vector is,

$$\mathbf{x} \approx \hat{\mathbf{x}} - \sum_{i \in \mathcal{I}_n} y_i \mathbf{p}_i.$$

To choose the set $\mathcal{I}_n$ we,

1. map the i^{th} basis vector $\mathbf{p}_i \in \mathcal{R}^{MN}$ to its corresponding basis patch $\mathbf{V}_i \in \mathcal{R}^{M\times N}$ via lexicographic reordering, so that it has dimensions of time t and space x, that is $\mathbf{V}_i = \mathbf{V}_i(t,x)$;

2. Radon transform the i^{th} basis patch $\mathbf{V}_i(t,x)$, mapping it from the $t-x$ domain to the $\tau - p$ domain, giving $\mathbf{V}_i(\tau,p)$;

3. compute a $\tau - p$ sorting norm that measures the maximum ray-parameter in $\mathbf{V}_i(\tau,p)$,

$$m_i = \frac{\arg\max_{r.p.} |\mathbf{V}_i(\tau,p)|}{\delta(r.p.)} - \frac{||\mathbf{p}_i||_\infty}{||\mathbf{P}||_\infty}, \tag{8.11}$$

where $||\cdot||_\infty$ is the infinity norm (largest element of the matrix or vector), and $\delta(r.p.)$ is the sampling interval for the ray-parameter axis. The function $\arg\max_{r.p.} |\mathbf{V}_i(\tau,p)|$ is the ray-parameter corresponding to the maximum absolute value in the Radon panel $\mathbf{V}_i(\tau,p)$.

Given the above thee-step procedure, we define the set:

$$\mathcal{I}_n = \{i \,|\, m_i > \epsilon \max(m_i),\, i = 1 \ldots n\}\,, \tag{8.12}$$

where $\epsilon \in [0, 1]$ is some user-defined threshold. In particular, the larger we choose ϵ, the fewer sparse coding basis vectors $\mathbf{p}_i$ are used in the reconstructed multiples. The metric in equation 8.11 is chosen so that, all things being equal, a basis patch dominated by a large ray-parameter has a large norm. This is, in particular, measured by the first right-hand-side term in equation 8.11. The second right-hand-side term in equation 8.11 refines the sorting norm so that if the largest element in any given basis patch is large compared to the maximum value of the basis, the norm shrinks.

8.4 Example: Gaussian random noise attenuation

For our first example, we revisit sparse coding for random Gaussian noise attenuation applied to a synthetic seismic data example (Kaplan and Ulrych, 2005). The aim of the example is two-fold. First, it illustrates the nature of the sparse coding basis $\mathbf{p}_i$. Second, it shows their use for random noise attenuation.

Figure 8.1a is data $\hat{d}(t, x) = d(t, x) + n(t, x)$ with t being its vertical time axis and x its horizontal trace number axis. The noisy data $\hat{d}$ consists of three apex shifted hyperbolic events and three linear events, corrupted by band-limited Gaussian random noise n. We extract data patches $W_j \in \mathcal{R}^{32\times16}$ from $d(t, x)$ using a moving window. The set of windows $\{W_j\}$ sample the entirety of $d(t, x)$, and have redundancy so that the lower left corner of window W_j is shifted by either 5 traces or 5 time samples to give W_{j+1}. In total, 3478 realizations of W_j are collected. The data patches give the realizations of $\hat{\mathbf{x}}$ whose dimension is reduced from $m = 512$ to $n = 300$ using the whitening matrix $\mathbf{\Gamma}$. From the rank reduced and whitened data, we compute the sparse code $\hat{\mathbf{y}}$ and the basis $\hat{\mathbf{P}}$ (equation 8.9). The sparse coding basis patches $\mathbf{V}_i$ are mapped, lexicographically, from the columns of $\hat{\mathbf{P}}$. Interestingly, this data-driven basis, shown in Figure 8.2, has captured, locally, the behaviour of the signal.

We filter the components of the sparse code $\hat{\mathbf{y}}$ using the shrinkage function (equation 8.10) with $a = 0.8, b = 0.2$, giving $\mathbf{y}$, which along with $\hat{\mathbf{P}}$ is used to reconstruct $\mathbf{x}$, the filtered data vector. In equation 8.10 our estimate of the variance of the noise in the data is set to σ. Figure 8.1b is the sparse coding filtered data, say $f(\hat{d})$, recovered from $\mathbf{x}$ using the inverse of our windowing procedure. Finally, we plot the residual $(\hat{d} - f(\hat{d}))$ in Figure 8.1c.

In Figure 8.3 we compare the sparse coding result (Figure 8.3a) to those obtained by $f - x$ deconvolution (Figure 8.3b) and curvelet de-noising (Figure 8.3c). The curvelet de-noising example uses the CurveLab software package described in, for example, Candès et al. (2006),

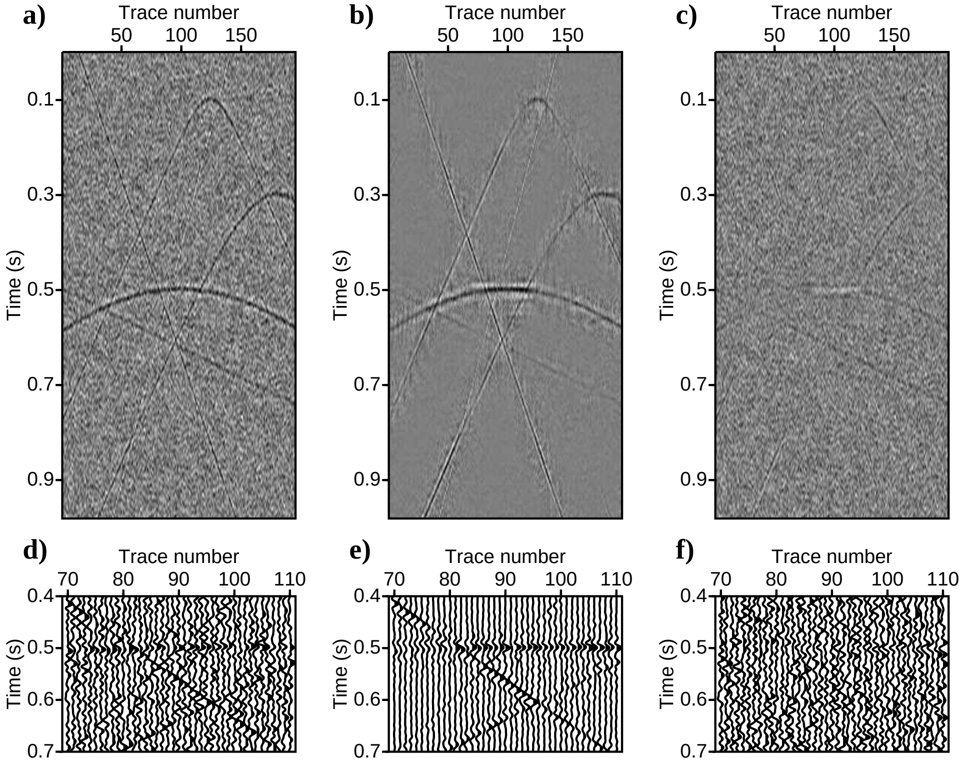

Figure 8.1: Random noise attenuation example: a) the input data, b) noise attenuation result using sparse coding, c) residual, or difference between a) and b). Plots d)-f) show a window from, respectively, a)-c).

and produces artifacts similar to those shown in their example. To make a quantitative comparison of the results, we show signal-to-noise ratios in Table 1.

8.5 Example: multiple attenuation

For our second example, we introduce a new application for sparse coding, applying it to multiple attenuation. In particular, we consider a single common midpoint (CMP) data gather which has undergone normal move-out (NMO) correction. Due to NMO, the primary

noisy data	sparse coding	$f - x$ deconvolution	curvelet transform
-1.5	0.8	2.4	2.9

Table 8.1: Table of signal-to-noise ratios in decibels for the noisy, and filtered data.

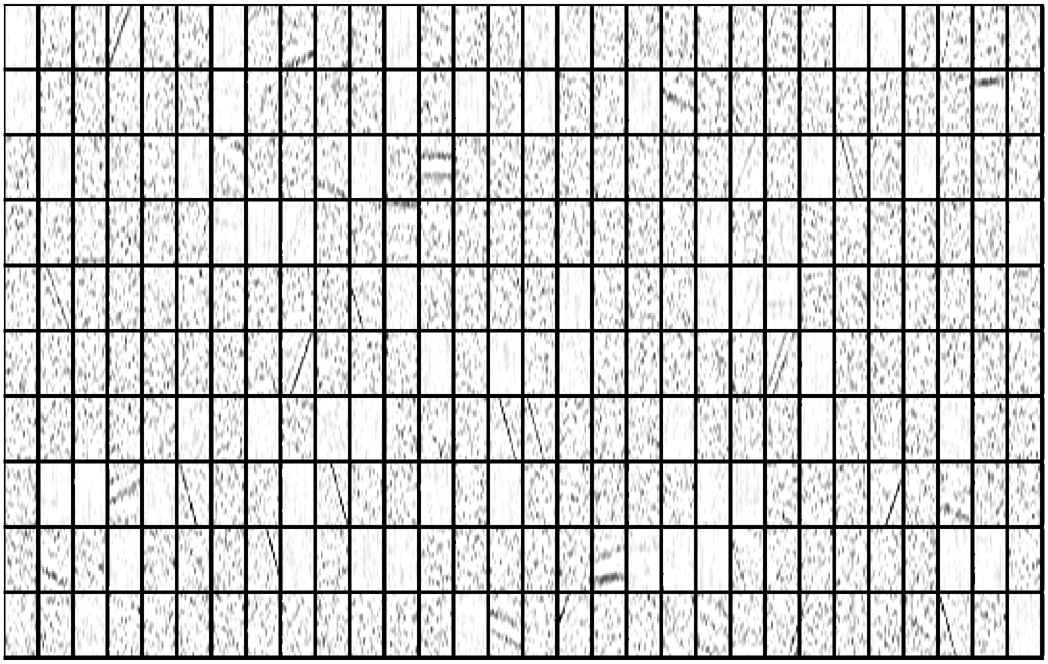

Figure 8.2: Random noise attenuation example: the sparse coding basis patches $\mathbf{V}_i$, $i = 1 \ldots 300$.

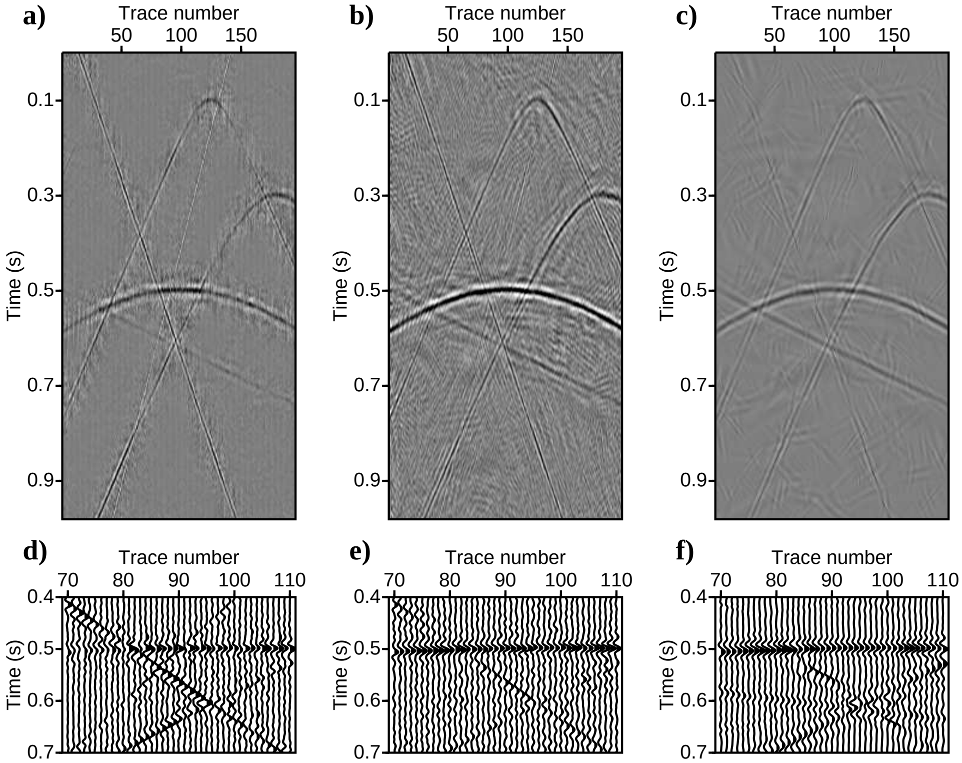

Figure 8.3: Random noise attenuation example: noise attenuation results using a) sparse coding, b) $f - x$ deconvolution, and c) curvelet transform. Plots d)-f) show a window from, respectively, a)-c).

events are flat, but the multiples are under-corrected, and are parabolic. The idea, then, is that the parabolic and flat events are captured by disjoint subsets of the sparse coding basis functions, allowing us to use the $\tau - p$ truncation filter to capture only the multiple events which are subsequently subtracted from the data.

To test this idea, we consider the real data example in Figure 8.4. In Figure 4a, we plot $d(t, x)$, the NMO corrected CMP gather. 3478 realizations of $\mathbf{x}$ are extracted from $d(t, x)$ using the same windowing procedure described in the previous section. Then, we compute the sparse code $\mathbf{y}$ and the basis $\mathbf{P}$, plotting the lexicographic reordered columns of $\mathbf{P}$ (the sparse coding basis patches) in Figure 8.5a. The ordering of the basis depends on the initialization of the FastICA optimization procedure. We sort the sparse coding basis by increasing norm m_i in equation 8.11. The re-ordered basis are shown in Figure 8.5b. We plot the values of m_i in Figures 8.5c and 8.5d for, respectively, the basis before and after sorting. The multiples are constructed by selecting ϵ so that the first 40 sorted sparse coding basis functions are excluded, giving the multiple estimate d_m in Figure 8.4b. Subtracting d_m from d give the multiple attenuated data in Figure 8.4c. For comparison, we plot estimated multiples using the high resolution parabolic Radon transform (Trad, 2003) in Figure 8.4e, and the corresponding multiple attenuated data in Figure 8.4f. We note that the artifacts in the Radon transform result are small in amplitude. Figures 8.4e and 8.4f are clipped to 30% of their maximum value.

8.6 Summary

When using sparse coding for noise attenuation, we are required to select three sets of parameters. First, in generating the sparse coding basis patches, we choose the size of the data patches $\mathbf{W}_j$, and how these patches sample the data (e.g. using a moving window). Second, for the FastICA algorithm, we choose the number of dimensions in $\mathbf{z}$, the whitened data vector. Third, we choose parameters that filter the sparse code. For random de-noising, we have three parameters, the variance of the noise, as well as a and b in equation 8.10. The latter two of which we use to control the sparsity of the underlying distribution of the sparse code. In multiple attenuation, we choose ϵ in equation 8.12, removing a subset of the basis for the reconstruction. In this chapter, we have chosen these parameters by inspection. For example, analysis of the error panel in the random noise attenuation example, guides our choice of a, b and σ. While we do not discuss the topic in this chapter, for random noise attenuation many of these parameters could be selected using methods from inverse theory where we not only ask that the sparse code follows a sparse distribution, but also that the reconstructed data falls close (within the expected variance of the noise) to the data (e.g. Beck and Teboulle, 2009). For multiple suppression, the burden for choosing parameters

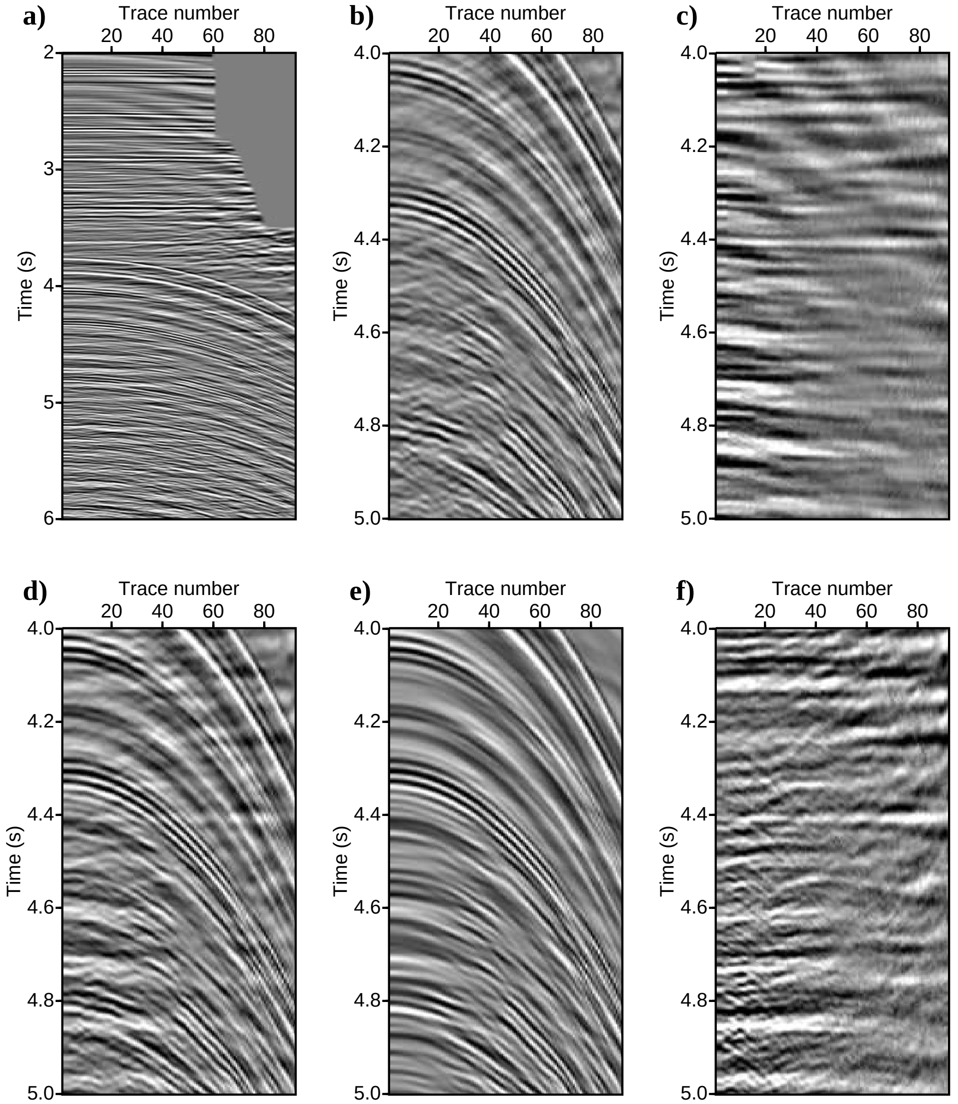

Figure 8.4: Multiple suppression example: a) the input data; b) sparse coding estimate of multiples; c) sparse coding multiple attenuation result, or the difference between d) and b); d) the input data (windowed); e) Parabolic Radon transform estimate of multiples; f) Parabolic Radon transform multiple attenuation result, or the difference between d) and e).

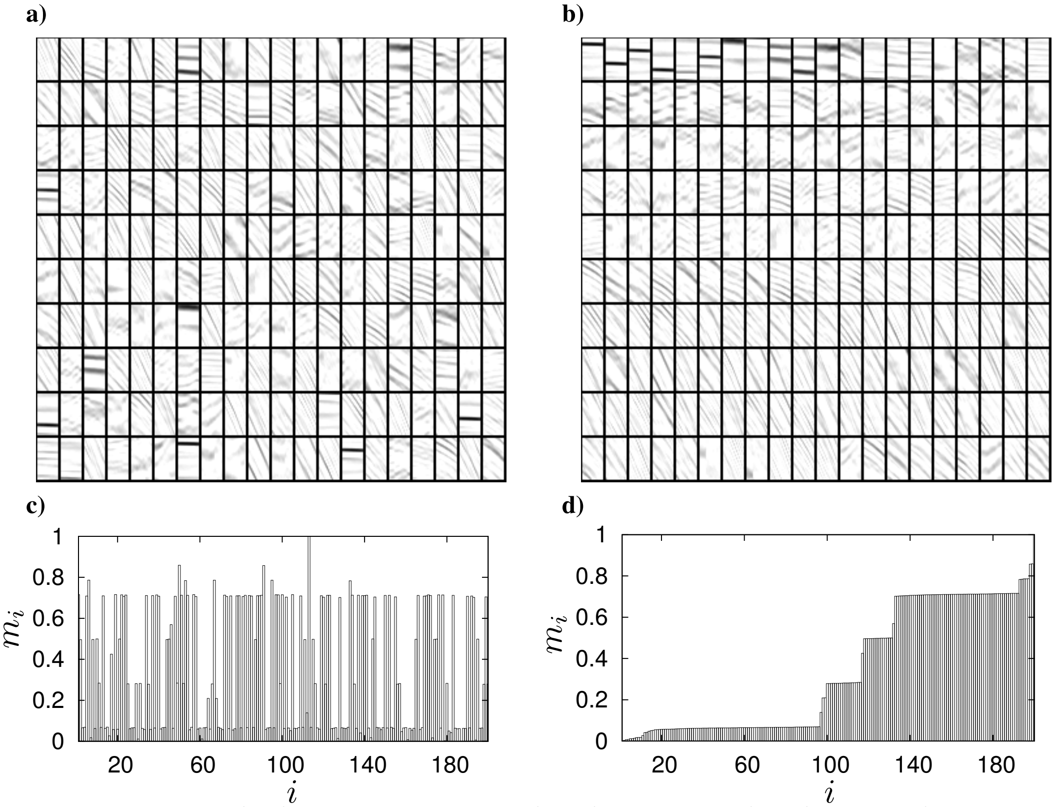

Figure 8.5: Multiple suppression example: the sparse coding basis patches $\mathbf{V}_i$, $i = 1 \dots 200$ a) before sorting and b) after sorting; the norm m_i (equation 8.11) versus the sparse coding basis index i c) before and d) after sorting.

(e.g. ϵ) is placed on the user of the algorithm who must interpret the ordered basis patches, and decide which basis patches span the primary events, and, likewise, which basis patches span the multiple events.

Similar statements can be made about the methods that we use, in this , chapter for comparison to sparse coding. For example, in the curvelet transform we choose parameters for shrinking the curvelet coefficients. In $f - x$ deconvolution, we window the data so that within each window the signal is approximately linear, as well as choosing the length of the filter and the amount of padding. In the Radon transform, we choose how to apply the mute in the Radon panel.

In the random noise attenuation examples, and for the purposes of this chapter, we have been aggressive in our threshold. This is true for both the curvelet transform example and the sparse coding example. We note that by being this aggressive, we likely filtered some signal (in addition to the noise), damaging signal-to-noise ratios. In practise, one might be less aggressive in choosing the thresholding parameters, giving less dramatic and more conservative results.

Noise attenuation methods such as the curvelet transform use basis functions that represent either signal or noise, but not both. In the data driven approach of sparse coding, it is clear that many of the sparse coding basis patches in Figure 8.2 contain both signal and noise. The result of this can be seeing in the reconstructed data where within the vicinity of the signal, the noise level increases. This is, of course, not a desirable feature of the sparse coding representation of seismic data, and is likely responsible for the small signal-to-noise ratio shown in Table 1.

We used sparse coding to attenuate both random and coherent (multiples) noise in seismic data. Sparse coding is used to estimate from the data themselves the basis functions required to represent signal and noise. In the random noise scenario, filtering can be implemented via thresholding the coefficients of the data expansion. In the coherent noise scenario, we selected a subset of the basis functions by using a $\tau - p$ sorting norm. In both cases, we compared sparse coding to existing methods. In particular, $f - x$ deconvolution and the curvelet transform for random noise, and the Radon transform for coherent noise.

■ CHAPTER 9

Conclusions

9.1 Summary

The primary subject of this thesis is regularized migration. It is used, both, to find a regularized approximation to the scattering potential, and as a tool for data reconstruction. Important to this thesis is our chosen parameterizations of the scattering potential (shot-receiver in Chapter 2 versus shot-profile in Chapter 5). The choice of parameterization influences our choice of model prior, and has practical implications for algorithm implementations and requirements. For example, it was exactly the shot-profile parameterization of the scattering potential that allowed for the data reconstruction algorithm in Chapter 6.

In Chapter 2, we derived source-receiver split-step wave-field modelling and migration operators. We took care to derive the operators from first principles before finding the expected and classic survey sinking interpretation for the migration operator. In Chapter 3 we provided an analysis of regularized (specifically, least-squares) migration using Ritz values. In Chapter 4, we used the expected distribution of energy in the source-receiver parameterization of the scattering potential to influence a sparse model prior. In Chapter 5, we derived wave-field modelling and migration algorithms under the shot-profile parameterization of the scattering potential. This parameterization provided a basis for the data reconstruction algorithm shown in Chapter 6. In Chapter 7, we described our implementation of the source-receiver regularized migration algorithm for distributed computing systems. Finally, in Chapter 8, we introduced a noise suppression algorithm by applying sparse coding to seismic data processing.

In Chapter 1, we described seismic data processing as an ordered sequence of tasks. In this thesis, our primary focus was on the task of migration. However, we additionally, applied migration to the task of data reconstruction, a step that precedes migration. We note that this does not present a contradiction. In particular, this is because in our development of our data reconstruction algorithms we used wave-field modelling and migration operators built from Green's functions propagating in a constant velocity reference wave-speed.

9.2 Contributions

The contributions of this thesis are as follows. First, we provide derivations of wave-field modelling and migration operators (Chapters 2 and 5). The resulting operators are, certainly, known in geophysics. However, a formal derivation of the operators is necessary in the context of regularized migration. In particular, this is because we require that the wave-field modelling and migration operators form a forward/adjoint pair. We note that Chapter 5 provides novel applications of least-squares migration to the shot-profile parameterization of the scattering potential. Second, we provide novel algorithms for seismic data processing: migration with a sparse model prior (Chapter 4), and shot-profile migration data reconstruction (Chapter 6). Third, we provide a novel parallel implementation of source-receiver least-squares migration (Chapter 7). Finally, we give two novel applications of sparse coding to seismic data processing; namely, random noise attenuation and multiple suppression.

9.3 Further work

In Chapter 1, we described five dimensional data collected in a reflection seismic experiment. While the theory in this thesis was, primarily, developed for five dimensional data, the implementation of the algorithms was done for three dimensions, such that two spatial dimensions were held constant. This is certainly not a desirable feature of this thesis; however, it is understandable given the size of five dimensional reflection seismic data (occupying Tera-Bytes of computer storage). As resources become available, extensions of the algorithms in this thesis to five dimensional data would be an interesting engineering task.

In Chapter 4, we used a sparse model prior to regularize the inversion. While this certainly gave some interesting results, the algorithm is not robust. We imagine that improvements could be obtained by allowing the scale parameter for the Cauchy distribution and the noise variance to vary as a function of space, allowing the character of the solution to, in turn, vary over space.

This thesis used the scalar wave-equation to approximate the physics of wave-propagation. This ignores the shear component of the wave-field, and assumes an isotropic earth. One may want to investigate the extension of the studied methods to, for example, an elastic and anisotropic earth model.

Bibliography

Amundsen, L., T. Rosten, J. O. A. Robertsson, and E. Kragh. “Rough-sea deghosting of streamer seismic data using pressure gradient approximation.” *Geophysics* 70 (2005): V1–V9.

Babu, P. and P. Stoica. “Spectral analysis of nonuniformly sampled data – a review.” *Digital Signal Processing* 20 (2010): 359–378.

Baumstein, A. and M. T. Hadidi. “3D surface-related multiple elimination: data reconstruction and application to field data.” *Geophysics* 71 (2006): E25–E33.

Bayes, T. “An essay towards solving a problem in the docterine of chances.” *Philosophical Transactions of the Royal Society of London* 53 (1763): 370–418.

Beck, A. and M. Teboulle. “A fast iterative shrinkage-thresholding algorithm for linear inverse problems.” *SIAM Journal on Imaging Sciences* 2 (2009): 183–202.

Bekara, M. and M. van der Baan. “Random and coherent noise attenuation by empirical mode decomposition.” *Geophysics* 74 (2009): V89–V98.

Bergsma, E. “Sigsbee2A 2D synthetic dataset.”
http://www.delphi.tudelft.nl/SMAART/sigsbee2a, accessed 9 February 2010 (2007).

Berkhout, A. J., G. Blacquiere, and D. J. Verschuur. “The concept of double blending: combining incoherent shooting with incoherent sensing.” *Geophysics* 74 (2009): A59–A62.

Biondi, B. “Short note: equivalence of source-receiver and shot-profile migration.” *Geophysics* 68 (2003): 1340–1347.

Canales, L. “Random noise reduction.” *54th Annual International Meeting, SEG, Expanded Abstracts* (1984): 525–527.

Candès, E., L. Demanet, D. Donoho, and L. Ying. “Fast discrete curvelet transforms.” *Multiscale Modeling & Simulation* 5 (2006): 861–899.

Candès, E. J. and D. L. Donoho. “New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities.” *Communications on Pure and Applied Mathematics* 57 (2004): 219–266.

Chiu, S. K. and R. Stolt. “Applications of 3-D data mapping – azimuth moveout and acquistion-footprint reduction.” *72nd Annual International Meeting, SEG, Expanded Abstracts* (2002): 2134–2137.

Chunduru, R. K., M. K. Sen, and P. L. Stoffa. “Hybrid optimization methods for geophyisical inversion.” *Geophysics* 62 (1997): 1196–1207.

Claerbout, J. F. “Towards a unified theory of reflector mapping.” *Geophysics* 36 (1971): 467–481.

Claerbout, J. F. *Earth sounding analysis: processing versus inversion.* Blackwell Science, 1992.

Clayton, R. W. and R. H. Stolt. “A Born-WKBJ inversion method for acoustic reflection data.” *Geophysics* 46 (1981): 1559–1567.

Cohen, J. K. and N. Bleistein. “Velocity inversion procedure for acoustic waves.” *Geophysics* 44 (1979): 1077–1087.

Common, P. “Independent component analysis, a new concept?” *Signal Processing* 36 (1994): 287–314.

Cover, T. M. and J. A. Thomas. *Elements of information theory.* 2nd edition. Wiley-interscience, 2006.

Dai, W. and J. Schuster. “Least-squares migration of simultaneous sources data with a deblurring filter.” *79th Annual International Meeting, SEG, Expanded Abstracts* (2009): 2990–2993.

Deng, F. and G. A. McMechan. “Viscoelastic true-amplitude prestack reverse-time depth migration.” *Geophysics* 73 (2008): S157–S168.

DeSanto, J. A. *Scalar wave theory: Green’s functions and applications.* Volume 12 of Springer Series on Wave Phenomena. Springer-Verlag, 1992.

Feit, M. D. and Jr. J. A. Fleck. “Light propagation in graded-index optical fibers.” *Applied Optics* 17 (1978): 3990–3998.

Fomel, S. “Shaping regularization in geophysical-estimation problems.” *Geophysics* 72 (2007): R29–R36.

Foster, D. J. and C. C. Mosher. “Suppression of multiple reflections using the Radon transform.” *Geophysics* 57 (1992): 386–395.

Freire, S. L. M. and T. J. Ulrych. “Application of singular value decomposition to vertical seismic profiling.” *Geophysics* 53 (1988): 83–91.

Galbraith, M. “Random noise attenuation by $f - x$ prediction: a tutorial.” *61st Annual International Meeting, SEG, Expanded Abstracts* (1991): 1428–1431.

Gazdag, J. “Wave equation migration with phase-shift method.” *Geophysics* 43 (1978): 1342–1351.

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

Gersztenkorn, A., J. B. Bednar, and L. R. Lines. “Robust iterative inversion for the one-dimensional acoustic wave equation.” *Geophysics* 51 (1986): 357–368.

Ghatak, A., R. Gallawa, and I. Goyal. *Modified Airy function and WKB solutions to the wave equation.* Volume 176 of NIST Monograph. United States Department of Commerce, National Institute of Standards and Technology, 1991.

Gülünay, N. “FXDECON and complex Wiener prediction filter.” *56th Annual International Meeting, SEG, Expanded Abstracts* (1986): 279–281.

Gülünay, N. “Seismic trace interpolation in the Fourier transform domain.” *Geophysics* 68 (2003): 355–369.

Gupta, A., V. Kumar, and A. Sameh. “Performance and scalability of preconditioned conjugate gradient methods on parallel computers.” *IEEE Transactions on Parallel and Distributed Systems* 6 (1995): 455–469.

Hampson, D. “Inverse velocity stacking for multiple elimination.” *Journal of the Canadian Society of Exploration Geophysicists* 22 (1986): 44–55.

Hansen, P. C. *Rank-deficient and discrete ill-posed problems.* Siam, 1998.

Hennenfent, G., E. van den Berg, M. P. Friedlander, and F. J. Herrmann. “New insights into one-norm solvers from the Pareto curve.” *Geophysics* 73 (2008): A23–A26.

Hermmann, F. J. “Compressive imaging by wavefield inversion with group sparsity.” *79th Annual International Meeting, SEG, Expanded Abstracts* (2009): 2337–2340.

Herrmann, F. J. and G. Hennenfent. “Non-parametric seismic data recovery with curvelet frames.” *Geophysical Journal International* 173 (2008): 233–248.

Hestenes, M. R. and E. Stiefel. “Methods of conjugate gradients for solving linear systems.” *Journal of Research of the National Bureau of Standards* 49 (1952): 409–438.

Hoyer, P. Independent component analysis in image denoising. Master’s thesis, Helsinki University of Technology, 1999.

Huang, L., M. C. Fehler, and R. Wu. “Extended local Born Fourier migration method.” *Geophysics* 64 (1999): 1524–1534.

Huang, N. E., Z. Shen, S. R. Long, M. L. Wu, H. H. Shih, Q. Zheng, N. C. Yen, C. C. Tung, and H. H. Liu. “The empirical mode decomposition and Hilbert spectrum for nonlinear and nonstationary time series analysis.” *Proceedings of the Royal Society of London. Series A* 454 (1998): 903–995.

Hyvärinen, A. “New approximations of differential entropy for independent component analysis and projection pursuit.” *Advances in Neural Information Processing Systems.* Ed. M. I. Jordan, M. J. Kearns, and S. A. Solla 1998, 273–279.

Hyvärinen, A. “Fast and robust fixed-point algorithms for independent component analysis.” *IEEE Transactions on Neural Networks* 10 (1999): 626–634.

Hyvärinen, A., P. Hoyer, and E. Oja. “Image denoising by sparse code shrinkage.” *Intelligent Signal Processing.* Ed. S. Haykin and B. Kosko 1999, 554–568.

Innanen, K. A. and A. B. Weglein. “Two non-linear forward and inverse approximations for wavefields in the presence of sustained perturbations.” *75th Annual International Meeting, SEG, Expanded Abstracts* (2005): 2362–2365.

Irons, T. “Sigsbee2 Models.” *http://www.reproducibility.org/RSF/book/data/sigsbee/paper_html/paper.html, accessed 9 October 2009* (2007).

Jeannot, J. P. “Full prestack versus shot record migration: practical aspects.” *58th Annual International Meeting, SEG, Expanded Abstracts* (1988): 966–968.

Johnson, N. L. and S. Kotz. *Distribution in statistics: continuous multivariate distributions.* John Wiley & Sons Inc, 1972.

Kaplan, S. T. and T. J. Ulrych. “From eigenfaces to eigensections.” *Journal of Seismic Exploration* 10 (2002): 353–366.

Kaplan, S. T. and T. J. Ulrych. “Noise suppression in seismic data with sparse coding.” *9th International Congress, SBGf, Expanded Abstracts* (2005).

Kaplan, Sam T. "Pre-stack sparse least-squares migration." *70th Annual International Meeting, EAGE, Expanded Abstracts* (2008).

Kessinger, W. "Extended split-step Fourier migration." *62nd Annual International Meeting, SEG, Expanded Abstracts* (1992): 917–920.

Kober, H. "The extended Weyl integral and related operations." *Proceedings of the American Mathematical Society* 19 (1968): 285–291.

Kowalski, M. and B. Torrésani. "Sparsity and persistence: mixed norms provide simple signal models with dependent coefficients." *Signal, Image and Video Processing* 3 (2009): 251–264.

Kuehl, H. *Least-squares wave-equation migration/inversion.* PhD thesis, University of Alberta, 2002.

Kühl, H. and M. Sacchi. "Least-squares wave-equation migration for AVP/AVA inversion." *Geophysics* 68 (2003): 262–273.

Lathi, B. P. *Signal processing and linear systems.* Berkeley Cambridge Press, 1998.

Lawson, C. L. and R. J. Hanson. *Solving least squares problems.* Prentice-Hall, Inc., 1974.

Liu, B. and M. D. Sacchi. "Minimum weighted norm interpolation of seismic records." *Geophysics* 69 (2004): 1560–1568.

Liu, F., A. B. Weglein, K. A. Innanen, and B. G. Nita. "Multi-dimensional seismic imaging using the inverse scattering series." *76th Annual International Meeting, SEG, Expanded Abstracts* (2006): 3026–3030.

Naghizadeh, M. and M. D. Sacchi. "Multistep autoregressive reconstruction of seismic records." *Geophysics* 72 (2007): V111–V118.

Nemeth, T., C. Wu, and G. T. Schuster. "Least-squares migration of incomplete reflection data." *Geophysics* 64 (1999): 208–221.

Oja, E., A. Hyvärinen, and P. Hoyer. "Image feature extraction and denoising by sparse coding." *Pattern Analysis & Applications* 2 (1999): 104–110.

Olshausen, B. A. and D. J. Field. "Sparse coding With an overcomplete basis set: a strategy employed by V1?" *Vision Research* 37 (1997): 3311–3325.

Ostrander, W. J. "Plane wave reflection coefficients for gas sands at nonnormal angles of incidence." *SEG Technical Program, Expanded Abstracts* (1982): 216–218.

Ottolini, R. and J. F. Claerbout. “The migration of common midpoint slant stacks.” *Geophysics* 49 (1984): 237–249.

Pidlisecky, A., E. Haber, and R. Knight. “RESINVM3D: a 3D resistivity inversions package.” *Geophysics* 72 (2007): H1–H10.

Pilkington, M. “3-D magnetic imaging using conjugate gradients.” *Geophysics* 62 (1997): 1132–1142.

Popovici, A. M. “Prestack migration by split-step DSR.” *Geophysics* 61 (1996): 1412–1416.

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

Ramírez, A. C. and A. B. Weglein. “Green’s theorem as a comprehensive framework for data reconstruction, regularization, wavefield separation, seismic interferometry, and wavelet estimation: A tutorial.” *Geophysics* 74 (2009): W35–W62.

Ramírez, A. C., A. B. Weglein, and K. Hokstad. “Near offset data extrapolation.” *76th Annual International Meeting, SEG, Expanded Abstracts* (2006): 2554–2558.

Rickett, J. E. *Spectral factorization of wavefields and wave operators.* PhD thesis, Stanford University, 2001.

Rickett, J. E. “Illumination-based normalization for wave-equation depth migration.” *Geophysics* 68 (2003): 1371–1379.

Rodriguez, I. V., M. D. Sacchi, and J. Gu. “Continuous hypocenter and source mechanism inversion via a Green’s function-based matching pursuit algorithm.” *The Leading Edge* 29 (2010): 334–337.

Saad, Y. and M. H. Schultz. “GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems.” *SIAM Journal on Scientific and Statistical Computing* 7 (1986): 856–869.

Sacchi, M. D., T. J. Ulrych, and C. J. Walker. “Interpolation and extrapolation using a high-resolution discrete Fourier Transform.” *IEEE Transactions on Signal Processing* 46 (1998): 31–38.

Sacchi, M. D., D. J. Verchuur, and P. M. Zwartjes. “Data reconstruction by generalized deconvolution.” *74th Annual International Meeting, SEG, Expanded Abstracts* (2004): 1989–1992.

Saff, E. B. and A. D. Snyder. *Fundamentals of complex analysis for mathematics, science, and engineering.* 2nd edition. Prentice Hall, 1993.

Sava, P. "Stereographic imaging condition for wave-equation migration." *Geophysics* 72 (2007): A87–A91.

Sava, P., B. Biondi, and S. Fomel. "Amplitude-preserved common image gathers by wave-equation migration." *71st Annual International Meeting, SEG, Expanded Abstracts* (2001): 296–299.

Sava, P. and S. Fomel. "Angle-domain common-image gathers by wavefield continuation methods." *Geophysics* 68 (2003): 1065–1074.

Sava, P. and S. J. Hill. "Overview and classification of wavefield seismic imaging methods." *The Leading Edge* 28 (2009): 170–183.

Scales, J. A. "On the use of conjugate gradient to calculate the eigenvalues and singular values of large, sparse matrices." *Geophysical Journal International* 97 (1989): 197–193.

Scales, J. A. and A. Gersztenkorn. "Robust methods in inverse theory: inverse problems." *Inverse Problems* 4 (1988): 1071–1091.

Schultz, P. S. and J. F. Claerbout. "Velocity estimation and downward continuation by wavefront synthesis." *Geophysics* 43 (1978): 691–714.

Shaw, S. A., A. B. Weglein, F. J. Foster, K. H. Matson, and R. G. Keys. "Isolation of leading order depth imaging series and analysis of its convergence properties for a 1D acoustic medium." *Journal of Seismic Exploration* 13 (2004): 99–120.

Shearer, P. M. *Introduction to seismology.* Cambridge University Press, 1999.

Sollid, A. and B. Ursin. "Scattering-angle migration of ocean-bottom seismic data in weakly anisotropic media." *Geophysics* 68 (2003): 641–655.

Spitz, S. "Seismic trace interpolation in the F-X domain." *Geophysics* 56 (1991): 785–794.

Starck, J., E. J. Candés, and D. L. Donoho. "The curvelet transform for image denoising." *IEEE Transactions on Image Processing* 11 (2002): 670–684.

Stoffa, P., J. Fokkema, R. de Luna Freire, and W. Kessinger. "Split-step Fourier migration." *Geophysics* 55 (1990): 410–421.

Stolt, R. H. "Migration by Fourier transform." *Geophysics* 43 (1978): 23–48.

Stolt, R. H. "Seismic data mapping and reconstruction." *Geophysics* 67 (2002): 890–908.

Stolt, R. H. and A. K. Benson. *Seismic migration: theory and practice.* Ed. K. Helbig and S. Treitel. Volume 5 of Seismic Exploration. Geophysical Press, 1986.

Stork, C. "Reflection tomography in the postmigrated domain." *Geophysics* 57 (1992): 680–692.

Thomson, D. J. and N. R. Chapman. "A wide-angle split-step algorithm for the parabolic equation." *The Journal of the Acoustical Society of America* 74 (1983): 1848–1854.

Trad, D. "Interpolation and multiple attenuation with migration operators." *Geophysics* 68 (2003): 2043–2054.

Trad, D., T. J. Ulrych, and M. D. Sacchi. "Latest views of the sparse Radon transform." *Geophysics* 68 (2003): 386–399.

Ulrych, T. J., M. D. Sacchi, and A. Woodbury. "A Bayes tour of inversion." *Geophysics* 66 (2001): 55–69.

Verschuur, D. J., A. J. Berkhout, and C. P. A. Wapenaar. "Adaptive surface-related multiple elimination." *Geophysics* 57 (1992): 1166–1177.

Versteeg, R. "Sensitivity of prestack depth migration to the velocity model." *Geophysics* 58 (1993): 873–882.

Versteeg, R. "The Marmousi experience: velocity model determination on a synthetic complex data set." *The Leading Edge* 18 (1994): 86–91.

Vinje, W. E. and J. L. Gallant. "Sparse coding and decorrelation in primary visual cortex during natural vision." *Science* 287 (2000): 1273–1276.

Wang, J. *3-D least-squares wave-equation AVP/AVA migration of common azimuth data.* PhD thesis, University of Alberta, 2005.

Wang, J. and M. D. Sacchi. "High-resolution wave-equation amplitude-variation-with-ray-parameter (AVP) imaging with sparseness constraints." *Geophysics* 72 (2007): S11–S18.

Wapenaar, C. P. A. and A. J. Berkhout. "Full prestack versus shot record migration." *69th Annual International Meeting, SEG, Expanded Abstracts* (1987): 761–764.

Weglein, A. B., F. V. Araújo, P. M. Carvalho, R. H. Stolt, K. H. Matson, R. T. Coats, D. Corrigan, D. J. Foster, S. A. Shaw, and H. Zhang. "Inverse scattering series and seismic exploration." *Inverse Problems* 19 (2003): R27–R83.

Weglein, A. B., F. A. Gasparotto, P. M. Carvalho, and R. H. Stolt. "An inverse-scattering series method for attenuating multiples in seismic reflection data." *Geophysics* 62 (1997): 1975–1989.

Weglein, A. B. and B. G. Secrest. "Wavelet estimation for a multidimensional acoustic earth model." *Geophysics* 55 (1990): 902–913.

Welford, J. K. and R. Zhang. "Ground-roll suppression from deep crustal seismic reflection data using a wavelet-based approach: a case study from western Canada." *Geophysics* 69 (2004): 877–884.

Wenzel, F. "Frequency-wavenumber migration in laterally heterogeneous media." *Geophysics* 56 (1991): 1671–1673.

Wiggins, R. A. "Minimum entropy deconvolution." *Geoexploration* 16 (1978): 21–35.

Yu, J., J. Hu, G. T. Schuster, and R. Estill. "Prestack migration deconvolution." *Geophysics* 71 (2006): S53–S62.

■ APPENDIX A

Derivations for wave-field modelling within the Born approximation

A.1 The Born approximation

To derive equation 2.4, we begin with the perturbed Helmholtz equation shown in equation 2.1, and introduce a perturbation in wave-speed using the scattering potential α and the reference wave-speed c_0. In particular, we write,

$$(\omega/c(\mathbf{x},z))^2 = (\omega/c_0(\mathbf{x},z))^2\left(1-\alpha(\mathbf{x},z)\right), \tag{A.1}$$

where $\alpha(\mathbf{x},z) = 1-(c_0(\mathbf{x},z)/c(\mathbf{x},z))^2$. Substituting equation A.1 into equation 2.1, gives,

$$\mathcal{L}\left(\psi,c_0\right) = f(\omega)\delta(\mathbf{x}-\mathbf{x}_s)\delta(z-z_s) + (\omega/c_0(\mathbf{x},z))^2\alpha(\mathbf{x},z)\psi(\mathbf{x},z|\mathbf{x}_s,z_s;\omega), \tag{A.2}$$

where, assuming a Green's function G_0 such that $\mathcal{L}(G_0,c_0) = \delta(\mathbf{x}-\mathbf{x}_s)\delta(z-z_s)$, allows us to write the Lippmann-Schwinger equation,

$$\begin{aligned}\psi(\mathbf{x},z|\mathbf{x}_s,z_s;&\omega) = f(\omega)G_0(\mathbf{x},z|\mathbf{x}_s,z_s;\omega)\\ &+\iint_{-\infty}^{\infty}\left(\frac{\omega}{c_0(\mathbf{x}',z')}\right)^2 G_0(\mathbf{x},z|\mathbf{x}',z';\omega)\alpha(\mathbf{x}',z')\psi(\mathbf{x}',z'|\mathbf{x}_s,z_s;\omega)d\mathbf{x}'dz'.\end{aligned} \tag{A.3}$$

From equation A.3, the Born (forward scattering) series is derived. In particular, defining $\psi_s = \psi - f(\omega)G_0$ to be the scattered wave-field, and using the recursion apparent in the

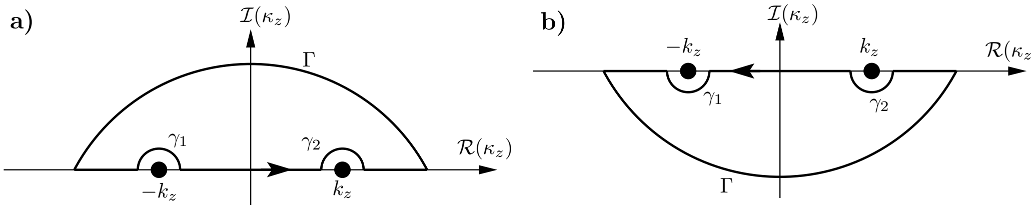

Figure A.1: Contours for evaluating the Weyl integral. a) for $z > z_s$ and b) for $z < z_s$.

Lippmann-Schwinger equation, we have,

$$\psi_s(\mathbf{x}, z|\mathbf{x}_s, z_s; \omega) = \iint_{-\infty}^{\infty} \frac{\omega^2 f(\omega)}{c_0^2(\mathbf{x}', z')} G_0(\mathbf{x}, z|\mathbf{x}', z'; \omega)\alpha(\mathbf{x}', z')G_0(\mathbf{x}', z'|\mathbf{x}_s, z_s; \omega)d\mathbf{x}'dz' \\ + \iint_{-\infty}^{\infty} \left(\frac{\omega}{c_0(\mathbf{x}', z')}\right)^2 G_0(\mathbf{x}', z'|\mathbf{x}'', z''; \omega)\alpha(\mathbf{x}'', z'')\psi(\mathbf{x}'', z''|\mathbf{x}_s, z_s; \omega)d\mathbf{x}''dz''.$$

Continuing the recursion yields the full Born series, and truncating at the first term, and evaluating at the measurement surface gives the desired Born approximation in equation 2.4.

A.2 Constant velocity Green's function

We derive the Green's functions in equations 2.6 and 2.7 for when the reference velocity c_0 is constant. Again, we begin with the Helmholtz operator $\mathcal{L}$ defined in equation 2.1, and define a Green's function G_0 such that $\mathcal{L}(G_0, c_0) = \delta(\mathbf{x} - \mathbf{x}_s)\delta(z - z_s)$. Then, taking the Fourier transform with respect the spatial coordinates ($\mathbf{x}$ and z), we find,

$$\left((\omega/c_0)^2 - (\mathbf{k}_x \cdot \mathbf{k}_x) - \kappa_z^2\right) G_0(\mathbf{k}_x, \kappa_z|\mathbf{x}_s, z_s; \omega) = e^{-i(\mathbf{k}_x \cdot \mathbf{x}_s + \kappa_z z_s)},$$

where κ_z is the Fourier conjugate variable of z, and $\mathbf{k}_x$ is the Fourier conjugate variable of $\mathbf{x}$. Solving for G_0, we find,

$$G_0(\mathbf{k}_x, \kappa_z|\mathbf{x}_s, z_s; \omega) = \frac{e^{-i(\mathbf{k}_x \cdot \mathbf{x}_s + \kappa_z z_s)}}{(\kappa_z - k_z)(\kappa_z + k_z)},$$

where we have used the following definition,

$$k_z^2 = (\omega/c_0)^2 - \mathbf{k}_x \cdot \mathbf{k}_x. \tag{A.4}$$

Taking the inverse Fourier transform over z gives the Weyl integral (e.g. Kober, 1968),

$$G_0(\mathbf{k}_x, z|\mathbf{x}_s, z_s; \omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \frac{e^{-i(\mathbf{k}_x\cdot\mathbf{x}_s)}e^{i\kappa_z(z-z_s)}}{(\kappa_z - k_z)(\kappa_z + k_z)} d\kappa_z. \tag{A.5}$$

Equation A.5 has two poles, both of order 1. Hence, we can solve equation A.5 using residue theory, (e.g. Saff and Snyder, 1993), the end result being the Green's function in equations 2.6 and 2.7.

First, we consider the case where $z > z_s$. We let κ_z be complex, and construct the contour in Figure A.1a such that it does not enclose the poles of the integrand $f(\kappa_z)$ in equation A.5. Hence, by Cauchy's integral theorem,

$$\left(\int_{-\infty}^{-k_z-r_1} + \int_{-k_z+r_1}^{k_z-r_2} + \int_{k_z+r_2}^{\infty} + \int_{\gamma_1} + \int_{\gamma_2} + \int_{\Gamma}\right) f(\kappa_z)d\kappa_z = 0, \tag{A.6}$$

where γ_1, γ_2 and Γ are contours shown in Figure A.1a, and the first three integrals are constrained to the real axis. In the limit as r_1 and r_2 go to zero, the first three integrals in equation A.6 collapse to one from $-\infty$ to ∞, along the real axis. Further, our choice $z > z_s$, allows the application of Jordan's lemma, so that the Γ contour integral vanishes. Hence,

$$\begin{aligned}\int_{-\infty}^{\infty} f(\kappa_z)d\kappa_z &= -\left(\int_{\gamma_1} + \int_{\gamma_2}\right) f(\kappa_z)d\kappa_z \\ &= \pi i\, Res(f(\kappa_z), k_z) + \pi i\, Res(f(\kappa_z); -k_z) \\ &= \frac{\pi i}{2k_z} e^{-i(\mathbf{k}_x\cdot\mathbf{x}_s)}e^{ik_z(z-z_s)} - \frac{\pi i}{2k_z} e^{-i(\mathbf{k}_x\cdot\mathbf{x}_s)}e^{-ik_z(z-z_s)}. \end{aligned} \tag{A.7}$$

Using the $e^{i\omega t}$ convention for the Fourier transform over time t (Clayton and Stolt, 1981), we can show, by plane wave analysis, that for $\omega > 0$, the first term in equation A.7 is causal, while the second is anti-causal. We will retain only the causal part of the solution in our final answer. Hence, for $z > z_s$, we have,

$$G_0(\mathbf{k}_x, z|\mathbf{x}_s, z_s; \omega) = -\frac{1}{i4k_z} e^{-i(\mathbf{k}_x\cdot\mathbf{x}_s)}e^{ik_z(z-z_s)}. \tag{A.8}$$

Performing a similar analysis for $z < z_s$, we use the contour in Figure A.1b so that, again, Jordan's lemma is applied to the Γ contour integral, and find,

$$\begin{aligned}\int_{-\infty}^{\infty} f(\kappa_z)d\kappa_z &= \left(\int_{\gamma_1} + \int_{\gamma_2}\right) f(\kappa_z)d\kappa_z \\ &= -\pi i\, Res(f(\kappa_z), k_z) - \pi i\, Res(f(\kappa_z), -k_z) \\ &= -\frac{\pi i}{2k_z} e^{-i(\mathbf{k}_x\cdot\mathbf{x}_s)}e^{ik_z(z-z_s)} + \frac{\pi i}{2k_z} e^{-i(\mathbf{k}_x\cdot\mathbf{x}_s)}e^{-ik_z(z-z_s)}, \end{aligned} \tag{A.9}$$

with the first term being anti-causal, and the second term being causal. Retaining only the causal term gives for $z < z_s$,

$$G_0(\mathbf{k}_x, z|\mathbf{x}_s, z_s; \omega) = -\frac{1}{i4k_z} e^{-i(\mathbf{k}_x \cdot \mathbf{x}_s)} e^{-ik_z(z-z_s)}. \tag{A.10}$$

Finally, combining the results for $z > z_s$ (equation A.8) and $z < z_s$ (equation A.10), gives for all z,

$$G_0(\mathbf{k}_x, z|\mathbf{x}_s, z_s; \omega) = -\frac{1}{i4k_z} e^{-i(\mathbf{k}_x \cdot \mathbf{x}_s)} e^{ik_z|z-z_s|}, \tag{A.11}$$

where k_z is given by the dispersion relation in equation A.4, so that,

$$k_z = \pm\sqrt{(\omega/c_0)^2 - \mathbf{k}_x \cdot \mathbf{k}_x}, \tag{A.12}$$

which, when evaluated at the measurement surface, is the same as the Green's function in equation 2.6. To ensure a causal solution, we retain the following part of equation A.12:

$$k_z = \operatorname{sgn}(\omega)\sqrt{(\omega/c_0)^2 - \mathbf{k}_x \cdot \mathbf{k}_x}. \tag{A.13}$$

To arrive at Green's function in equation 2.7, we take the inverse Fourier transform over lateral dimensions $\mathbf{x}$ in equation A.11, so that,

$$\begin{aligned} G_0(\mathbf{x}, z|\mathbf{x}_s, z_s; \omega) &= \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} -\frac{1}{i4k_z} e^{-i(\mathbf{k}_x \cdot \mathbf{x}_s)} e^{ik_z|z-z_s|} e^{i(\mathbf{k}_x \cdot \mathbf{x})} d\mathbf{k}_x & \text{(A.14)} \\ &= \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} -\frac{1}{i4k_{sz}} e^{i(\mathbf{k}_{sx} \cdot \mathbf{x})} e^{ik_{sz}|z-z_s|} e^{-i(\mathbf{k}_{sx} \cdot \mathbf{x}_s)} d\mathbf{k}_{sx}. & \text{(A.15)} \end{aligned}$$

Hence,

$$G_0(\mathbf{x}, z|\mathbf{k}_{sx}, z_s; \omega) = -\frac{1}{i4k_{sz}} e^{i(\mathbf{k}_{sx} \cdot \mathbf{x})} e^{ik_{sz}|z-z_s|},$$

which when evaluated at the measurement surface is the same as equation 2.7.

A.3 Gazdag wave-field modelling

We derive the result in equation 2.16, the l^{th} term in the Gazdag modelling operator (equation 2.10). In the first layer D_1, we consider the previously discussed differential equation,

$$\mathcal{L}(\psi_{(1)}, c_{0(1)}) = f(\omega)\delta(\mathbf{x} - \mathbf{x}_s)\delta(z - z_0) + \frac{\omega^2}{c_{0(1)}^2}\alpha(\mathbf{x}, z)\psi_{(1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega),$$

so that using the constant velocity Green's function $G_{0(1)}$,

$$\begin{aligned}\psi_{(1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega) &= f(\omega) G_{0(1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega) \\ &+ \int_{-\infty}^{\infty}\int_{z_0}^{z_1} G_{0(1)}(\mathbf{x}, z|\mathbf{x}', z'; \omega)\frac{\omega^2}{c_{0(1)}^2}\alpha(\mathbf{x}', z')\psi_{(1)}(\mathbf{x}', z'|\mathbf{x}_s, z_0; \omega) dz' d\mathbf{x}'.\end{aligned} \tag{A.16}$$

Next, we recognize the recursion in equation A.16, finding to first order in α,

$$\begin{aligned}\psi_{(1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega) &\approx f(\omega) G_{0(1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega) \\ &+ f(\omega)\int_{-\infty}^{\infty}\int_{z_0}^{z_1} G_{0(1)}(\mathbf{x}, z|\mathbf{x}', z'; \omega)\frac{\omega^2}{c_{0(1)}^2}\alpha(\mathbf{x}', z') G_{0(1)}(\mathbf{x}', z'|\mathbf{x}_s, z_0; \omega) dz' d\mathbf{x}'.\end{aligned} \tag{A.17}$$

The second term on the right-hand-side of equation A.17 is equivalent to $\psi_{s(1)}$ in equation 2.16. To show this, we evaluate it at the measurement surface $(\mathbf{x}_g, z_0)$, and take the Fourier transforms over $\mathbf{x}_g$ and $\mathbf{x}_s$ so that,

$$\begin{aligned}&\psi_{s(1)}(\mathbf{k}_{gx}, z|\mathbf{k}_{sx}, z_0; \omega) \\ &= f(\omega)\int_{-\infty}^{\infty}\int_{z_0}^{z_1} G_{0(1)}(\mathbf{k}_{gx}, z_0|\mathbf{x}', z'; \omega)\frac{\omega^2}{c_{0(1)}^2}\alpha(\mathbf{x}', z') G_{0(1)}(\mathbf{x}', z'|\mathbf{k}_{sx}, z_0; \omega) dz' d\mathbf{x}'.\end{aligned} \tag{A.18}$$

Next, we substitute the constant velocity Green's functions in equations 2.6 and 2.7 while assuming that the support for α is below z_0, giving,

$$\begin{aligned}&\psi_{s(1)}(\mathbf{k}_{gx}, z_0|\mathbf{k}_{sx}, z_0; \omega) = f(\omega) \\ &\times \int_{-\infty}^{\infty}\int_{z_0}^{z_1}\left(-\frac{e^{ik_{gz(1)}(z'-z_0)}}{i4k_{gz(1)}}e^{-i\mathbf{k}_{gx}\cdot\mathbf{x}'}\right)\frac{\omega^2}{c_{0(1)}^2}\alpha(\mathbf{x}', z')\left(-\frac{e^{ik_{sz(1)}(z'-z_0)}}{i4k_{sz(1)}}e^{i\mathbf{k}_{sx}\cdot\mathbf{x}'}\right) dz' d\mathbf{x}'.\end{aligned} \tag{A.19}$$

Finally, recognizing the Fourier kernel $\exp(-i(\mathbf{k}_{gx} - \mathbf{k}_{sx})\cdot\mathbf{x}')$ allows us to express equation A.19 using the Fourier transform, with respect to $\mathbf{x}'$, of α so that,

$$\begin{aligned}\psi_{s(1)}(\mathbf{k}_{gx}, z_0|\mathbf{k}_{sx}, z_0; \omega) &= f(\omega)\int_{z_0}^{z_1}\left(-\frac{e^{i(k_{gz(1)}+k_{sz(1)})(z'-z_0)}}{16k_{gz(1)}k_{sz(1)}}\right)\frac{\omega^2}{c_{0(1)}^2}\alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z') dz' \\ &= f(\omega)\int_{z_0}^{z_1} u_{p(1)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z', \omega)\frac{\omega^2}{c_{0(1)}^2}\alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z') dz',\end{aligned}$$

which is the same as equation 2.11 with $u_{p(1)}$ defined in equation 2.12.

Next, we are interested in computing the scattered portion of the wave-field contributed from the scattering potential α in the second layer denoted by D_2. The first right-hand-side term in equation A.17 propagates the wave-field (without scattering) through to the bottom of D_1, and can be used as a boundary condition at the top of D_2. Translating this boundary condition into a forcing term gives for D_2,

$$\mathcal{L}(\psi_{(2,1)}, c_{0(2)}) = f(\omega)G_{0(1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega)\delta(z - z_1) + \frac{\omega^2}{c_{0(2)}^2}\alpha(\mathbf{x}, z)\psi_{(2,1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega),$$

so that,

$$\begin{aligned}\psi_{(2,1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega) &= f(\omega)\int_{-\infty}^{\infty} G_{0(1)}(\mathbf{x}', z_1|\mathbf{x}_s, z_0; \omega)G_{0(2)}(\mathbf{x}, z|\mathbf{x}', z_1; \omega)d\mathbf{x}' \\ &+ \int_{-\infty}^{\infty}\int_{z_1}^{z_2} G_{0(2)}(\mathbf{x}, z|\mathbf{x}', z'; \omega)\frac{\omega^2}{c_{0(2)}^2}\alpha(\mathbf{x}', z')\psi_{(2,1)}(\mathbf{x}', z'|\mathbf{x}_s, z_0; \omega)dz'd\mathbf{x}'.\end{aligned} \tag{A.20}$$

We recognize that the second term on the right-hand-side of equation A.20 contains the scattered portion of the wave-field which, here, we denote as $\psi_{s(2,1)}$. As before, recognizing the recursion in equation A.20 allows us to write $\psi_{s(2,1)}$ to first order in α,

$$\begin{aligned}\psi_{s(2,1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega) &= f(\omega)\iint_{-\infty}^{\infty} G_{0(1)}(\mathbf{x}'', z_1|\mathbf{x}_s, z_0; \omega) \\ &\times \int_{z_1}^{z_2} G_{0(2)}(\mathbf{x}, z|\mathbf{x}', z'; \omega)\frac{\omega^2}{c_{0(2)}^2}\alpha(\mathbf{x}', z')G_{0(2)}(\mathbf{x}', z'|\mathbf{x}'', z_1; \omega)dz'd\mathbf{x}''d\mathbf{x}'.\end{aligned} \tag{A.21}$$

Equation A.21 is the scattered wave-field in D_2 due to the scattering potential $\alpha \in D_2$. To find the scattered wave-field at the measurement surface due to $\alpha \in D_2$, we use $\psi_{s(2,1)}$ as a boundary condition at the bottom surface of D_1. Then, we let $\psi_{s(2)}$ be the scattered wave-field in D_1 due to $\alpha \in D_2$. In particular, we have,

$$\mathcal{L}(\psi_{s(2)}, c_{0(1)}) = \psi_{s(2,1)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega)\delta(z - z_1),$$

so that for $z \in D_1$,

$$\begin{aligned}\psi_{s(2)}(\mathbf{x}, z|\mathbf{x}_s, z_0;\omega) &= f(\omega)\iiint_{-\infty}^{\infty} G_{0(1)}(\mathbf{x}, z|\mathbf{x}', z_1; \omega)\int_{z_1}^{z_2} G_{0(2)}(\mathbf{x}', z_1|\mathbf{x}'', z''; \omega) \\ &\times\frac{\omega^2}{c_{0(2)}^2}\alpha(\mathbf{x}'', z'')G_{0(2)}(\mathbf{x}'', z''|\mathbf{x}''', z_1; \omega)dz''G_{0(1)}(\mathbf{x}''', z_1|\mathbf{x}_s, z_0; \omega)d\mathbf{x}'''d\mathbf{x}''d\mathbf{x}',\end{aligned}$$

which is equivalent to equation 2.16 for $l = 2$. To show this we, as before, evaluate it at the measurement surface $(\mathbf{x}_g, z_0)$, and take the Fourier transforms over $\mathbf{x}_g$ and $\mathbf{x}_s$ so that,

$$\psi_{s(2)}(\mathbf{k}_{gx}, z_0|\mathbf{k}_{sx}, z_0; \omega) = f(\omega) \iiint_{-\infty}^{\infty} G_{0(1)}(\mathbf{k}_{gx}, z_0|\mathbf{x}', z_1; \omega) \int_{z_1}^{z_2} G_{0(2)}(\mathbf{x}', z_1|\mathbf{x}'', z''; \omega) \\ \times \frac{\omega^2}{c_{0(2)}^2} \alpha(\mathbf{x}'', z'') G_{0(2)}(\mathbf{x}'', z''|\mathbf{x}''', z_1; \omega) dz'' G_{0(1)}(\mathbf{x}''', z_1|\mathbf{k}_{sx}, z_0; \omega) d\mathbf{x}''' d\mathbf{x}'' d\mathbf{x}'.$$

Next, we substitute the Green's functions in equations 2.6 and 2.7, so that,

$$\psi_{s(2)}(\mathbf{k}_{gx}, z_0|\mathbf{k}_{sx}, z_0; \omega) = f(\omega) \iiint_{-\infty}^{\infty} \left(-\frac{e^{ik_{gz(1)}(z_1-z_0)}}{i4k_{gz(1)}} e^{-i\mathbf{k}_{gx}\cdot\mathbf{x}'} \right) \\ \times \int_{z_1}^{z_2} G_{0(2)}(\mathbf{x}', z_1|\mathbf{x}'', z''; \omega) \frac{\omega^2}{c_{0(2)}^2} \alpha(\mathbf{x}'', \mathbf{z}'') G_{0(2)}(\mathbf{x}'', z''|\mathbf{x}''', z_1; \omega) dz'' \\ \times \left(-\frac{e^{ik_{sz}(1)(z_1-z_0)}}{i4k_{sz(1)}} e^{i\mathbf{k}_{sx}\cdot\mathbf{x}'''} \right) d\mathbf{x}''' d\mathbf{x}' d\mathbf{x}'',$$

so that recognizing Fourier transforms over $\mathbf{x}'$ and $\mathbf{x}'''$ gives,

$$\psi_{s(2)}(\mathbf{k}_{gx}, z_0|\mathbf{k}_{sx}, z_0;\omega) = f(\omega) \int_{-\infty}^{\infty} \left(-\frac{e^{ik_{gz(1)}(z_1-z_0)}}{i4k_{gz(1)}} \right) \int_{z_1}^{z_2} G_{0(2)}(\mathbf{k}_{gx}, z_1|\mathbf{x}'', z''; \omega) \\ \times \frac{\omega^2}{c_{0(2)}^2} \alpha(\mathbf{x}'', \mathbf{z}'') G_{0(2)}(\mathbf{x}'', z''|\mathbf{k}_{sx}, z_1; \omega) dz'' \left(-\frac{e^{ik_{sz(1)}(z_1-z_0)}}{i4k_{sz(1)}} \right) d\mathbf{x}''. \tag{A.22}$$

Finally, we recognize that the remaining integrals are the D_2 analogy to those in equation A.18 so that,

$$\psi_{s(2)}(\mathbf{k}_{gx}, z_0|\mathbf{k}_{sx}, z_0; \omega) \\ = f(\omega) u_{p(1)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z_1, \omega) \int_{z_1}^{z_2} u_{p(2)}(\mathbf{k}_{gx}, \mathbf{k}_{sx}, z', \omega) \frac{\omega^2}{c_{0(2)}^2} \alpha(\mathbf{k}_{gx} - \mathbf{k}_{sx}, z') dz'. \tag{A.23}$$

The generalization to the l^{th} layer follows from the pattern of integrals that we recognized in deriving equation A.23 from equation A.22, so that $\psi_{s(l)}$ is given by equation 2.16.

A.4 Derivation of the WKBJ Green's function

We roughly follow the derivation in Ghatak et al. (1991) to derive a WKBJ Green's function that can be substituted into the Born approximation of the wave-field (as is done in Clayton and Stolt (1981)), or used in conjunction with the split-step approximation (as is done in

Section 2.5.2).

We start with the homogeneous Helmholtz equation $\mathcal{L}(\psi, c_0(z)) = 0$ where the operator $\mathcal{L}$ is defined by equation 2.1, and take the two dimensional Fourier transform over the lateral dimensions $\mathbf{x}$ so that,

$$\left(\frac{\partial^2}{\partial z^2} + k_z^2(z)\right)\psi(\mathbf{k}_x, z|\mathbf{x}_s, z_s; \omega) = 0, \tag{A.24}$$

where,

$$k_z^2(z) = \omega^2 c_0^{-2}(z)\left(1 - \frac{c_0^2(z)}{\omega^2}\mathbf{k}_x \cdot \mathbf{k}_x\right) = \omega^2 n^2(z), \tag{A.25}$$

and,

$$n^2(z) = 1 - \frac{c_0^2(z)}{\omega^2}\mathbf{k}_x \cdot \mathbf{k}_x.$$

Substituting equation A.25 into equation A.24, gives,

$$\left(\frac{\partial^2}{\partial z^2} + \omega^2 n^2(z)\right)\psi(\mathbf{k}_x, z|\mathbf{x}_s, z_s; \omega) = 0. \tag{A.26}$$

Next, we make the ansatz,

$$\psi(\mathbf{k}_x, z; \omega) = e^{i\omega g(z)}, \tag{A.27}$$

and proceed to find an appropriate form for $g(z)$. The second derivative of ψ is,

$$\frac{\partial^2}{\partial z^2}\psi(\mathbf{k}_x, z; \omega) = e^{i\omega g(z)}\left[(i\omega g'(z))^2 + (i\omega g''(z))\right]. \tag{A.28}$$

Upon substitution of equation A.28 into equation A.26, we find,

$$-(g'(z))^2 + ig''(z)/\omega + n^2(z) = 0, \tag{A.29}$$

where we have divided through by ω^2 in order to make the coefficient of the n^2 term unity. Next, we write $g(z)$ using a power series in ω^{-1} so that,

$$g(z) = g_0(z) + g_1(z)\omega^{-1} + g_2(z)\omega^{-2} + \cdots \quad ,$$

and substituting back into equation A.29, we find by equating terms that are zero order in ω^{-1},

$$g_0(z) = \pm\int^z n(z')dz'.$$

Likewise, using terms that are first order in ω^{-1}, we find,

$$g_1(z) = \frac{i}{2}\ln n + C,$$

so that to first order in ω^{-1},

$$g(z) = g_0(z) + g_1(z) = \pm \int^z n(z')dz' + \frac{i}{2} \ln n + C. \tag{A.30}$$

Substitution of equation A.30 into equation A.27 gives,

$$\psi(\mathbf{k}_x, z; \omega) = e^{i\omega(\pm \int^z n(z')dz' + (i/2)\ln n + C)} = \frac{C_\pm}{\sqrt{n(z)}} e^{\pm i\omega \int^z n(z')dz'}.$$

To find $C_\pm$, we interpret an initial condition from the forcing term in the definition of the Green's function so that at $t = 0$ the wave-field is equal to the constant velocity Green's function with $c(\mathbf{x}, z) = c(\mathbf{x}_s, z_s)$. In particular, we let,

$$H(z - z_s)\psi(\mathbf{k}_x, z; t = 0) = G_0(\mathbf{k}_x, z|\mathbf{x}_s, z_s; t = 0),$$

where $G_0 = 0$ for $(\mathbf{x}, z) \neq (\mathbf{x}_s, z_s)$, and H is the Heaviside step function. Hence, for the non-trivial case where $(\mathbf{x}, z) = (\mathbf{x}_s, z_s)$, and using the constant velocity and causal Green's function in equation 2.6 so that $C_+ = 0$, we find,

$$H(z - z_s) \int_{-\infty}^{\infty} e^{-i\omega t} \frac{C_-}{\sqrt{n(z_s)}} e^{-i\omega \int^{z_s} n(z')dz'} d\omega = -\int_{-\infty}^{\infty} e^{-i\omega t} \frac{1}{i4k_z} e^{-i\mathbf{k}_x \cdot \mathbf{x}_s} e^{ik_z|z_s - z_s|} d\omega,$$

or,

$$\int_{-\infty}^{\infty} e^{-i\omega t} \frac{C_-}{\sqrt{n(z_s)}} e^{-i\omega \int_{z_s}^{z_s} n(z')dz'} d\omega = -\int_{-\infty}^{\infty} e^{-i\omega t} \frac{1}{i4k_z} e^{-i\mathbf{k}_x \cdot \mathbf{x}_s} e^{ik_z|z_s - z_s|} d\omega,$$

where k_z is evaluated at $z = z_s$. It follows that,

$$\frac{C_-}{\sqrt{n(z_s)}} = -\frac{1}{i4k_z} e^{-i\mathbf{k}_x \cdot \mathbf{x}_s} \Rightarrow C_- = -\frac{\sqrt{n(z_s)}}{i4k_z} e^{-i\mathbf{k}_x \cdot \mathbf{x}_s} = -\frac{1}{i4\omega\sqrt{n(z_s)}} e^{-i\mathbf{k}_x \cdot \mathbf{x}_s}.$$

Hence, the WKBJ Green's function is,

$$G_z(\mathbf{k}_x, z|\mathbf{x}_s, z_s; \omega) = -\frac{1}{i4\sqrt{k_z(z)k_z(z_s)}} e^{-i\mathbf{k}_x \cdot \mathbf{x}_s} e^{i\int_{z_s}^{z} k_z(z')dz'}, \tag{A.31}$$

which is equivalent to equation 2.24. Finally, to find the form of equation 2.25, we take the inverse Fourier transform of equation A.31 over lateral space $\mathbf{x}$, so that,

$$G_z(\mathbf{x}, z|\mathbf{x}_s, z_s; \omega) = \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} e^{i\mathbf{k}_x \cdot \mathbf{x}} \left[-\frac{1}{i4\sqrt{k_z(z)k_z(z_s)}} e^{-i\mathbf{k}_x \cdot \mathbf{x}_s} e^{i\int_{z_s}^{z} k_z(z')dz'} \right] d\mathbf{k}_x.$$

Making a change of variables from $\mathbf{k}_x$ to $\mathbf{k}_{sx}$ gives,

$$G_z(\mathbf{x}, z|\mathbf{x}_s, z_s; \omega) = \left(\frac{1}{2\pi}\right)^2 \int_{-\infty}^{\infty} e^{-i\mathbf{k}_{sx}\cdot\mathbf{x}_s} \left[-\frac{1}{i4\sqrt{k_z(z)k_z(z_s)}} e^{i\mathbf{k}_{sx}\cdot\mathbf{x}} e^{i\int_{z_s}^{z} k_{sz}(z')dz'}\right] d\mathbf{k}_{sx},$$

so that,

$$G_z(\mathbf{x}, z|\mathbf{k}_{sx}, z_s; \omega) = -\frac{1}{i4\sqrt{k_{sz}(z)k_{sz}(z_s)}} e^{i\mathbf{k}_{sx}\cdot\mathbf{x}} e^{i\int_{z_s}^{z} k_{sz}(z')dz'},$$

which is the same as equation 2.25.

■ APPENDIX B

Derivations for Ritz values

B.1 Derivation of ω for the Jacobian operator

We derive equation 3.29 which, in turn, is used in the derivation of the Jacobian operator in Chapter 3. We begin with the equation for k_z in equation 3.27, and which, at the risk of being redundant, we repeat here,

$$k_z = \sqrt{\frac{\omega^2}{c_0^2} - \frac{1}{4}|\omega \mathbf{p}_h + \mathbf{k}_m|^2} + \sqrt{\frac{\omega^2}{c_0^2} - \frac{1}{4}|\omega \mathbf{p}_h - \mathbf{k}_m|^2}.$$

Taking the square of both sides and re-arranging terms gives,

$$\left(k_z^2 + \frac{|\mathbf{k}_m|^2}{2}\right) + \omega^2\left(\frac{|\mathbf{p}_h|^2}{2} - \frac{2}{c_0^2}\right) = 2\sqrt{\frac{\omega^2}{c_0^2} - \frac{1}{4}|\omega \mathbf{p}_h + \mathbf{k}_m|^2}\sqrt{\frac{\omega^2}{c_0^2} - \frac{1}{4}|\omega \mathbf{p}_h - \mathbf{k}_m|^2}.$$

Again, we take the square of both sides and group terms according to powers of ω finding,

$$\begin{aligned}\left(k_z^2 + \frac{|\mathbf{k}_m|^2}{2}\right)^2 + \omega^4\left(\frac{4}{c_0^2} + \frac{1}{4}(|\mathbf{p}_h|^2)^2 - 2\frac{|\mathbf{p}_h|^2}{c_0^2}\right) + 2\omega^2\left(\frac{|\mathbf{p}_h|^2}{2} - \frac{2}{c_0^2}\right)\left(k_z^2 + \frac{|\mathbf{k}_m|^2}{2}\right)& \\ = \omega^4\left(\frac{4}{c_0^4} + \frac{1}{4}(|\mathbf{p}_h|^2)^2 - 2\frac{|\mathbf{p}_h|^2}{c_0^2}\right) - \omega^2\left(\frac{1}{2}|\mathbf{p}_h|^2|\mathbf{k}_m|^2 + 2\frac{|\mathbf{k}_m|^2}{c_0^2}\right) + \frac{1}{4}(|\mathbf{k}_m|^2)^2.&\end{aligned} \tag{B.1}$$

In equation B.1 we notice that the ω^4 terms cancel giving,

$$\begin{aligned}\left(k_z^2 + \frac{|\mathbf{k}_m|^2}{2}\right)^2 + 2\omega^2\left(\frac{|\mathbf{p}_h|^2}{2} - \frac{2}{c_0^2}\right)\left(k_z^2 + \frac{|\mathbf{k}_m|^2}{2}\right)& \\ = -\omega^2\left(\frac{1}{2}|\mathbf{p}_h|^2|\mathbf{k}_m|^2 + 2\frac{|\mathbf{k}_m|^2}{c_0^2}\right) + \frac{1}{4}(|\mathbf{k}_m|^2)^2.&\end{aligned}$$

Finally, grouping terms in ω^2 gives,

$$\left(k_z^2 + \frac{|\mathbf{k}_m|^2}{2}\right)^2 - \frac{1}{4}(|\mathbf{k}_m|^2)^2$$
$$= \omega^2 \left[-\frac{1}{2}|\mathbf{p}_h|^2|\mathbf{k}_m|^2 - 2\frac{|\mathbf{k}_m|^2}{c_0^2} - 2\left(\frac{|\mathbf{p}_h|^2}{2} - \frac{2}{c_0^2}\right)\left(k_z^2 + \frac{|\mathbf{k}_m|^2}{2}\right)\right],$$

which, in turn, is equivalent to,

$$k_z^2(k_z^2 + |\mathbf{k}_m|^2) = \omega^2 \left[-|\mathbf{p}_h|^2|\mathbf{k}_m|^2 - |\mathbf{p}_h|^2 k_z^2 + 4\frac{k_z^2}{c_0^2}\right]. \tag{B.2}$$

Solving equation B.2 for ω such that $\omega \geq 0$ gives the desired result in equation 3.29.

B.2 Derivation of the pre-conditioned conjugate gradient method

We derive the pre-conditioned conjugate gradient method in Algorithm 3.2 using the conjugate gradient method in Algorithm 3.1 applied to the pre-conditioned normal equations (3.31). In particular, the conjugate gradient method used to solve equation 3.31 gives Algorithm B.1 where we have used $\mathbf{E}^{-1}\mathbf{A}^H\mathbf{A}\mathbf{E}$ in place of $\mathbf{A}^H\mathbf{A}$, $\mathbf{E}^{-1}\mathbf{A}^H\mathbf{d}$ in place of $\mathbf{A}^H\mathbf{d}$, and $\hat{\mathbf{m}}$ in place of $\mathbf{m}$ with $\mathbf{m} = \mathbf{E}^{-H}\hat{\mathbf{m}}$ (see equation 3.31). In addition, in Algorithm B.1, we have introduced new search directions $\hat{\mathbf{p}}_k$ and residual vectors $\hat{\mathbf{r}}_k$.

Algorithm B.1 Derivation of the pre-conditioned conjugate gradient method, step 1

```
$\hat{\mathbf{m}}_0 = \mathbf{0}$, $\hat{\mathbf{r}}_0 = \mathbf{E}^{-1}\mathbf{A}^H\mathbf{d}$, $\hat{\mathbf{p}}_0 = \hat{\mathbf{r}}_0$
for $k = 1$ to $n$ do
    $\alpha_k = ||\hat{\mathbf{r}}_k||_2^2/(\hat{\mathbf{p}}_{k-1}^H\mathbf{E}^{-1}\mathbf{A}^H\mathbf{A}\mathbf{E}^{-H}\hat{\mathbf{p}}_{k-1})$
    $\hat{\mathbf{m}}_k = \hat{\mathbf{m}}_{k-1} + \alpha_k\hat{\mathbf{p}}_{k-1}$
    $\hat{\mathbf{r}}_k = \mathbf{E}^{-1}\mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{E}^{-H}\hat{\mathbf{m}}_k)$
    $\beta_k = ||\hat{\mathbf{r}}_k||_2^2/||\hat{\mathbf{r}}_{k-1}||_2^2$
    $\hat{\mathbf{p}}_k = \hat{\mathbf{r}}_k + \beta_k\hat{\mathbf{p}}_{k-1}$
end for
```

To find Algorithm 3.2, we start with Algorithm B.1, and substitute for $\hat{\mathbf{m}}_k = \mathbf{E}^H\mathbf{m}_k$ so that the residual $\hat{\mathbf{r}}_k$ becomes,

$$\hat{\mathbf{r}}_k = \mathbf{E}^{-1}\mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{E}^{-H}\mathbf{E}^H\mathbf{m}_k) = \mathbf{E}^{-1}\mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{m}_k) = \mathbf{E}^{-1}\mathbf{r}_k,$$

and where we have let $\mathbf{r}_k = \mathbf{A}^H(\mathbf{d} - \mathbf{A}\mathbf{m}_k)$. We notice that this this definition of $\mathbf{r}_k$ is the same as was used in the conjugate gradient method in Algorithm 3.1. Substitution of $\hat{\mathbf{m}}_k = \mathbf{A}^H\mathbf{m}_k$ and $\hat{\mathbf{r}}_k = \mathbf{E}^{-1}\mathbf{r}_k$ in Algorithm B.1 gives several simplifications, and results in Algorithm B.2. In going from Algorithm B.1 to Algorithm B.2, we have used $\mathbf{J}^{-1} = \mathbf{E}^{-H}\mathbf{E}^{-1}$.

Algorithm B.2 Derivation of the pre-conditioned conjugate gradient method, step 2

```
$\mathbf{m}_0 = \mathbf{0}$, $\mathbf{r}_0 = \mathbf{A}^H\mathbf{d}$, $\hat{\mathbf{p}}_0 = \mathbf{E}^{-1}\mathbf{r}_0$
for $k = 1 \dots n$ do
    $\alpha_k = (r_k^H\mathbf{J}^{-1}\mathbf{r}_k)/(\hat{\mathbf{p}}_{k-1}^H\mathbf{E}^{-1}\mathbf{A}^H\mathbf{A}\mathbf{E}^{-H}\hat{\mathbf{p}}_{k-1})$
    $\mathbf{m}_k = \mathbf{m}_{k-1} + \alpha_k\mathbf{E}^{-H}\hat{\mathbf{p}}_{k-1}$
    $\mathbf{r}_k = \mathbf{A}^H(d - \mathbf{A}\mathbf{m}_k)$
    $\beta_k = (\mathbf{r}_k^H\mathbf{J}^{-1}\mathbf{r}_k)/(\mathbf{r}_{k-1}^H\mathbf{J}^{-1}\mathbf{r}_{k-1}$
    $\hat{\mathbf{p}}_k = \mathbf{E}^{-1}\mathbf{r}_k + \beta_k\hat{\mathbf{p}}_{k-1}$
end for
```

Finally, to arrive at Algorithm 3.2, we let $\hat{\mathbf{p}}_k = \mathbf{E}^H\mathbf{p}_k$ so that in Algorithm B.2, the search directions simplify to,

$$\mathbf{E}^H\mathbf{p}_0 = \mathbf{E}^{-1}\mathbf{r}_0 \Rightarrow \mathbf{p}_0 = \mathbf{E}^{-H}\mathbf{E}^{-1}\mathbf{r}_0 = \mathbf{J}^{-1}\mathbf{r}_0, \tag{B.3}$$

for $k = 0$, and similarly the search directions for $k > 0$ become,

$$\mathbf{E}^H\mathbf{p}_k = \mathbf{E}^{-1}\mathbf{r}_k + \beta_k\mathbf{E}^H\mathbf{p}_{k-1} \Rightarrow \mathbf{p}_k = \mathbf{E}^{-H}\mathbf{E}^{-1}\mathbf{r}_k + \beta_k\mathbf{E}^{-H}\mathbf{E}^H\mathbf{p}_{k-1} = \mathbf{J}^{-1}\mathbf{r}_k + \beta_k\mathbf{p}_{k-1}. \tag{B.4}$$

Likewise the model update computing $\mathbf{m}_k$ from $\mathbf{m}_{k-1}$ becomes,

$$\mathbf{m}_k = \mathbf{m}_{k-1} + \alpha_k\mathbf{E}^{-H}\hat{\mathbf{p}}_{k-1} = \mathbf{m}_{k-1} + \alpha_k\mathbf{E}^{-H}\mathbf{E}^H\mathbf{p}_{k-1} = \mathbf{m}_{k-1} + \alpha_k\mathbf{p}_{k-1}. \tag{B.5}$$

Applying equations B.3-B.5 to Algorithm B.2, and approximating $\mathbf{J}^{-1}$ by $\mathbf{J}^\dagger$ gives the desired result in Algorithm 3.2.

■ APPENDIX C

Derivations for regularized migration with sparseness constraints

To derive equation 4.2, we start with the forward operator in equation 2.4, Fourier transform over $\mathbf{x}_g$ and $\mathbf{x}_s$, and substitute for the constant velocity Green's functions in equations 2.6 and 2.7. Finally, we let the support of α be below both z_s and z_g, giving,

$$\psi_s(\mathbf{k}_{gx}, z_g|\mathbf{k}_{sx}, z_s; \omega) \propto \iint e^{-i(\mathbf{k}_{gx}-\mathbf{k}_{sx})\cdot\mathbf{x}'} e^{ik_{gz}(z'-z_g)} \alpha(\mathbf{x}', z') e^{ik_{sz}(z'-z_s)} d\mathbf{x}' dz'. \tag{C.1}$$

In equation C.1, the relation is proportional due to omitted amplitude terms. Recognizing the two dimensional Fourier transform over $\mathbf{x}'$, we find,

$$\psi_s(\mathbf{k}_{gx}, z_g|\mathbf{k}_{sx}, z_s; \omega) \propto \int e^{ik_{gz}(z'-z_g)} \alpha(\mathbf{k}_{gx}-\mathbf{k}_{sx}, z') e^{ik_{sz}(z'-z_s)} d\mathbf{x}' dz', \tag{C.2}$$

and whose adjoint is,

$$\alpha^\dagger(\mathbf{k}_{gx}-\mathbf{k}_{sx}, z') = \int e^{-ik_{gz}(z'-z_g)} \psi_s(\mathbf{k}_{gx}, z_g|\mathbf{k}_{sx}, z_s; \omega) e^{-ik_{sz}(z'-z_s)} d\omega. \tag{C.3}$$

The dispersion relation in equations 2.8, gives for $\omega > 0$ (see Figure 2.2),

$$k_{gz}(z'-z_g) = \omega\frac{z'-z_g}{c_0}\sqrt{1-\frac{c_0^2}{\omega^2}\mathbf{k}_{gx}\cdot\mathbf{k}_{gx}} = \omega\frac{z'-z_g}{c_0}\sqrt{1-\sin^2\theta_g} = \omega\frac{z'-z_g}{c_0}\cos\theta_g. \tag{C.4}$$

Letting, $\tau_{gz} = (z'-z_g)/c_0$, and making a similar argument for τ_{sz} gives equation 4.2.

APPENDIX D

Derivations for shot-profile wave-field modelling

We derive $\psi_{s(1)}$ in equation 5.4 and $\psi_{s(2)}$ in equation 5.9. First, we consider $\psi_{s(1)}$. We substitute the Green's function in equation 5.1 into the Born approximation in equation 2.4, taking the support of α below the measurement surface and within D_1,

$$\begin{aligned}\psi_{s(1)}(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega) = &\left(\frac{\omega}{c_{1(1)}}\right)^2 \left(\frac{1}{2\pi}\right)^4 \int_{z_0}^{z_1} \int_{-\infty}^{\infty} \\ &\times e^{i\omega(c_{0(1)}^{-1}(\mathbf{x}_g) - c_{1(1)}^{-1})(z'-z_0)} \int_{-\infty}^{\infty} \left(-\frac{1}{i4k_{gz(1)}}\right) e^{-i\mathbf{k}_{gx}\cdot(\mathbf{x}'-\mathbf{x}_g)} e^{ik_{gz(1)}(z'-z_0)} d\mathbf{k}_{gx} \alpha(\mathbf{x}', z') \\ &\times e^{i\omega(c_{0(1)}^{-1}(\mathbf{x}') - c_{1(1)}^{-1})(z'-z_0)} \int_{-\infty}^{\infty} \left(-\frac{1}{i4k'_{z(1)}}\right) e^{i\mathbf{k}'_x\cdot\mathbf{x}'} e^{ik'_{z(1)}(z'-z_0)} g(\mathbf{k}'_x, \mathbf{x}_s, \omega) d\mathbf{k}'_x d\mathbf{x}' dz'.\end{aligned} \tag{D.1}$$

In equation D.1, we have recognized an expression for the synthetic source used in shot-profile migration algorithms,

$$g(\mathbf{k}'_x, \mathbf{x}_s, \omega) = f(\omega) e^{-i\mathbf{k}'_x\cdot\mathbf{x}_s}.$$

Rearranging terms, and regrouping integrals gives,

$$\begin{aligned}\psi_{s(1)}(\mathbf{x}_g, z_0|\mathbf{x}_s, z_0; \omega) = &\left(\frac{\omega}{c_{1(1)}}\right)^2 \left(\frac{1}{2\pi}\right)^4 \int_{z_0}^{z_1} e^{i\omega(c_{0(1)}^{-1}(\mathbf{x}_g) - c_{1(1)}^{-1})(z'-z_0)} \int_{-\infty}^{\infty} e^{-i\mathbf{k}_{gx}\cdot\mathbf{x}_g} \\ &\times \left(-\frac{1}{i4k_{gz(1)}}\right) e^{ik_{gz(1)}(z'-z_0)} \int_{-\infty}^{\infty} e^{-i\mathbf{k}_{gx}\cdot\mathbf{x}'} e^{i\omega(c_{0(1)}^{-1}(\mathbf{x}') - c_{1(1)}^{-1})(z'-z_0)} \\ &\times \int_{-\infty}^{\infty} e^{i\mathbf{k}'_x\cdot\mathbf{x}'} \left(-\frac{1}{i4k'_{z(1)}}\right) e^{ik'_{z(1)}(z'-z_0)} g(\mathbf{k}'_x, \mathbf{x}_s, \omega) d\mathbf{k}'_x \alpha(\mathbf{x}', z') d\mathbf{x}' d\mathbf{k}_{gx} dz'.\end{aligned} \tag{D.2}$$

Finally, we make change of variables from $\mathbf{k}'_x$ to $\mathbf{k}_{gx}$, and $\mathbf{x}'$ to $\mathbf{x}_g$, and recognize three two-dimensional Fourier transforms to give equation 5.4. Note that the change of integration variables to lateral geophone (space and wave-number) is somewhat arbitrary. In practise, we extend $\mathbf{x}_g$ beyond the aperture of the geophones by adding zero-traces to either side of the shot gather. This, in effect, makes the aperture of α for shot $\mathbf{x}_s$ (the migrated image shot gather) independent of the recording aperture.

Next, we consider $\psi_{s(2)}$, the contribution to the scattered wave-field from the second layer denoted by D_2. In equation 5.4, $\psi_{s(1)}$ is the scattered portion of the first order solution to the perturbed Helmholtz equation,

$$\mathcal{L}(\psi_{(1)}, c_{0(1)}) = f(\omega)\delta(\mathbf{x}-\mathbf{x}_s)\delta(z-z_0) + \frac{\omega^2}{c_{0(1)}^2}\alpha(\mathbf{x},z)\psi_{(1)}(\mathbf{x},z|\mathbf{x}_s,z_0;\omega), \tag{D.3}$$

where $\mathcal{L}$ is the Helmholtz operator. The corresponding direct wave-field $\psi_{d(1)} = f(\omega)G_{0(1)}$, propagates the wave-field (without scattering) from the source through to the bottom of D_1, and can be used as a boundary condition at the top of D_2. Translating this boundary condition into a forcing term gives for D_2,

$$\mathcal{L}(\psi_{(2,1)}, c_{0(2)}) = f(\omega)G_{0(1)}(\mathbf{x},z|\mathbf{x}_s,z_0;\omega)\delta(z-z_1) + \frac{\omega^2}{c_{0(2)}^2}\alpha(\mathbf{x},z)\psi_{(2,1)}(\mathbf{x},z|\mathbf{x}_s,z_0;\omega), \tag{D.4}$$

so that,

$$\begin{aligned}\psi_{(2,1)}(\mathbf{x},z|\mathbf{x}_s,z_0;\omega) = f(\omega)\int_{-\infty}^{\infty} G_{0(1)}(\mathbf{x}',z_1|\mathbf{x}_s,z_0;\omega)G_{0(2)}(\mathbf{x},z|\mathbf{x}',z_1;\omega)d\mathbf{x}' \\ + \int_{-\infty}^{\infty}\int_{z_1}^{z_2} G_{0(2)}(\mathbf{x},z|\mathbf{x}',z';\omega)\frac{\omega^2}{c_{0(2)}^2}\alpha(\mathbf{x}',z')\psi_{(2,1)}(\mathbf{x}',z'|\mathbf{x}_s,z_0;\omega)dz'd\mathbf{x}'.\end{aligned} \tag{D.5}$$

We recognize that the second term on the right-hand-side of equation D.5 contains the scattered portion of the wave-field which, here, we denote as $\psi_{s(2,1)}$. Recognizing the recursion in equation D.5 allows us to write $\psi_{s(2,1)}$ to first order in α,

$$\begin{aligned}\psi_{s(2,1)}(\mathbf{x},z|\mathbf{x}_s,z_0;\omega) \approx f(\omega)\iint_{-\infty}^{\infty} G_{0(1)}(\mathbf{x}'',z_1|\mathbf{x}_s,z_0;\omega) \\ \times \int_{z_1}^{z_2} G_{0(2)}(\mathbf{x},z|\mathbf{x}',z';\omega)\frac{\omega^2}{c_{0(2)}^2}\alpha(\mathbf{x}',z')G_{0(2)}(\mathbf{x}',z'|\mathbf{x}'',z_1;\omega)dz'd\mathbf{x}''d\mathbf{x}'.\end{aligned} \tag{D.6}$$

Equation D.6 is the scattered wave-field in D_2 due to the scattering potential $\alpha \in D_2$. To find the scattered wave-field at the measurement surface due to $\alpha \in D_2$, we use $\psi_{s(2,1)}$ as a boundary condition at the bottom surface of D_1 (neglecting reflected energy). Then, we let $\psi_{s(2)}$ be the scattered wave-field in D_1 due to $\alpha \in D_2$. In particular, we have,

$$\mathcal{L}(\psi_{s(2)}, c_{0(1)}) = \psi_{s(2,1)}(\mathbf{x},z|\mathbf{x}_s,z_0;\omega)\delta(z-z_1), \tag{D.7}$$

so that for $z \in D_1$,

$$\psi_{s(2)}(\mathbf{x}, z|\mathbf{x}_s, z_0; \omega) = f(\omega) \iiint_{-\infty}^{\infty} G_{0(1)}(\mathbf{x}, z|\mathbf{x}', z_1; \omega) \int_{z_1}^{z_2} G_{0(2)}(\mathbf{x}', z_1|\mathbf{x}'', z''; \omega) \\ \times \frac{\omega^2}{c_{0(2)}^2} \alpha(\mathbf{x}'', z'') G_{0(2)}(\mathbf{x}'', z''|\mathbf{x}''', z_1; \omega) dz'' G_{0(1)}(\mathbf{x}''', z_1|\mathbf{x}_s, z_0; \omega) d\mathbf{x}''' d\mathbf{x}'' d\mathbf{x}', \tag{D.8}$$

which can be made equivalent to equation 5.10 for $l = 2$. To show this, we evaluate it at the measurement surface $(\mathbf{x}_g, z_0)$, and substitute for the Green's function in equation 5.1 so that after re-grouping integrals, we find,

$$\psi_{s(2)}(\mathbf{x}_g, z_g|\mathbf{x}_s, z_s; \omega) = \left(\frac{1}{2\pi}\right)^8 \frac{\omega^2}{c_{1(2)}^2} \int_{z_1}^{z_2} u_{s(1)}(\mathbf{x}_g, z_1; \omega) \int e^{i\mathbf{k}_{gx}\cdot\mathbf{x}_g} u_{p(1)}(\mathbf{k}_{gx}, z_1; \omega) \\ \times \int e^{-i\mathbf{k}_{gx}\cdot\mathbf{x}'} u_{s(2)}(\mathbf{x}', z''; \omega) \int e^{i\mathbf{k}'_x\cdot\mathbf{x}'} u_{p(2)}(\mathbf{k}'_x, z''; \omega) \int e^{-i\mathbf{k}'_x\cdot\mathbf{x}''} \Big[u_{s(2)}(\mathbf{x}_g, z''; \omega) \\ \times \int e^{i\mathbf{k}''_x\cdot\mathbf{x}''} u_{p(2)}(\mathbf{k}''_x, z''; \omega) \int e^{-i\mathbf{k}''_x\cdot\mathbf{x}'''} u_{s(1)}(\mathbf{x}''', z_1; \omega) \int e^{i\mathbf{k}'''_x\cdot\mathbf{x}'''} u_{p(1)}(\mathbf{k}'''_x, z_1; \omega) \\ \times g(\mathbf{k}'''_x, \mathbf{x}_s, \omega) d\mathbf{k}'''_x d\mathbf{x}''' d\mathbf{k}''_x \Big] \alpha(\mathbf{x}'', z'') d\mathbf{x}'' d\mathbf{k}'_x d\mathbf{x}' d\mathbf{k}_{gx} dz'', \tag{D.9}$$

where we have, again, used the simplifying assumption in equation 5.8. Additionally, to avoid notational clutter, we have used the definitions in equations 5.5 and 5.6, and where the limits of integration are not specified, they are $(-\infty, \infty)$. In equation D.9, we make a change of integration variables from each of $\mathbf{x}'$, $\mathbf{x}''$ and $\mathbf{x}'''$ to $\mathbf{x}_g$, and from each of $\mathbf{k}'_x$, $\mathbf{k}''_x$ and $\mathbf{k}'''_x$ to $\mathbf{k}_{gx}$. Then, recognizing four two-dimensional inverse Fourier transforms, and three two-dimensional Fourier transforms gives equation 5.10 for $l = 2$.

■ APPENDIX E

Derivations for shot-profile migration data reconstruction

E.1 Derivation of the forward operator

We find ψ_s in equation 6.1 using the Born approximation in equation 2.4, and the Green's function in equation 5.1. This derivation risks being redundant to an analogous derivation in Appendix E.1, the difference being in its use of a constant reference wave-speed. Taking $z_g = z_s = z_0$, and the support of α to be below z_0, we find by substitution of equation 5.1 into equation 2.4,

$$\psi_s(\mathbf{x}_g,\omega;\mathbf{x}_s) = \left(\frac{1}{2\pi}\right)^4 \left(\frac{\omega}{c_0}\right)^2 \int_{z_0}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \left(-\frac{1}{i4k_{gz}}\right) e^{-i\mathbf{k}_{gx}\cdot(\mathbf{x}'-\mathbf{x}_g)} e^{ik_{gz}(z'-z_0)} d\mathbf{k}_{gx} \\ \times \alpha(\mathbf{x}',z') \int_{-\infty}^{\infty} \left(-\frac{1}{i4k'_z}\right) e^{i\mathbf{k}'_x\cdot\mathbf{x}'} e^{ik'_z(z'-z_0)} g(\mathbf{k}'_x,\omega,\mathbf{x}_s) d\mathbf{k}'_x d\mathbf{x}' dz'. \tag{E.1}$$

In equation E.1, $\mathbf{k}'_x$ is the Fourier conjugate variable of $\mathbf{x}'$, and we have recognized an expression for the synthetic source used in shot-profile migration algorithms,

$$g(\mathbf{k}'_x,\omega,\mathbf{x}_s) = f(\omega)e^{-i\mathbf{k}'_x\cdot\mathbf{x}_s}.$$

Moreover, we have used the dispersion relation,

$$k'_z = sgn(\omega)\sqrt{\frac{\omega^2}{c_0^2} - \mathbf{k}'_x\cdot\mathbf{k}'_x}.$$

Rearranging terms and regrouping integrals in equation E.1 gives,

$$\psi_s(\mathbf{x}_g,\omega;\mathbf{x}_s)=\left(\frac{1}{2\pi}\right)^4\left(\frac{\omega}{c_0}\right)^2\int_{z_0}^{\infty}\int_{-\infty}^{\infty}e^{i\mathbf{k}_{gx}\cdot\mathbf{x}_g}\left(-\frac{1}{i4k_{gz}}\right)e^{ik_{gz}(z'-z_0)}\int_{-\infty}^{\infty}e^{-i\mathbf{k}_{gx}\cdot\mathbf{x}'}$$
$$\times\int_{-\infty}^{\infty}e^{i\mathbf{k}'_x\cdot\mathbf{x}'}\left(-\frac{1}{i4k'_z}\right)e^{ik'_z(z'-z_0)}g(\mathbf{k}'_x,\omega,\mathbf{x}_s)d\mathbf{k}'_x\alpha(\mathbf{x}',z')d\mathbf{x}'d\mathbf{k}_{gx}dz'.$$

Finally, we make a one-to-one change of notation for variables of integration, from $\mathbf{k}'_x$ to $\mathbf{k}_{gx}$, and $\mathbf{x}'$ to $\mathbf{x}_g$, and recognize three two dimensional Fourier transforms to give equation 6.1. The conventions that we use for the Fourier kernels are indicated by the operator notation that we use in equation 6.1. Note that, as was the case in Appendix D, the change of integration variables to lateral geophone (space and wave-number) is somewhat arbitrary. Again, we extend $\mathbf{x}_g$ beyond the aperture of the geophones by adding zero-traces to either side of the shot gather. Moreover, we can let x_g correspond to the nominal grid used for data reconstruction, rather than a grid dictated by the geophone locations. This, in effect, makes the aperture of α for shot $\mathbf{x}_s$ (the migrated image shot gather) independent of the recording aperture. This was important in our discussion of least-squares shot-profile migration, and becomes even more important in our discussion of SPDR and SPDR2.

E.2 Derivation of adjoint sampling operator for SPDR and SPDR2

We find $\alpha_p^\dagger$ in equations 6.19 and 6.39. We substitute equation 6.18 into equation 6.38 while letting $z_0=0$, so that,

$$\alpha_p^\dagger(x_g,z',\omega)=\left(\frac{1}{2\pi}\right)^2\left(\frac{\omega}{c_0}\right)^2\sum_{q=1}^{n_s}\left[\int_{-\infty}^{\infty}e^{ik_{gx}\cdot x_g}e^{-ik_{gz}z'}f^*(\omega)e^{ik_{gx}x_{s(q)}}dk_{gx}\right]$$
$$\times\int_{-\infty}^{\infty}e^{ik_{gx}x_g}e^{-ik_{gz}z'}\frac{1}{\Delta x_g^{o(q)}}\delta(k_{gx}-k_p)\psi_0(\omega)dk_{gx},\tag{E.2}$$

where

$$k_p=2\pi p/\Delta x_g^o.$$

In equation E.2, we use the sifting property of the Dirac delta function so that,

$$\alpha_p^\dagger(x_g, z', \omega) = \left(\frac{1}{2\pi}\right)^2 \left(\frac{\omega}{c_0}\right)^2 \sum_{q=1}^{n_s} \left[\int_{-\infty}^{\infty} e^{ik_{gx}\cdot x_g} e^{-ik_{gz}z'} f^*(\omega) e^{ik_{gx}x_{s(q)}} dk_{gx}\right] \times e^{ik_p x_g} e^{-ik_{gz(p)}z'} \frac{1}{\Delta x_g^o} \psi_0(\omega), \tag{E.3}$$

where,

$$k_{gz(p)} = sgn(\omega)\sqrt{\frac{\omega^2}{c_0^2} - k_p^2}.$$

Making a change of variables from k_{gx} to k'_{gx}, letting, $k'_{gz} = sgn(\omega)\sqrt{\omega^2/c_0^2 - (k'_{gx})^2}$, and taking the Fourier transform of equation E.3 with respect to x_g, gives,

$$\alpha_p^\dagger(k_{gx}, z', \omega) = \frac{f^*(\omega)\psi_0(\omega)}{(2\pi)^2 \Delta x_g^o} \left(\frac{\omega}{c_0}\right)^2 \times \sum_{q=1}^{n_s} e^{-ik_{gz(p)}z'} \int_{-\infty}^{\infty} e^{-ik_{gx}x_g} \int_{-\infty}^{\infty} e^{ik'_{gx}x_g} e^{ik_p x_g} e^{ik'_{gx}x_{s(q)}} e^{-ik'_{gz}z'} dk'_{gx} dx_g.$$

We change the order of integration, and rearrange terms to give,

$$\alpha_p^\dagger(k_{gx}, z', \omega) = \frac{f^*(\omega)\psi_0(\omega)}{(2\pi)^2 \Delta x_g^o} \left(\frac{\omega}{c_0}\right)^2 \times \sum_{q=1}^{n_s} e^{-ik_{gz(p)}z'} \int_{-\infty}^{\infty} e^{ik'_{gx}x_{s(q)}} e^{-ik'_{gz}z'} \int_{-\infty}^{\infty} e^{-ik_{gx}x_g} e^{i(k'_{gx}+k_p)x_g} dx_g dk'_{gx}. \tag{E.4}$$

To proceed, we recognize that,

$$\begin{aligned}\mathcal{F}^{-1}\left\{\delta(k_{gx} - (k'_{gx} + k_p))\right\} &= \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{ik_{gx}x_g} \delta(k_{gx} - (k'_{gx} + k_p)) dk_{gx} \\ &= \frac{1}{2\pi} e^{i(k'_{gx}+k_p)x_g}.\end{aligned}$$

Hence,

$$\mathcal{F}\left\{e^{i(k'_{gx}+k_p)x_g}\right\} = 2\pi\delta(k_{gx} - (k'_{gx} + k_p)). \tag{E.5}$$

We substitute equation E.5 into equation E.4 giving,

$$\alpha_p^\dagger(k_{gx}, z', \omega) = \frac{f^*(\omega)\psi_0(\omega)}{2\pi\Delta x_g^o} \left(\frac{\omega}{c_0}\right)^2 \sum_{q=1}^{n_s} e^{-ik_{gz(p)}z'} \int_{-\infty}^{\infty} e^{ik'_{gx}x_{s(q)}} e^{-ik'_{gz}z'} \delta(k_{gx}-(k'_{gx}+k_p)) dk'_{gx}. \tag{E.6}$$

As before, we use the sifting property of the Dirac delta function so that the integral in equation E.6 evaluates to where $k'_{gx} = k_{gx} - k_p$ in the integrand. In particular, if we recall that,

$$k'_{gz} = sgn(\omega)\sqrt{\frac{\omega^2}{c_0^2} - k'_{gx}},$$

then, equation E.6 becomes,

$$\alpha_p^\dagger(k_{gx}, z', \omega) = \frac{f^*(\omega)\psi_0(\omega)}{2\pi\Delta x_g^o}\left(\frac{\omega}{c_0}\right)^2 \sum_{q=1}^{n_s} e^{-ik_{gz(p)}z'} e^{i(k_{gx}-k_p)x_{s(q)}} e^{-i\,sgn(\omega)\sqrt{\omega^2/c_0^2-(k_{gx}-k_p)^2}z'}. \tag{E.7}$$

Next, we take the Fourier transform of equation E.7 with respect to z giving,

$$\begin{aligned}\alpha_p^\dagger(k_{gx}, \kappa_z, \omega) =& \frac{f^*(\omega)\psi_0(\omega)}{2\pi\Delta x_g^o}\left(\frac{\omega}{c_0}\right)^2 \\ &\times \sum_{q=1}^{n_s} e^{i(k_{gx}-k_p)x_{s(q)}} \int_{-\infty}^{\infty} e^{-i\kappa_z z'} e^{-i(k_{gz(p)}+sgn(\omega)\sqrt{\omega^2/c_0^2-(k_{gx}-k_p)^2})z'} dz',\end{aligned} \tag{E.8}$$

where κ_z is the Fourier conjugate variable of z. To simplify equation E.8, we let $\mathcal{F}_z$ denote the Fourier transform over z, and note that,

$$\begin{aligned}\mathcal{F}_z^{-1}&\left\{\delta(\kappa_z + (k_{gz(p)} + sgn(\omega)\sqrt{\omega^2/c_0^2 - (k_{gx}-k_p)^2}))\right\} \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{i\kappa_z z}\delta(\kappa_z + (k_{gz(p)} + sgn(\omega)\sqrt{\omega^2/c_0^2 - (k_{gx}-k_p)^2}))d\kappa_z,\end{aligned}$$

which simplifies to,

$$\begin{aligned}\mathcal{F}_z^{-1}&\left\{\delta(\kappa_z + (k_{gz(p)} + sgn(\omega)\sqrt{\omega^2/c_0^2 - (k_{gx}-k_p)^2}))\right\} \\ &= \frac{1}{2\pi}e^{-i(k_{gz(p)}+sgn(\omega)\sqrt{\omega^2/c_0^2-(k_{gx}-k_p)^2})z}.\end{aligned}$$

Hence,

$$\begin{aligned}\mathcal{F}_z&\left\{e^{-i(k_{gz(p)}+sgn(\omega)\sqrt{\omega^2/c_0^2-(k_{gx}-k_p)^2})z}\right\} \\ &= 2\pi\delta(\kappa_z + (k_{gz(p)} + sgn(\omega)\sqrt{\omega^2/c_0^2 - (k_{gx}-k_p)^2})).\end{aligned} \tag{E.9}$$

With equation E.9, we can simplify equation E.8, giving,

$$
\begin{aligned}
&\alpha_p^\dagger(k_{gx},\kappa_z,\omega) \\
&\quad = \frac{f^*(\omega)\psi_0(\omega)}{\Delta x_g^o}\left(\frac{\omega}{c_0}\right)^2 \delta(\kappa_z + (k_{gz(p)} + sgn(\omega)\sqrt{\omega^2/c_0^2 - (k_{gx}-k_p)^2}))\sum_{q=1}^{n_s} e^{i(k_{gx}-k_p)x_{s(q)}},
\end{aligned}
\tag{E.10}
$$

which is the same as equation 6.39. If we let $n_s = 1$ and $x_{s(1)} = 0$, then equation E.10 simplifies to equation 6.19.

■ APPENDIX F

Derivations for sparse coding

F.1 The maximum entropy distribution

We find the maximum entropy distribution for a continuous random variable y that satisfies the usual conditions,

$$p_Y(y) \geq 0 \quad , \quad y \in \mathcal{R} \qquad\qquad \int_{-\infty}^{\infty} p_Y(y)dy = 1, \tag{F.1}$$

and the moment constraints in equation 8.5. Hence, the appropriate cost function (for maximization) is,

$$\phi(p_Y) = h(P_Y) + \lambda_0 \left(\int_{-\infty}^{\infty} p_Y(y')dy' - 1 \right) + \sum_{k=1}^{l} \lambda_k \left(\int_{-\infty}^{\infty} r_k(y')p_Y(y')dy' - c_k \right),$$

where λ_k are Lagrange multipliers, and the entropy $h(p_Y)$ is given by equation 8.4. Differentiating with respect to $p_Y(y)$ gives,

$$\frac{\partial \phi}{\partial p_Y} = -\ln p_Y(y) - 1 + \lambda_0 + \sum_{k=1}^{l} \lambda_k r_k(y), \tag{F.2}$$

and setting this result to zero yields the extreme point of the cost function,

$$p_Y(y) = \exp\left(-1 + \lambda_0 + \sum_{k=1}^{l} \lambda_k r_k(y) \right). \tag{F.3}$$

Independent components are assigned a mean of 0 and a variance of 1. Hence, $l = 2$ and equation 8.6 becomes,

$$p_Y(y) = e^{\lambda_0 - 1} e^{\lambda_1 y + \lambda_2 y^2}.$$

Setting $\lambda_0 = \ln(2\pi)^{-1/2} + 1$, $\lambda_1 = 0$ and $\lambda_2 = 1/2$ yields a Gaussian distribution which satisfies the constraints in equation 8.5 and the maximum entropy distribution in equation F.1 and, hence, maximizing entropy.

F.2 FastICA update rule

To minimize ϕ in equation 8.8, we use the FastICA algorithm, proposed by Hyvärinen (1999) which employs approximate Newton steps in an iterative scheme. The gradient of ϕ (equation 8.8) is,

$$\nabla\phi(\mathbf{q}_i) = -\mathrm{E}(r(y_i))\mathrm{E}\left(r'(y_i)\mathbf{z}\right), \tag{F.4}$$

and ignoring the scalar value $-\mathrm{E}\left(r(y_i)\right)$ allows for computation of an approximate Hessian $\mathbf{H}$ such that,

$$\mathbf{H} \approx \mathrm{E}\left(r''(y_i)\right)\mathrm{E}(\mathbf{z}\mathbf{z}^T) = \mathrm{E}(r''(y_i))\mathbf{I}.$$

This approximation gives a Hessian that is easily inverted, leading to the approximate Newton step (from iteration k to $k+1$) given by,

$$\mathbf{q}_i^{(k+1)} = \mathbf{q}_i^{(k)} - \frac{\mathrm{E}(r'(y_i)\mathbf{z})}{\mathrm{E}(r''(y_i))}. \tag{F.5}$$

Multiplying equation F.5 through by the denominator in its third term yields,

$$\mathrm{E}\left(r''(y_i)\right)\mathbf{q}_i^{(k+1)} = \mathrm{E}\left(r''(y_i)\right)\mathbf{q}_i^{(k)} - \mathrm{E}\left(r'(y_i)\mathbf{z}\right).$$

Hence, an appropriate update rule for the i^{th} row of $\mathbf{Q}$ is,

$$\begin{aligned} \mathbf{q}_i^{(k+1)} &= \mathrm{E}(r''(y_i))\mathbf{q}_i^{(k)} - \mathrm{E}(r'(y_i)\mathbf{z}) \\ \mathbf{q}_i^{(k+1)} &\leftarrow \frac{\mathbf{q}_i^{(k+1)}}{||\mathbf{q}_i^{(k+1)}||_2} \end{aligned} . \tag{F.6}$$

The projection back onto the unit circle accounts for the constraint $\mathbf{q}_i^T\mathbf{q}_i = 1$. For the algorithm used in this thesis, all rows of $\mathbf{Q}$ are updated simultaneously. That is, for each iteration of the optimization routine, 1) each row $\mathbf{q}_i^T$ of $\mathbf{Q}$ is updated according to equation F.6, and 2) the rows of $\mathbf{Q}$ are made orthogonal using symmetric orthogonalization such that,

$$\mathbf{Q} \leftarrow \mathbf{Q}\left(\mathbf{Q}^T\mathbf{Q}\right)^{-1/2}.$$

F.3 Sparse code shrinkage

We derive equation 8.10 where the sparse code y_i is corrupted with noise so that, dropping the subscript, $\hat{y} = y + n$. First, we write Bayes formula,

$$p(y|\hat{y}) = \frac{p(\hat{y}|y)p(y)}{p(\hat{y})},$$

where $p(\hat{y}) = \int_{-\infty}^{\infty} p(y)p(\hat{y}|y)dy$ is a constant. Here, we let $p(\hat{y}|y) = p_n(\hat{y} - y)$ where n is random noise. Next, we maximize the posterior density $p(y|\hat{y})$ with respect to y. In particular, letting $n \sim N(0, \sigma_n)$ gives,

$$\begin{aligned} \arg\max_y p\left(y|\hat{y}\right) &= \arg\max_y \left[\frac{1}{\sqrt{2\pi}\sigma_n} \exp\left(-\frac{\left(\hat{y}-y\right)^2}{2\sigma_n^2}\right) p\left(y\right)\right] \\ &= \arg\max_y \left[\exp\left(-\frac{\left(\hat{y}-y\right)^2}{2\sigma_n^2} + \ln p\left(y\right)\right)\right] \\ &= \arg\min_y \left[\frac{1}{2\sigma_n^2}\left(\hat{y}-y\right)^2 - \ln p\left(y\right)\right] \\ &= \arg\min_y \left[\frac{1}{2\sigma_n^2}\left(\hat{y}-y\right)^2 + \phi_p\left(y\right)\right], \end{aligned} \tag{F.7}$$

where $\phi_p\left(y\right) = -\ln p\left(y\right)$ is the model norm. Equation F.7 has its solution when,

$$\frac{1}{\sigma_n^2}\left(y - \hat{y}\right) + f'\left(y\right) = 0 \Rightarrow y = \hat{y} - \sigma_n^2 f'\left(y\right). \tag{F.8}$$

For example, consider, as Hoyer (1999) does, the following probability density function,

$$p\left(y\right) = c\exp\left(-ay^2/2 - b|y|\right),$$

where c is a constant, and the parameters a and b are adjusted, allowing control over the sparseness of y. In this case, the model norm is,

$$\phi_p\left(y\right) = -\ln\left[c\exp\left(-ay^2/2 - b|y|\right)\right] = -\ln c + ay^2/2 + b|y|. \tag{F.9}$$

Taking the derivative of equation F.9 with respect to y gives,

$$\phi_p'\left(y\right) = ay + b\frac{y}{|y|} = ay + b\,\mathrm{sign}(y),$$

and applying this result to equation F.8 gives,

$$\begin{aligned} y &= \frac{1}{1+\sigma^2 a}\left(\hat{y} - \sigma^2 b \operatorname{sign}(y)\right) \\ &= \frac{1}{1+\sigma^2 a}\operatorname{sign}(y)\left(\frac{\hat{y}}{\operatorname{sign}(y)} - \sigma^2 b\right). \end{aligned}$$

Finally, we let $\operatorname{sign}(y) = \operatorname{sign}(\hat{y})$, and ensure that the choice of b does not flip the sign of the coefficients such that,

$$y = \frac{1}{1+\sigma^2 a}\operatorname{sign}(\hat{y})\max\left(0, |\hat{y}| - \sigma^2 b\right),$$

which is the same as equation 8.10.