Wavelet Domain Inversion and Joint Deconvolution/Interpolation of Geophysical Data

by

Jonathan A. Kane

B.S., Virginia Tech (1996)

Submitted to the Department of Earth Atmospheric and Planetary Sciences
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2003

Author ....
Department of Earth Atmospheric and Planetary Sciences
January 13, 2003

Certified by. ....
M. Nafi Toksöz
Professor
Thesis Supervisor

Accepted by ....
Ronald Prinn
Chairman, Department Committee on Graduate Students

Wavelet Domain Inversion and Joint Deconvolution/Interpolation of Geophysical Data

by

Jonathan A. Kane

Submitted to the Department of Earth Atmospheric and Planetary Sciences
on January 13, 2003, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

This thesis presents two innovations to geophysical inversion. The first provides a framework and an algorithm for combining linear deconvolution methods with geostatistical interpolation techniques. This allows for sparsely sampled data to aid in image deblurring problems, or, conversely, noisy and blurred data to aid in sample interpolation. In order to overcome difficulties arising from high dimensionality, the solution must be derived in the correct framework and the structure of the problem must be exploited by an iterative solution algorithm. The effectiveness of the method is demonstrated first on a synthetic problem involving satellite remotely sensed data, and then on a real 3-D seismic data set combined with well logs.

The second innovation addresses how to use wavelets in a linear geophysical inverse problem. Wavelets have lead to great successes in image compression and denoising, so it is interesting to see what, if anything, they can do for a general linear inverse problem. It is shown that a simple nonlinear operation of weighting and thresholding wavelet coefficients can consistently outperform classical linear inverse methods in terms of mean-square error across a broad range of noise magnitude in the data. Wavelets allow for an adaptively smoothed solution: smoothed more in uninteresting regions, less at geologically important transitions.

A third issue is also addressed, somewhat separate from the first two: the correct manipulation of discrete geophysical data. The theory of fractional splines is introduced, which allows for optimal approximation of real signals on a digital computer. Using splines, it can be shown that a linear operation on the spline can be equivalently represented by a matrix operating on the coefficients of a certain spline basis function. The form of the matrix, however, depends completely on the spline basis, and incorrect discretization of the operator into a matrix can lead to large errors in the resulting matrix/vector product.

Thesis Supervisor: M. Nafi Toksöz
Title: Professor

Acknowledgments

I have had the privilege to cross paths with many intelligent and gifted people at MIT. Being surrounded by such a talented community has pushed me to be the best I can be, and probably made me into a better scientist than any other university would have. The road was not easy, though, and I could not have done it without the help of others.

There are many people that I must thank for my success. First, my advisor, Nafi Toksöz, who gave me years of financial and emotional support, and a free hand to shape my doctoral thesis. Along with Nafi I must thank Felix Herrmann, who turned me on to wavelets and fractional splines; Bill Rodi, for innumerable conversations about, and insight into, inverse theory; and John Grotzinger, for many trips to the field and a general appreciation for geology.

I owe special gratitude to my parents, whose support, encouragement, and sacrifices through the years gave me the opportunity to pursue an academic career. I cannot thank them enough. I must also acknowledge the rest of my family and close friends for constant support and companionship through good and bad times, and for providing balance in a place that tends ruin one's social life. In no particular order, I thank my brother David and family, Matt, Mike and Suzy, Usman, Scott, Maria and family, Gordon, Gil, Omar, Isabel, Rosalia, Danny and family, Nana, Chris, Lili, Ximena, Alicia, Paulina, the Souths, my friends at GCF, the bible study group - and too many others to name. Your love has kept me strong through these years.

My greatest thanks are to God for the multitude of blessings in my life that I so often take for granted. He gave me the talent and desire to achieve a Ph.D. in Geophysics from MIT. I owe everything to Him.

"You anoint my head with oil; my cup overflows. Surely goodness and love will follow me all the days of my life ....." (Psalm 23)

Contents

List of Figures

Chapter 1

Introduction

In geophysics one is often faced with the problem of inferring physical properties of the Earth from indirect noisy measurements. Examples range from inferring subsurface petrophysical properties from surface seismic data, to mapping topographic features via satellite-borne sensors. The process of obtaining information about these parameters is known alternatively as **estimation** or **inversion**. Estimation can be difficult due to the complicated physical laws governing the observed system, excessive noise in data, or the large size of geophysical data sets. The inherent numerical inaccuracy of digital computers can also complicate the problem.

This thesis is essentially about geophysical inversion and offers two main contributions. The first is a method to combine two seemingly different geophysical problems: deconvolution and interpolation. To link them, they are both posed in the framework of inverse theory, and then subsequently combined in a single joint inverse problem. The result is a deconvolution method that integrates blurred data with existing point samples of some parameter field and geostatistical constraints. Conversely, it can be viewed as an interpolation method that uses blurred data to constrain the interpolation results away from the point samples. We apply this method to two different applications, first, to 3-D poststack seismic and well log data (Chapter 3), then to satellite based remote sensing data in the form of a digital elevation model (DEM) and global positioning system (GPS) data (Chapter 4). Due to the large size of the data sets dealt with, standard algorithms cannot be applied. One is required to make clever use of the structure of the problem along with an efficient iterative solution method. We use the conjugate gradients algorithm to solve the problem.

The second major contribution of the thesis is the use of wavelets (Meyer, 1992; Strang and Nguyen, 1997; Mallat, 1998) in solving linear geophysical inverse problems. In the past 10 years wavelets have shown themselves to be superb for image compression (DeVore et al., 1992) and denoising (Mallat, 1998). We ask the question of what (if anything) they can gain in a linear inverse problem. We follow recent work in abstract estimation theory by Donoho (1995), who has introduced the so-called *wavelet-vaguelette* decomposition for solving a certain class of linear inverse problems. We elaborate on and extend his work in the context of a 1-D geophysical slowness estimation problem. Our wavelet based strategy is applied to both synthetic and real examples. For the synthetic problem, wavelet based inversion beats classical linear inverse methods across a large range of noise magnitudes in the data.

Another goal of this thesis, somewhat disjoint from the first two, is to present fractional splines, a new class of functions useful for representing and discretizing continuous functions. The discretization process and correct manipulation of the resulting discrete vectors is often overlooked in the geophysical community. Careless treatment of these coefficients can needlessly incur error in computer applications. The theory of fractional splines is presented in Chapter 5, along with a simple application of discretizing the derivative operator and operating it on real well log data. Fractional spline basis functions can also be converted into wavelets. We connect wavelets and fractional splines on a more rigorous level in Chapter 6. Fractional spline wavelets inherit all the beneficial properties of the fractional spline basis functions, plus the ability to analyze and synthesize non-stationary signals more efficiently. These wavelets are used in the wavelet inversion of Chapter 7.

1.1 Outline and contributions

Chapters 2, 5, and 6 of this thesis concentrate on essential background theory necessary for understanding the other chapters. Chapter 2 gives a brief overview of classical linear statistical estimation theory, which is necessary for Chapters 3, 4, and 7. Chapter 5 presents splines, and Chapter 6, wavelets, both essential for Chapter 7. Chapters 3, 4, and 7 present applications of new inversion methods to geophysical data, and are the meat of the thesis. A reader with with a solid background in the material of chapters 2, 5, and 6 is recommended to go directly to the application chapters.

We list here a brief synopsis of each chapter along with its major contributions to the thesis as a whole.

Chapter 2: Discrete linear inverse theory

This chapter presents an introduction to discrete linear inverse theory. We follow the Bayesian point of view (Tarantola, 1987), which specifies a *prior* probability distribution on the parameter(s) to be estimated, and then, combined with a *likelihood function*, gives the solution as a *posterior* probability density function. Bayesian methods require finite dimensional model and data vectors in order to be correctly posed. Since all applications involve discrete data and operators we limit ourselves to such finite dimensional system. We also limit ourselves to problems with linear forward modeling operators. This is sufficient for all applications in the thesis.

The chapter begins by stating the general inference problem, then the Bayesian inference problem, and then narrows the scope to Bayesian linear inverse problems. Two frameworks are used to derive a solution to the problem. The first usually is associated with *minimum variance* estimation methods and seeks to derive a function called an *estimator* that maps data to models. The second framework is usually associated with *maximum likelihood* estimation methods. It searches for a model vector that minimizes the data misfit, subject to a prior covariance constraint. Under certain conditions these methods are equivalent, but one may be more computationally efficient than the other, depending on the relative sizes of model and data vectors. The second framework is the more versatile one and is used throughout the thesis.

Chapter 3: 3-D geostatistically constrained seismic deconvolution

Deconvolution has been worked on in the geophysical community for the last half century (Robinson, 1954; Robinson and Trietel, 1980). A variety of linear and nonlinear methods have been applied to produce optimal reflectivity sequences or wavelets (depending on the application). These methods have been largely applied in a 1-D sense, where a separate inverse problem is essentially solved sequentially at each seismic trace location; i.e., if a seismic data cube consists of large number of horizontally correlated seismic traces, nothing in the deconvolution process takes into account the fact that the underlying reflectivity solution should also be horizontally correlated. Nor does it honor the existing well data.

Interpolation methods used for geophysical data typically go by the name *kriging* (Deutsch and Journel, 1998; Isaaks and Srivastava, 1989), named after the South African mining engineer, D. Krige, who pioneered the method. Kriging is the domain of the geostatistics community. They traditionally have derived a solution to the interpolation problem in a minimum variance framework. This produces computational difficulties that limit the size of the data sets to be interpolated. In the last few years a few workers (Fomel, 2001) have discovered that the same interpolation problem can be posed in an alternate, but equivalent, framework. This change of thinking leads to a way to interpolate large 3-D data sets that, until now, have only been possible by using approximate kriging methods.

Posing both the deconvolution and kriging problems in this alternate inversion framework allows them to be combined into a single inverse problem. Using multiple data sets will then lead to a better solution than using only one of them.

Using a 3-D poststack, post time migrated seismic data set along with velocity and density logs, we jointly deconvolve for reflectivity under a zero offset seismic convolution model, while honoring and interpolating the well data.

It is easier to pose the solution of this joint inverse problem than to actually solve it. The sheer size of 3-D seismic data sets precludes all but a few solution methods.

It is shown that the conjugate gradients algorithm, along with the structure of the problem, allow for efficient solution.

Chapter 4: Joint deconvolution and interpolation of remote sensing data

This chapter essentially applies the same mathematical method of Chapter 3 to a different application: estimating topography jointly from satellite data and point samples of elevation. A synthetic problem is constructed by taking a real DEM, blurring it, and adding noise. This is the first data set. Point samples are randomly extracted from the DEM and noise is added to them to produce the second data set. This second data set mimics GPS data, which measure elevation at a much higher resolution than DEM's do. The two data sets are then simultaneously inverted for an optimal solution. It is shown that the joint solution can be computed efficiently by the same method as in Chapter 3, and leads to smaller mean square error than using only one of the data sets by itself.

Chapter 5: Fractional splines and discrete data

Splines are very useful functions for a variety of reasons:

- **Approximation**

  A finite linear combination of spline basis functions can be used to approximate other, more complicated, functions. The resulting approximation error can be show to be less for an appropriately chosen spline than for any other approximating function.

- **Continuous⟷discrete**

  Projecting a complicated function or operator onto a spline space allows for discrete representation of both the function and operator. Thus manipulation of the spline can be carried out equivalently by discrete operations on its coefficients. Incorrect discretization can lead large errors when manipulating discrete data sets.

- **Wavelets**

  Splines basis functions are suitable for creating wavelets, a class of mathematical functions with additional useful properties.

Fractional splines are a generalization of the traditional integer degree splines studied in the mathematical community for the last 50 years (Schoenberg, 1946; Unser, 1999). They allow for optimal representations of a continuous function given knowledge only of its Hölder continuity.

This chapter first introduces concepts such as Riesz bases and the projection of a continuous function into a function space spanned by a certain set of basis functions. It then introduces splines as an ideal basis for the projections. Several candidate spline basis functions are presented, each with useful mathematical properties. Next, spline bases are also used to discretize linear operators. An example of taking the derivative of well data is shown.

Chapter 6: Wavelets

Building upon the material of Chapter 5, this chapter introduces wavelets, functions capable of analyzing and synthesizing other continuous functions locally in both the spatial and frequency domains. This property makes them optimal for representing non-stationary signals, and led to their success in signal compression and denoising.

This chapter is mostly a repetition of the theory presented in Mallat (1998) and Strang and Nguyen (1997), along with the new theory of fractional splines (Unser and Blu, 1999, 2000). It attempts to give a clear derivation of all formulas and concepts needed to fully understand semi-orthogonal spline wavelets basis, and associated filter banks. Filter banks allow for fast implementations of the wavelet transform on a digital computer, which is required for applications in Chapter 7.

Chapter 7: Wavelet domain geophysical inversion

In 1995 Donoho presented an exciting new methodology for solving linear inverse problems known as the wavelet-vaguelette decomposition (WVD). It involves wavelet transforming the solution of an inverse problem and then thresholding the wavelet coefficients that lie below some threshold level and then inverse wavelet transforming. Donoho (1995) theoretically predicts that this nonlinearly thresholded solution will asymptotically have a smaller mean-square error (MSE) as the noise level decreases than solutions obtained via classical regularized inverse methods.

This chapter presents and expands upon Donoho's WVD method. The method is presented in a linear algebraic formulation, which allows for a more intuitive and understandable derivation. It is applied to the 1-D linear inverse problem of estimating a slowness model from noisy traveltime data, simulating the VSP experiment of exploration seismology. MSE error of the WVD method is compared to the results from classical regularized inverse methods across a large range of noise magnitudes in the data.

Chapter 8: Conclusions

In this chapter we summarize the the major contributions and limitations of the thesis. We also postulate future directions for the research.

Chapter 2

Survey of linear inverse theory

Most techniques used by geophysicists to infer information about the subsurface can be posed as special cases of inverse theory. Examples include deconvolution, travel-time estimation, and the interpolation methods commonly used by the geostatistical community. In this chapter we will limit ourselves to discrete inverse problems involving linear forward modeling operators. We derive classical linear inverse operators to solve the problem.

Inversion has been well studied in and out of the geophysical community on both computational and theoretical levels. Books such as Tarantola (1987); Morozov (1993); Parker (1994); Bertero and Boccacci (1998); Vogel (2002) lay out the theory in detail. We do not attempt to be rigorous here, but rather give an overview and a few essential results as a background for later chapters.

We begin this chapter by describing the general Bayesian inference problem, which involves estimating a probabilistic model of reality from a probabilistic observation of the world. We then narrow the inference problem down to Bayesian linear inversion, where the model and observation are vectors of real numbers related to each other by some linear forward operator. We then derive two frameworks for solving such a problem. The benefits and drawbacks of each framework are explained. We do not describe solution algorithms, but rather refer the reader to the above mentioned sources.

2.1 Bayesian inference

We define **inference** as the procedure of observing some event and obtaining information from it. More specifically, it involves the following system:

1. A set, $\mathcal{F}$, of possible **models** of reality which may exist.

2. A set, $\mathcal{Z}$, of possible observations called **data**.

3. A relation, H, that associates models in $\mathcal{F}$ with data in $\mathcal{Z}$.

Solving the inference problem involves taking a particular $z_{obs} \in \mathcal{Z}$ that we observe and trying to infer its associated element(s) $f \in \mathcal{F}$. We define the relation H to be a subset of the Cartesian product $\mathcal{P} = \mathcal{F} \times \mathcal{Z}$ (Naylor and Sell, 1982):

$$\mathrm{H} = \{(f, z) : f \text{ is related to } z\}. \tag{2.1}$$

The solution to the inference problem is then the set of elements

$$\{f \in \mathcal{F} : (f, z_{obs}) \in \mathrm{H}\}. \tag{2.2}$$

The inference problem can be generalized to **Bayesian inference** by assigning a probability to each of the elements of both $\mathcal{F}$ and $\mathcal{Z}$. In order to assign probability, the elements of these sets must be finite dimensional random vectors.

We will denote such vectors by boldface italic letters: $\boldsymbol{f}, \boldsymbol{g}, \dots$, etc. Realizations of random vectors will be identified by boldface Roman letters: $\mathbf{f}, \mathbf{g}, \dots$, etc. Individual coordinates of any of the above vector objects will be given by the equivalent non-boldface font with a subscript, i.e. $f_1, f_2, \dots$, etc. We let the sets of events corresponding to the model and data spaces be denoted $\mathfrak{F}$ and $\mathfrak{Z}$, respectively. Vector valued probabilistic events from these sets will be denoted by boldface frakture letters: $\boldsymbol{\mathfrak{f}}, \boldsymbol{\mathfrak{g}}, \dots$, etc.

Restricting ourselves to such finite dimensional sets we state the basic axioms of probability (Papoulis, 1991; Willsky et al., 1999) in terms of the set $\mathfrak{P} = \mathfrak{F} \times \mathfrak{Z}$ and arbitrary events[1] $\mathfrak{f}, \mathfrak{z} \in \mathfrak{P}$:

$$P(\mathfrak{f}) \geq 0, \tag{2.3}$$

$$P(\mathfrak{P}) = 1, \tag{2.4}$$

$$\text{if } \mathfrak{f} \cap \mathfrak{z} = \{\emptyset\} \text{ then } P(\mathfrak{f} \cup \mathfrak{z}) = P(\mathfrak{f}) + P(\mathfrak{z}), \tag{2.5}$$

where $P : \mathfrak{P} \mapsto \mathbb{R}^+$ is a functional that returns the probability of a given event.

The conditional probability of an event $\mathfrak{f}$ given another event $\mathfrak{z}$ is defined through the following relationship:

$$P(\mathfrak{f}|\mathfrak{z}) = \frac{P(\mathfrak{f} \cap \mathfrak{z})}{P(\mathfrak{z})}. \tag{2.6}$$

Similarly, the conditional probability of $\mathfrak{z}$ given $\mathfrak{f}$ is defined as

$$P(\mathfrak{z}|\mathfrak{f}) = \frac{P(\mathfrak{z} \cap \mathfrak{f})}{P(\mathfrak{f})}. \tag{2.7}$$

From these equations we can see that

$$P(\mathfrak{f} \cap \mathfrak{z}) = P(\mathfrak{f}|\mathfrak{z})P(\mathfrak{z}) = P(\mathfrak{z}|\mathfrak{f})P(\mathfrak{f}). \tag{2.8}$$

We also can combine Equation 2.6 and 2.7 to get

$$P(\mathfrak{f}|\mathfrak{z}) = \frac{P(\mathfrak{z}|\mathfrak{f})P(\mathfrak{f})}{P(\mathfrak{z})}. \tag{2.9}$$

Furthermore, if we let $\mathfrak{f}_j$ be an element in a partition, $\{\mathfrak{f}_1, \mathfrak{f}_2, \ldots, \mathfrak{f}_N\}$, of $\mathfrak{P}$, and $\mathfrak{z}$ an arbitrary event, then Equation 2.9 can be extended (Papoulis, 1991, p. 30) to

$$P(\mathfrak{f}_j|\mathfrak{z}) = \frac{P(\mathfrak{z}|\mathfrak{f}_j)P(\mathfrak{f}_j)}{P(\mathfrak{z}|\mathfrak{f}_1)P(\mathfrak{f}_1) + P(\mathfrak{z}|\mathfrak{f}_2)P(\mathfrak{f}_2) + \cdots + P(\mathfrak{z}|\mathfrak{f}_N)P(\mathfrak{f}_N)}. \tag{2.10}$$

This equation is known as **Bayes rule** and is of central importance to inverse theory.

[1] An **event** here will be a partial or full realization of vector $\mathfrak{f} \in \mathfrak{P}$, i.e. a realization $\mathbf{s} \in \mathcal{P}$ can be an event but so can the realization of just a subset of its coordinates, such as $\mathbf{f}$ or $\mathbf{z}$.

Now we let the event $\mathfrak{f}_j$ in Equation 2.10 be $\mathbf{f}$, the realization of a finite dimensional model space vector, and the event $\mathfrak{z}$ be $\mathbf{z}$, an observation of an element of the data space. For the purposes of this thesis we restrict each coordinate, f_j, in the random vector $\boldsymbol{f}$ to be a random variable defined over the real numbers with a continuous probability density function (PDF). We denote the joint PDF of $\boldsymbol{f}$ as $p_{\boldsymbol{f}}(\mathbf{f})$. These specifications allow us to represent Bayes rule (Equation 2.10) in terms of PDF's (Tarantola, 1987):

$$p_{\boldsymbol{f}|\boldsymbol{z}}(\mathbf{f}|\mathbf{z}) = \frac{p_{\boldsymbol{z}|\mathbf{f}}(\mathbf{z}|\mathbf{f})p_{\boldsymbol{f}}(\mathbf{f})}{\int_{-\infty}^{\infty} p_{\boldsymbol{z}|\mathbf{f}}(\mathbf{z}|\mathbf{f})p_{\boldsymbol{f}}(\mathbf{f})d\mathbf{f}}. \tag{2.11}$$

Noticing that $\int_{-\infty}^{\infty} p_{\boldsymbol{z}|\mathbf{f}}(\mathbf{z}|\mathbf{f})p_{\boldsymbol{f}}(\mathbf{f})d\mathbf{f} = p_{\boldsymbol{z}}(\mathbf{z})$ we can also write the equivalent of Equation 2.8:

$$p_{\boldsymbol{f},\boldsymbol{z}}(\mathbf{f},\mathbf{z}) = p_{\boldsymbol{f}|\boldsymbol{z}}(\mathbf{f}|\mathbf{z})p_{\boldsymbol{z}}(\mathbf{z}) = p_{\boldsymbol{z}|\mathbf{f}}(\mathbf{z}|\mathbf{f})p_{\boldsymbol{f}}(\mathbf{f}). \tag{2.12}$$

$p_{\boldsymbol{z}|\mathbf{f}}(\mathbf{z}|\mathbf{f})$ is known as the **likelihood function** and is only a PDF for a given vector $\mathbf{f}$. $p_{\boldsymbol{f}}(\mathbf{f})$ and $p_{\boldsymbol{f}|\boldsymbol{z}}(\mathbf{f}|\mathbf{z})$ are known as the **prior** and **posterior** PDF's, respectively. Solution to the estimation problem involves calculating some statistic of $p_{\boldsymbol{f}|\boldsymbol{z}}(\mathbf{f}|\mathbf{z})$. This may be its maximal value but also can be another statistic such as its mean value.

It should be noted that for geophysicists the elements of $\mathcal{F}$ are often continuous (infinite dimensional) petrophysical parameter fields, $f(\mathbf{x})$, defined over real Euclidean space, $\mathbf{x} \in \mathbb{R}^d$. The elements of $\mathcal{Z}$ are also often continuous fields, $z(\mathbf{x}')$, defined on another Euclidean space $\mathbf{x}' \in \mathbb{R}^{d'}$. d and d' usually range from 1 to 4 spatio-temporal dimensions. Such a system does not fit into the Bayesian framework[2] of equation 2.11. To assign probability to such geophysical parameter fields, we must discretize the continuous function over a finite interval to obtain finite dimensional model and

[2] One can define **stochastic processes**, the infinite dimensional analog to random vectors, in an infinite dimensional space. By making use of the second moments of the stochastic processes one can still define estimation of continuous functions with prior models. The method is known as **stochastic inversion**. It is, however, limited to linear forward operators. Interested readers are referred to Franklin (1970) and Prenter and Vogel (1985).

data spaces. Discretization is also essential for manipulating elements of $\mathcal{F}$ or $\mathcal{Z}$ on a computer. The process of discretization is outlined in Chapter 5.

2.2 Bayesian linear inversion

For the situations that will interest us we have the relation between elements of $\mathcal{F}$ and $\mathcal{Z}$ as

$$\boldsymbol{z} = \mathbf{H}\boldsymbol{f} + \boldsymbol{n}, \tag{2.13}$$

where $\mathbf{H} : \mathcal{F} \mapsto \mathcal{Z}$ is a linear operator (a matrix) and $\boldsymbol{n}$ is a vector of zero mean white noise[3]. $\boldsymbol{n}$ models the inherent inaccuracy of the operator $\mathbf{H}$ as well as independent processes in the system unrelated to the operator $\mathbf{H}$ or the vector $\boldsymbol{f}$.

Given an observed vector $\mathbf{z}_{obs}$, the Bayesian linear inverse problem involves finding the single most probable vector, $\mathbf{f}_{est}$, called the **maximum a-posteriori** (MAP) estimate, that satisfies Equation 2.13.

In what follows we will restrict the PDF of both $\boldsymbol{f}$, $\boldsymbol{n}$, and $\boldsymbol{z}$ to be Gaussian. Using $\boldsymbol{f}$ as an example, Gaussian vector takes the form:

$$p_{\boldsymbol{f}}(\mathbf{f}) = \frac{1}{2\pi^{N/2} \det \mathbf{C}_{\mathbf{f}}{}^{1/2}} \exp\left(-\frac{1}{2}\mathbf{f}^T \mathbf{C}_{\mathbf{f}}^{-1} \mathbf{f}\right). \tag{2.14}$$

This is done for two reasons. The derivation of a solution to the linear inverse problem is much simpler in the Gaussian case, and, empirical PDF's encountered in many applications are well approximated by a Gaussian distribution.

There are basically two methodologies that lead to $\mathbf{f}_{est}$. The first seeks an inverse operator of $\mathbf{H}$ that produces statistically optimal solutions. The second optimizes an objective function for a best fitting solution $\mathbf{f}$. Each of these methods has its benefits and drawbacks which we will elaborate on.

[3] Discrete "white noise" is a vector of independent random variables.

2.2.1 Method 1

This first solution method usually goes by the name of minimum variance estimation. It involves constructing a function, $H^\dagger$, called the **estimator** that maps vectors in $\mathcal{Z}$ back into $\mathcal{F}$. Even if $\mathbf{H}$ is a linear forward modeling operator, $H^\dagger$ is not restricted to be linear[4]. We will restrict it here to be a linear matrix operator $\mathbf{H}^\dagger$, and we have

$$\boldsymbol{f} = \mathbf{H}^\dagger \boldsymbol{z} + \boldsymbol{e}. \tag{2.15}$$

$\boldsymbol{e}$ is another random vector, called the **error**, measuring the inaccuracy of the inverse operator. Rearranging Equation 2.15, we have

$$\boldsymbol{e} = \boldsymbol{f} - \mathbf{H}^\dagger \mathbf{z}. \tag{2.16}$$

We have the PDF of both $\boldsymbol{z}$ and $\boldsymbol{f}$; it remains to find the inverse operator $\mathbf{H}^\dagger$. This is done by minimizing the expected value of some **cost function**, $C(\cdot)$, of $\boldsymbol{e}$ over all possible matrix operators $\mathbf{A}$:

$$\begin{aligned} \mathbf{H}^\dagger &= \arg\min_{\mathbf{A}} E\left[C(\boldsymbol{e})\right]. && (2.17)\\ &= \arg\min_{\mathbf{A}} E\left[C(\boldsymbol{f} - \mathbf{A}\boldsymbol{z})\right] && (2.18)\\ &= \arg\min_{\mathbf{A}} \int\int C(\mathbf{f} - \mathbf{A}\mathbf{z}) p_{\boldsymbol{f},\boldsymbol{z}}(\mathbf{f},\mathbf{z}) d\mathbf{f} d\mathbf{z} && (2.19) \end{aligned}$$

Minimizing different cost functions will give different solutions. The most convenient cost functions to minimize are weighted ℓ_p norms to the p power. Minimizing an ℓ_1 norm leads to the median value of the posterior PDF of $\mathbf{f}_{est}$; the ℓ_2 norm leads to the mean value; the ℓ_∞ leads to the *mid-range* (Tarantola, 1987, p. 18). Which norm to use is an interesting question. To answer this we first notice that as long as $\mathbf{H}^\dagger$ is a matrix operator and $\boldsymbol{f}$ is a Gaussian random vector, $\boldsymbol{e}$ will tend to have a Gaussian distribution regardless of the PDF defined on $\boldsymbol{z}$. This is because linear

[4] The term "linear" in linear inversion pertains to the forward modeling operator. We show in Chapter 7 that simple nonlinear estimators for linear inverse problems can outperform linear estimators when the model space is poorly approximated as a stationary Gaussian distribution.

combinations of any random variables of finite variance tends towards a Gaussian distribution. Therefore the ℓ_2 norm is the most natural, leading to the mean of the posterior PDF. If $H^\dagger$ is not linear, or the prior PDF non-Gaussian, the posterior PDF may be very far from Gaussian and it is hard to tell what norm to minimize.

The easiest way to obtain the estimator is to do it point-wise for each row of $\mathbf{H}^\dagger$. Denoting the j^{th} row as $\mathbf{h}_j^\dagger$, and rewriting Equation 2.16 in the scalar random variable case, we have

$$e_j = f_j - \mathbf{h}_j^\dagger \boldsymbol{z}. \tag{2.20}$$

To estimate $\mathbf{h}_j^\dagger$ we must minimize the expected value of a scalar random variable cost function of e_j. In this scalar case the ℓ_p cost functions correspond to absolute moments of e_j (Papoulis, 1991). The expected cost of Equation 2.20 is

$$E\left[C(e_j)\right] = \int\int C(e_j) p_{f_j,\boldsymbol{z}}(\mathrm{f}_j, \mathbf{z}) d\mathrm{f}_j d\mathbf{z}. \tag{2.21}$$

Assuming, for the moment, that f_j and $\boldsymbol{z}$ have zero mean, we use an ℓ_2 cost function and minimize variance over all vectors $\mathbf{a}$:

$$\begin{aligned}
\mathbf{h}_j^\dagger &= \arg\min_{\mathbf{a}_j} E\left[C(\boldsymbol{e}_j)\right] && (2.22)\\
&= \arg\min_{\mathbf{a}_j} E\left[(\mathrm{f}_j - \boldsymbol{z}^T\mathbf{a}_j)^2\right] && (2.23)\\
&= \arg\min_{\mathbf{a}_j} E\left[(\mathrm{f}_j)^2 + \mathbf{a}_j^T\boldsymbol{z}\boldsymbol{z}^T\mathbf{a}_j - 2\mathrm{f}_j\boldsymbol{z}^T\mathbf{a}_j)\right] && (2.24)\\
&= \arg\min_{\mathbf{a}_j} \left(E\left[(\mathrm{f}_j)^2\right] + \mathbf{a}_j^T E\left[\boldsymbol{z}\boldsymbol{z}^T\right]\mathbf{a}_j - 2E\left[\mathrm{f}_j\boldsymbol{z}^T\right]\mathbf{a}_j\right) && (2.25)\\
&= \arg\min_{\mathbf{a}_j} \left(\sigma_{\mathrm{f}_j}^2 + \mathbf{a}_j^T\mathbf{C}_{\boldsymbol{z}}\mathbf{a}_j - 2\mathbf{a}_j^T\mathbf{c}_{f_j,\boldsymbol{z}}\right), && (2.26)
\end{aligned}$$

where $\mathbf{c}_{f_j,\boldsymbol{z}}$ is the cross-covariance between f_j and $\mathbf{z}$. The minimizing vector $\mathbf{h}_j^\dagger$ will satisfy

$$\begin{aligned}
0 &= \frac{\partial}{\partial \mathbf{a}_j}\left(\sigma_{\mathrm{f}_j}^2 + \mathbf{a}_j^T\mathbf{C}_{\boldsymbol{z}}\mathbf{a}_j - 2\mathbf{a}_j^T\mathbf{c}_{f_j,\boldsymbol{z}}\right) && (2.27)\\
&= 2\mathbf{C}_{\boldsymbol{z}}\mathbf{a}_j - 2\mathbf{c}_{f_j,\boldsymbol{z}} && (2.28)
\end{aligned}$$

This leads us to the solution

$$\mathbf{h}_j^\dagger = \mathbf{c}_{f_j,\boldsymbol{z}}^T \mathbf{C}_{\boldsymbol{z}}^{-1}. \tag{2.29}$$

This vector goes into the j^{th} row of $\mathbf{H}^\dagger$. Proceeding in the same way for all rows we arrive at the estimator for the Bayesian linear inverse problem:

$$\mathbf{H}^\dagger = \mathbf{C}_{\boldsymbol{f},\boldsymbol{z}} \mathbf{C}_{\boldsymbol{z}}^{-1}. \tag{2.30}$$

This operator can be applied to any observed vector to obtain the solution:

$$\mathbf{f}_{est} = \mathbf{C}_{\boldsymbol{f},\boldsymbol{z}} \mathbf{C}_{\boldsymbol{z}}^{-1} \mathbf{z}_{obs}. \tag{2.31}$$

If we let $E[\boldsymbol{f}] = E[\boldsymbol{n}] = 0$, and assume that $E\left[\boldsymbol{f}\boldsymbol{n}^T\right] = \mathbf{0}$, we can establish the following equalities:

$$\begin{aligned}
\mathbf{C}_{\boldsymbol{f},\boldsymbol{z}} &= E\left[\boldsymbol{f}\boldsymbol{z}^T\right] && (2.32)\\
&= E\left[\boldsymbol{f}(\mathbf{H}\boldsymbol{f} + \boldsymbol{n})^T\right] && (2.33)\\
&= E\left[\boldsymbol{f}\boldsymbol{f}^T\mathbf{H}^T\right] && (2.34)\\
&= \mathbf{C}_{\boldsymbol{f}}\mathbf{H}^T && (2.35)
\end{aligned}$$

and

$$\begin{aligned}
\mathbf{C}_{\boldsymbol{z}} &= E\left[\boldsymbol{z}\boldsymbol{z}^T\right] && (2.36)\\
&= E\left[(\mathbf{H}\boldsymbol{f} + \boldsymbol{n})(\mathbf{H}\boldsymbol{f} + \boldsymbol{n})^T\right] && (2.37)\\
&= E\left[(\mathbf{H}\boldsymbol{f}\boldsymbol{f}^T\mathbf{H}^T + \boldsymbol{n}\boldsymbol{n}^T)\right] && (2.38)\\
&= (\mathbf{H}\mathbf{C}_{\boldsymbol{f}}\mathbf{H}^T + \mathbf{C}_{\boldsymbol{n}}). && (2.39)
\end{aligned}$$

Plugging Equation 2.35 and 2.39 into Equation 2.31 we obtain

$$\mathbf{f}_{est} = \mathbf{C}_{\boldsymbol{f}}\mathbf{H}^T(\mathbf{H}\mathbf{C}_{\boldsymbol{f}}\mathbf{H}^T + \mathbf{C}_{\boldsymbol{n}})^{-1}\mathbf{z}_{obs}. \tag{2.40}$$

If $\boldsymbol{f}$ has a known mean, $\mathbf{m}_{\boldsymbol{f}}$, we can incorporate it into Equation 2.40 as

$$\mathbf{f}_{est} = \mathbf{m}_{\boldsymbol{f}} + \mathbf{C}_{\boldsymbol{f}}\mathbf{H}^T(\mathbf{H}\mathbf{C}_{\boldsymbol{f}}\mathbf{H}^T + \mathbf{C}_{\boldsymbol{n}})^{-1}(\mathbf{z}_{obs} - \mathbf{H}\mathbf{m}_{\boldsymbol{f}}). \tag{2.41}$$

In looking at Equation 2.41 we see a strength of the methodology: it can still solve an inverse problem in the case of no measurement error $\boldsymbol{n}$. This is useful in applications such as exact interpolation of data, a.k.a. *kriging* (Deutsch and Journel, 1998). Method 2, below, cannot do this.

From the above derivation, however, we see that method 1 is a rather obtuse way of estimating a vector. When the PDF's of $\boldsymbol{n}$ or $\boldsymbol{f}$ are non-Gaussian, or the operator $\mathbf{H}$ nonlinear, it is not clear what kind of cost function to minimize, nor whether an estimator $\mathbf{H}^{\dagger}$ will even exist (Willsky et al., 1999). We will overcome these ambiguities with method 2.

2.2.2 Method 2

This methodology is usually presented with *maximum likelihood* inverse methods. We assume for the moment that $\boldsymbol{f}$ is not stochastic. Equation 2.13 then becomes

$$\boldsymbol{z} = \mathbf{H}\mathbf{f} + \boldsymbol{n}. \tag{2.42}$$

Rearranging Equation 2.13 then gives

$$\boldsymbol{n} = \boldsymbol{z} - \mathbf{H}\mathbf{f}. \tag{2.43}$$

For a given vector $\mathbf{f}$ the PDF of $\boldsymbol{n}$ is the same as the PDF of $\boldsymbol{z} - \mathbf{H}\mathbf{f}$, which is the likelihood function $p_{\boldsymbol{z}|\mathbf{f}}(\mathbf{z}|\mathbf{f})$.

The maximum likelihood solution to the inverse problem would be to find the maximum of $p_{\boldsymbol{z}|\mathbf{f}}(\mathbf{z}|\mathbf{f})$. Since $\boldsymbol{n}$ is a Gaussian random vector, it has the form

$$p_{\boldsymbol{z}|\mathbf{f}}(\mathbf{z}|\mathbf{f}) = \frac{1}{2\pi^{N/2}\det \mathbf{C}_{\boldsymbol{n}}{}^{1/2}} \exp\left(-\frac{1}{2}\left(\mathbf{z} - \mathbf{H}\mathbf{f}\right)^T \mathbf{C}_{\boldsymbol{n}}^{-1} \left(\mathbf{z} - \mathbf{H}\mathbf{f}\right)\right). \tag{2.44}$$

Maximization of this function corresponds to the minimization of its exponent, which happens to be a weighted ℓ_2 norm to the second power:

$$\begin{aligned} ||\mathbf{C}_n^{-1/2}(\mathbf{z}-\mathbf{Hf})||_{\ell_2}^2 &= (\mathbf{z}-\mathbf{Hf})^T\mathbf{C}_n^{-1}(\mathbf{z}-\mathbf{Hf}) && (2.45) \\ &= \mathbf{z}^T\mathbf{C}_n^{-1}\mathbf{z} - 2\mathbf{f}^T\mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{z} + \mathbf{f}^T\mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{Hf}. && (2.46) \end{aligned}$$

This is the objective function that we must minimize. The minimum can be found by taking the derivative and setting the result to zero:

$$\begin{aligned} \frac{\partial}{\partial\mathbf{f}}\left[\mathbf{z}^T\mathbf{C}_n^{-1}\mathbf{z} - 2\mathbf{f}^T\mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{z} + \mathbf{f}^T\mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{Hf}\right] &= 0 && (2.47) \\ -2\mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{z} + 2\mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{Hf} &= 0 && (2.48) \\ \mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{Hf} &= \mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{z} && (2.49) \end{aligned}$$

This relationship gives us the maximum likelihood solution:

$$\mathbf{f}_{est} = \left(\mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{z}_{obs}. \qquad (2.50)$$

This equation may give rise to a noise corrupted solution, or to a non-unique solution, depending on the operator **H**. Another constraint may be required. The Bayesian solution offers this constraint in the form of the prior PDF.

We can induce a smooth prior on $\boldsymbol{f}$ by saying that a differencing operator acting on it produces white noise:

$$\boldsymbol{n}' = \mathbf{L}\boldsymbol{f}. \qquad (2.51)$$

The covariance of $\boldsymbol{f}$ (assuming zero mean) is then

$$\begin{aligned} \mathbf{C}_{\boldsymbol{f}} &= E\left[\mathbf{L}^{-1}\boldsymbol{n}'\boldsymbol{n}'^T\mathbf{L}^{-T}\right] && (2.52) \\ &= \mathbf{L}^{-1}E\left[\boldsymbol{n}'\boldsymbol{n}'^T\right]\mathbf{L}^{-T} && (2.53) \\ &= \sigma_{\boldsymbol{n}'}^2(\mathbf{L}^T\mathbf{L})^{-1}. && (2.54) \end{aligned}$$

We can use the PDF, $p_{\boldsymbol{n}'}(\mathbf{n}')$, to represent $\boldsymbol{f}$. Multiplying this PDF times the likelihood function of Equation 2.11 brings us back to the Bayesian solution:

$$p_{\boldsymbol{f}|\boldsymbol{z}}(\mathbf{f}|\mathbf{z}) = k\, p_{\boldsymbol{n}}(\mathbf{n}) p_{\boldsymbol{n}'}(\mathbf{n}'), \tag{2.55}$$

where k incorporates the denominator of Equation 2.11.

The joint exponent of this posterior PDF has the form of two ℓ_2 norms added together:

$$\left[||\mathbf{C}_{\boldsymbol{n}}^{-1/2}\left(\mathbf{z} - \mathbf{Hf}\right)||_{\ell_2}^2 + ||\mathbf{C}_{\boldsymbol{f}}^{-1/2}\mathbf{Lf}||_{\ell_2}^2\right] \tag{2.56}$$

The solution is then the minimizing vector:

$$\mathbf{f}_{est} = \arg\min_{\mathbf{a}} \left[||\mathbf{C}_{\boldsymbol{n}}^{-1/2}\left(\mathbf{z} - \mathbf{Ha}\right)||_{\ell_2}^2 + ||\mathbf{C}_{\boldsymbol{f}}^{-1/2}\mathbf{La}||_{\ell_2}^2\right]. \tag{2.57}$$

We expand this equation, take the derivative with respect to $\mathbf{a}$, and set the result equal to zero, as was done above. This leads to the our solution:

$$\mathbf{f}_{est} = \left(\mathbf{H}^T\mathbf{C}_{\mathbf{n}}^{-1}\mathbf{H} + \mathbf{C}_{\boldsymbol{f}}^{-1}\right)^{-1}\mathbf{H}^T\mathbf{C}_{\mathbf{n}}^{-1}\mathbf{z}_{obs}. \tag{2.58}$$

A known mean can also be included in this equation as

$$\mathbf{f}_{est} = \mathbf{m}_{\boldsymbol{f}} + \left(\mathbf{H}^T\mathbf{C}_{\mathbf{n}}^{-1}\mathbf{H} + \mathbf{C}_{\boldsymbol{f}}^{-1}\right)^{-1}\mathbf{H}^T\mathbf{C}_{\mathbf{n}}^{-1}\left(\mathbf{z}_{obs} - \mathbf{Hm}_{\boldsymbol{f}}\right). \tag{2.59}$$

This solution is equivalent to Equation 2.41. Which equation to use depends on the size of the vectors $\mathbf{f}$ and $\mathbf{z}$. Letting M be the length of the $\mathbf{z}$ and N be the length of the $\mathbf{f}$, Daniel (1997) shows that 2.41 is more efficiently solved when $M << N$. 2.59 is more efficient when $N << M$. This will be the case in chapters 3 and 4 where multiple data sets are simultaneously inverted.

The framework of Method 1 can be used to derive linear solutions to nonlinear problems, but, in general, this tends to be intractable and suboptimal. Method 2, on the other hand, extends easily to nonlinear forward operators. All that is needed is a

PDF defined on the data.

Chapter 3

3-D geostatistically constrained seismic deconvolution[†]

In this chapter we simultaneously solve two inverse problems: deconvolution and interpolation. The specific application is one useful in reservoir characterization: integrating poststack seismic data with available well logs for an optimal 3-D image of subsurface reflectivity.

3.1 Introduction

Seismic deconvolution has been used to improve temporal resolution of seismic data for the last 50 years (Robinson, 1954; Robinson and Trietel, 1980; Saggaf and Robinson, 2000). Typical deconvolution methods operate on a seismic volume trace-by-trace. Lateral consistency between adjacent traces is usually not enforced or only done so in ad-hoc manners. Lavielle (1991), Idier and Goussard (1993), and Kaaresen and Taxt (1998) all have used a Bernoulli-Gaussian probability distribution to model the horizontal correlation seen in geological structures. They then posed multitrace deconvolution as a Bayesian inverse problem and used this probabilistic model as a prior. Their choice of a prior, however, requires computationally intensive Monte

[†] This chapter was submitted to *Geophysics*.

Carlo algorithms to estimate a solution. Also, they did not incorporate well data in their inversion schemes.

Geostatistics is used to analyze the multi-dimensional correlation structure of a petrophysical parameter field. This information can then be used to interpolate isolated measurements of the parameters, such as well data, and provide error estimates on the interpolated values. Geostatistical interpolation usually goes by the name of *kriging*. Examples of kriging applied to well data abound (Gorell, 1995). *Cokriging* is another geostatistical technique that allows the incorporation of secondary variables as constraints on the estimation of a primary variable by assuming a statistical relationship between the two variables. Doyen (1988) used cokriging to extrapolate porosity in wells constrained to seismic data. Xu et al. (1992) used it to combine well and seismic data for mapping the top of a geologic structure. Although geostatistical methods make use of seismic and well data, they do not incorporate physical relationships between the seismic data and the estimated parameter. Some workers have attempted to address this problem. Haas and Dubrule (1994) provide a geostatistically based method for estimating a field of acoustic impedance from poststack seismic data by sequentially kriging well data and performing 1-D seismic inversion. Most notably, Eide et al. (2002) present a unified statistical methodology for integrating various well data with poststack seismic data by casting it in a Bayesian framework. They concentrate, however, on sampling the posterior rather than maximizing it, and are constrained by the computational cost of Monte Carlo sampling methods.

This chapter presents a joint linear inversion method that combines both deconvolution and geostatistical interpolation in a single framework. The statistical formulation is similar to that Eide et al. (2002) but provides the maximum of the posterior probability function, rather than samples of it. Being linear, it is computationally more efficient than Monte Carlo methods. Applying this method results in a statistically optimal estimate of 2-D or 3-D reflectivity constrained by both well and seismic data. Depending on the setting of adjustable parameters we can arrive at kriging, damped trace-by-trace deconvolution, laterally smoothed deconvolution, and joint deconvolution/kriging as special cases.

We begin by first describing the seismic and well data we use. We follow this with a concise presentation of the inversion method we use to combine these data sets. We then elaborate on how to set adjustable parameters in the inversion. We next discuss the estimation of a seismic wavelet from the data, a step necessary before any deconvolution is applied. Finally we apply the joint inversion method to the field data sets and discuss the results.

3.2 The field data

In this work we will use both a 2-D and 3-D seismic data set, each cut out of the same larger volume. The seismic data is a RAP-DMO migrated poststack volume with a sampling interval of 4 ms. For a detailed description of the this data set and its geological environment see Al-Moqbel et al. (2002).

Poststack data ideally means that previous processing (such as migration and stacking) has put the seismic data in a form such that the 1-D convolutional seismic model (Sengbush et al., 1960) holds. Thus our observed seismic data is assumed to be the result of a one dimensional linear operation (a convolution) of a seismic wavelet on a reflectivity sequence.

We plot the 3-D subvolume in Figure 3-1. The negative is plotted because we observe that positive values of reflectivity in the wells correspond to negative values in the seismic data. Plotting the negative will facilitate easier comparison to reflectivity obtained from inversion. The blank area at the top of the data is due to a later flattening of the seismic data along a horizon. The flattening pushes some of the seismic data out of the cube which is then not used when we perform inversion; therefore, we do not show it here. The flattened 3-D subvolume is shown in Figure 3-2. The flattening is done along a horizon corresponding to a known reservoir. After performing inversion the resulting reflectivity field is subsequently unflattened for comparison with the unflattened seismic.

The seismic subvolume includes 7 wells as illustrated in Figure 3-3. The 2-D seismic section intersects the 3-D subvolume at crossline 164 and contains one well

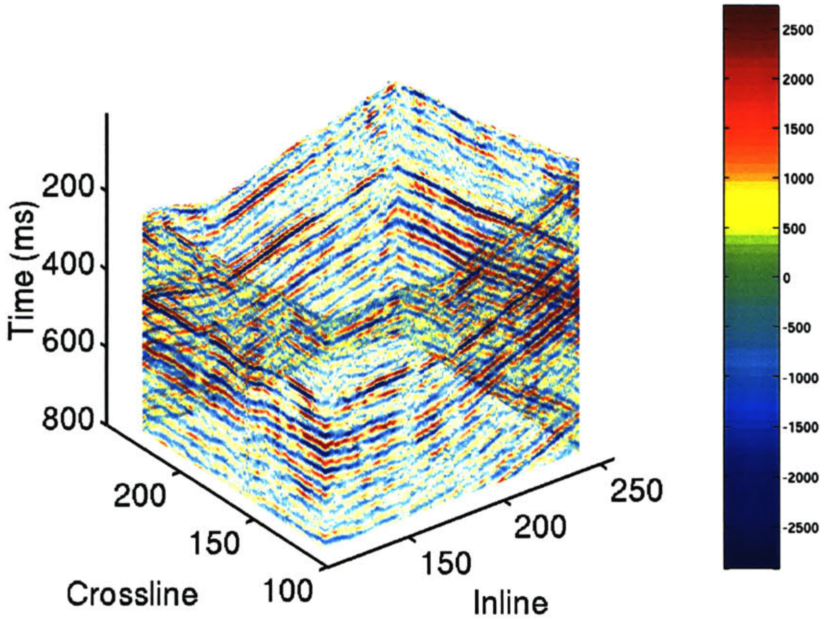

Figure 3-1: 3-D subvolume of field seismic data

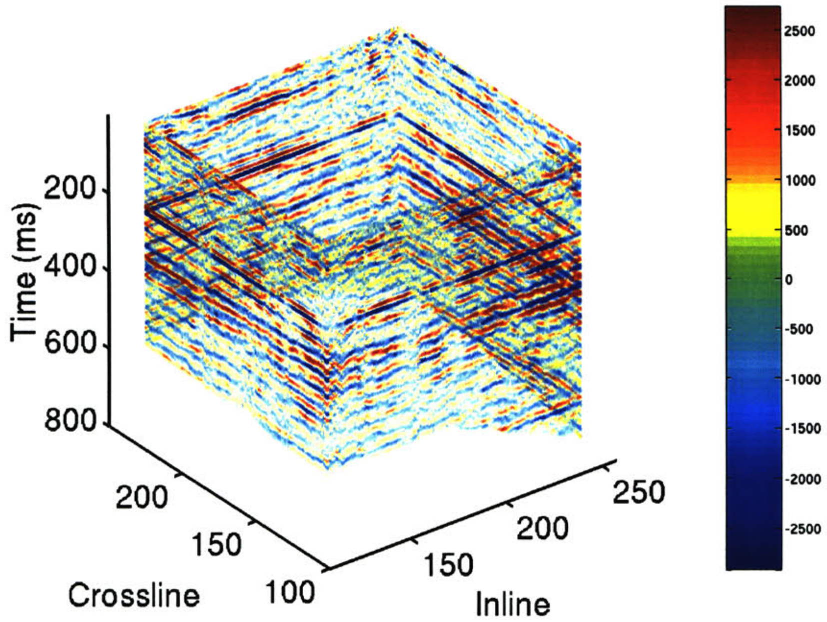

Figure 3-2: Flattened 3-D seismic data.

log at inline 211 that goes from 272 ms to 560 ms. The reflectivity profiles at the

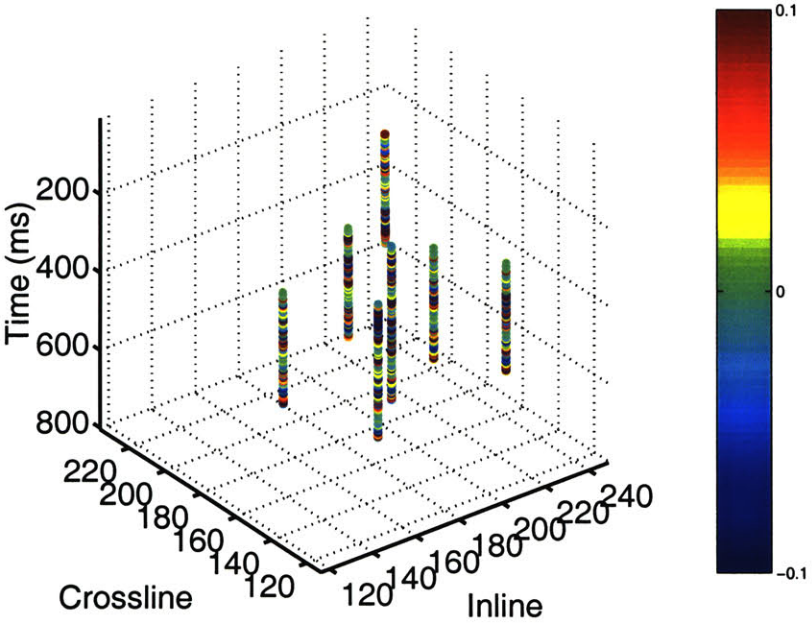

Figure 3-3: Reflectivity well data in unflattened 3-D subvolume.

well locations are obtained by taking existing velocity and density well logs measured in depth and converting them into reflectivity sequences measured in time. This reflectivity well data is correspondingly adjusted in time when the seismic data is flattened. Adjusting the logs with the flattened seismic data allows them to be interpolated along linear trends. Standard geostatistical interpolation techniques require this. Flattening the seismic data provides a simple solution to the problem of modeling non-horizontal layering in the Earth. No complicated statistical prior models like those used in Lavielle (1991) are required in the inversion below. Subsequent unflattening then restores the nonlinear structure to the estimated reflectivity field. This might present a problem if pinch-outs or faults are present but these were not observed in our seismic data.

3.3 Joint inversion

We present here a condensed version of the joint inversion methodology presented in Chapter 2. We refer the reader to that source for theoretical and computational details.

We describe the relationship between observed seismic and reflectivity by the following linear equation:

$$\mathbf{s} = \mathbf{Hf} + \mathbf{n_H}, \tag{3.1}$$

where the discrete vector $\mathbf{f}$ represents either a 2-D or 3-D reflectivity field, $\mathbf{s}$ represents the 2-D or 3-D field of seismic data, and $\mathbf{H}$ is a seismic convolution operator that takes each reflectivity trace measured in time and convolves it with a seismic wavelet $\mathbf{h}$. $\mathbf{n_H}$ represents inaccuracies in the operator $\mathbf{H}$, errors in previous processing steps, and noise in the data collection system. We assume $\mathbf{n_H}$ to be a vector of Gaussian white noise with variance $\sigma^2_{\mathbf{H}}$. Solving Equation 4.1 for $\mathbf{f}$ is ill-posed (Tikhonov and Arsenin, 1977). Without constraints, the solution may be highly contaminated with noise.

Another linear operator relates observed reflectivity well data, $\mathbf{d}$, to the reflectivity field:

$$\mathbf{d} = \mathbf{Pf} + \mathbf{n_P}. \tag{3.2}$$

$\mathbf{P}$ is the "picking" operator that samples the field at measurement locations. In this discrete linear equation it has the form of a subset of rows of the identity matrix. $\mathbf{P}$ is not an invertible operator and no geostatistical interpolation can be done unless other constraints are imposed. $\mathbf{n_P}$ is Gaussian white noise vector with variance $\sigma^2_{\mathbf{P}}$.

Likelihood functions (Tarantola, 1987) can be defined for both the seismic and well data through Equations 4.1 and 4.3, and the Gaussian probability distributions of the noise terms, $\mathbf{n_H}$ and $\mathbf{n_P}$:

$$Lk(\mathbf{s}|\mathbf{f}) = \frac{1}{(2\pi\sigma^2_{\mathbf{H}})^{\frac{N_{\mathbf{s}}}{2}}} \exp\left[-\frac{1}{2\sigma^2_{\mathbf{H}}}(\mathbf{s}-\mathbf{Hf})^T(\mathbf{s}-\mathbf{Hf})\right], \tag{3.3}$$

$$Lk(\mathbf{d}|\mathbf{f}) = \frac{1}{(2\pi\sigma^2_{\mathbf{P}})^{\frac{N_{\mathbf{d}}}{2}}} \exp\left[-\frac{1}{2\sigma^2_{\mathbf{P}}}(\mathbf{d}-\mathbf{Pf})^T(\mathbf{d}-\mathbf{Pf})\right]. \tag{3.4}$$

These functions are probability distributions only for fixed $\mathbf{f}$.

The solution of a maximum likelihood inverse problem involves multiplying together equations 3.3 and 3.4 to get

$$Lk(\mathbf{f}|\mathbf{s},\mathbf{d}) = k'\ Lk(\mathbf{s}|\mathbf{f})Lk(\mathbf{d}|\mathbf{f}), \tag{3.5}$$

and then maximizing over $\mathbf{f}$. k' is a normalization constant. Depending on the form of equation 3.5, there may be more than one vector $\mathbf{f}$ that maximizes, or the maximum might be highly influenced by the noise. In order to overcome the ill-posedness of such an inverse problem, we define an a-priori Gaussian probability distribution on $\mathbf{f}$:

$$Pr(\mathbf{f}) = \frac{1}{(2\pi^{\frac{N_\mathbf{f}}{2}})\ \sqrt{\det((\mathbf{L}^T\mathbf{L})^{-1})}}\ \exp\left[-\frac{1}{2}(\mathbf{f}^T\mathbf{L}^T\mathbf{L}\mathbf{f})\right]. \tag{3.6}$$

The effect of this term on the deconvolution will be to damp out the noise in the solution as well as to impose horizontal geostatistical correlation on the resulting reflectivity field. Using this term in conjunction with the picking operator allows for kriging. We note that kriging is traditionally not posed in this manner. It is usually defined in the minimum variance framework (Deutsch and Journel, 1998). We use the maximum likelihood framework for computational reasons.

Including Equation 3.6 in the product of Equation 3.5, we arrive at the posterior probability density function (Tarantola, 1987):

$$Pr(\mathbf{f}|\mathbf{s},\mathbf{d}) = k\ Lk(\mathbf{s}|\mathbf{f})Lk(\mathbf{d}|\mathbf{f})Pr(\mathbf{f}), \tag{3.7}$$

where k, again, is a normalizing constant.

Equation 3.7 is a joint Gaussian probability density function with the following exponent:

$$-\frac{1}{2}\left(\frac{1}{\sigma_\mathbf{H}^2}(\mathbf{s}-\mathbf{H}\mathbf{f})^T(\mathbf{s}-\mathbf{H}\mathbf{f}) + \frac{1}{\sigma_\mathbf{P}^2}(\mathbf{d}-\mathbf{P}\mathbf{f})^T(\mathbf{d}-\mathbf{P}\mathbf{f}) + \frac{1}{\sigma_\mathbf{L}^2}\mathbf{f}^T\mathbf{L}^T\mathbf{L}\mathbf{f}\right). \tag{3.8}$$

Maximizing Equation 3.7 involves minimizing the negative of this exponent. Taking the derivative of Equation 3.8 with respect to **f** and setting the result equal to 0 leads to the joint normal equations:

$$\underbrace{(\frac{1}{\sigma_{\mathbf{H}}^2}\mathbf{H}^T\mathbf{H} + \frac{1}{\sigma_{\mathbf{P}}^2}\mathbf{P}^T\mathbf{P} + \frac{1}{\sigma_{\mathbf{L}}^2}\mathbf{L}^T\mathbf{L})}_{\mathbf{A}}\,\mathbf{f} = \underbrace{\frac{1}{\sigma_{\mathbf{H}}^2}\mathbf{H}^T\mathbf{s} + \frac{1}{\sigma_{\mathbf{P}}^2}\mathbf{P}^T\mathbf{d}}_{\mathbf{b}}\,. \tag{3.9}$$

Solving this equation for **f** gives the vector that maximizes the posterior probability. $\sigma_{\mathbf{H}}^2$ expresses the degree of uncertainty in the seismic data and $\sigma_{\mathbf{P}}^2$ expresses the degree of uncertainty in the observed well data. $\sigma_{\mathbf{L}}^2$ can be interpreted as saying how much smoothness we wish to impose on the resulting reflectivity field.

To solve our joint inverse problem we must set the σ^2's and numerically perform the equivalent of

$$\mathbf{f} = \mathbf{A}^{-1}\mathbf{b}. \tag{3.10}$$

The size of the hypothetical **A** matrix precludes all but a few methods of solution. The matrix would be too large to store on a computer, let alone invert. It is very sparse, though, and can be operated very efficiently. This leads us to choose an iterative method of solution. The method we use is the conjugate gradients algorithm (Schneider, 2000).

3.3.1 Setting L, a, and the σ^2's

The matrix **L** in Equation 3.6 is related to the covariance matrix, **C**, traditionally used in geostatistics. In fact, $\mathbf{C} = (\mathbf{L}^T\mathbf{L})^{-1}$. **L** usually has the form of a differencing operator. To make a full connection to geostatistics we need to define an **L** that has a well defined associated covariance matrix used by geostatisticians. In 1-D such a matrix exists. We define

$$\mathbf{L} = \frac{\sqrt{a}}{\sigma_{\mathbf{L}}\sqrt{2}}\left(\frac{1}{a}\mathbf{I} - \mathbf{D}\right), \tag{3.11}$$

where $\mathbf{D}$ is a first (forward) differencing matrix and $\mathbf{I}$ is the identity matrix. In the limit as the discretization length goes to zero, $\mathbf{L}^T\mathbf{L}$ is the inverse of the exponential covariance operator:

$$C(x, x') = \sigma_{\mathbf{L}}^2 e^{\frac{|x-x'|}{a}}, \tag{3.12}$$

where a is the correlation length of the random field. In this work we define an $\mathbf{L}$ in two horizontal directions. Unfortunately this does not lead to the 2-D exponential covariance function. It does, however, give an associated covariance function with a correlation length that is controlled by the a parameter. The inversion results below give evidence of this.

We define covariance structure only in horizontal directions because we expect correlated geological structures in those directions. From the seismic data we empirically estimate a correlation length of $a = 200$ in each horizontal direction. We assume that this length is indicative of the correlation length of underlying reflectivity field. We impose no correlation structure on reflectivity in the vertical direction.

In order to estimate the seismic noise level, $\sigma_{\mathbf{H}}^2$, we use the amplitude spectrum of a seismic trace at high frequencies. We assume that this energy in the seismic data at high frequencies is the noise in the system (see Figure 3-5). Inverse Fourier transforming the power spectrum of the noise gives the covariance function. The value of the covariance function at zero lag is $\sigma_{\mathbf{H}}^2$. By this method we obtain the value $\sigma_{\mathbf{H}}^2 = 1.05 \times 10^6$ from the seismic data. This value is used in the inversion described below. When performing only kriging we set $\sigma_{\mathbf{H}}^2 = \infty$ in order to disregard the seismic data.

It can be safely assumed that the well data is of much higher accuracy than the seismic data. Therefore $\sigma_{\mathbf{P}}^2$ should be much smaller than $\sigma_{\mathbf{H}}^2$. As $\sigma_{\mathbf{P}}^2 \mapsto 0$ we have noise free well data. In this work we set $\sigma_{\mathbf{P}}^2 = 7.65 \times 10^{-7}$ for most results which, for all practical purposes, says that the well data should be exactly reproduced in the solution of the joint inverse problem. If we don't wish to use the well data in the inversion, we simply set $\sigma_{\mathbf{P}}^2 = \infty$.

We treat $\sigma_{\mathbf{L}}^2$ as an adjustable parameter. It represents how much prior information is needed to give us a desirable result. We calculate it by trial and error for each of the different joint inversion results below.

3.3.2 Generating a seismic wavelet

It is useful to extract a seismic trace at a location where there also exists reflectivity well data for a visual comparison. This is done for the well data at crossline 211 in the 2-D section, shown with its corresponding negative seismic trace in Figure 3-4. The scales of the reflectivity and seismic data differ by several orders of magnitude so, to visually compare them, the amplitude of the reflectivity data is multiplied by 12,000. Plotting the negative of the seismic versus the reflectivity in Figure 3-4 allows us to see the close match between major events in the log and the seismic trace. This match is not exact though. In addition to the polarity reversal and scale difference of the two data sets, we also see that they appear to have different frequency content (the seismic is lower frequency compared to the reflectivity). Assuming convolution with a wavelet as the operator that changes reflectivity into seismic data, we must

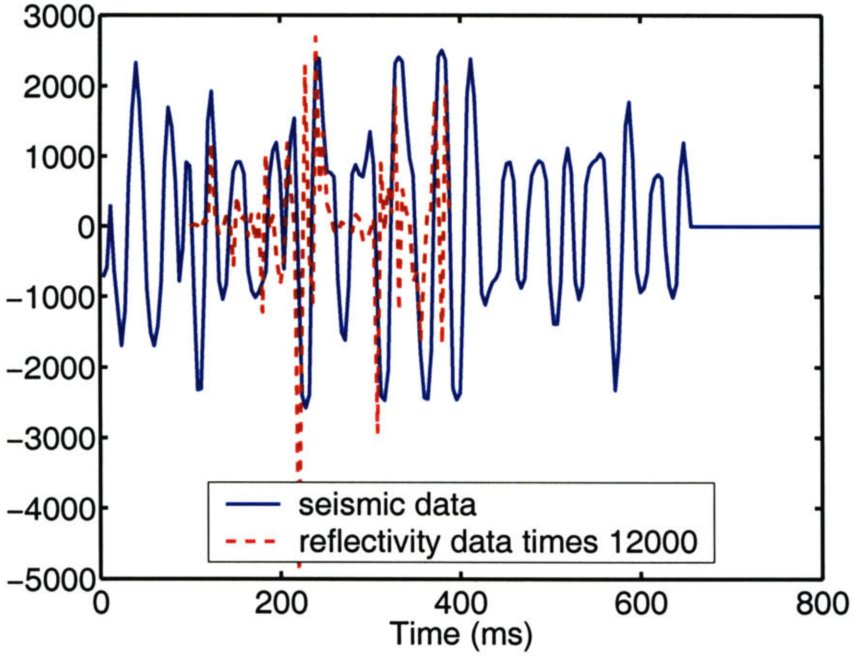

Figure 3-4: Seismic trace and reflectivity from well log.

estimate the wavelet.

In seismic analysis, different methods are used for estimating a wavelet (Robinson and Trietel, 1980). To simplify our wavelet estimation we assume that the seismic wavelet is a zero phase Ricker wavelet, and try to find its best fitting central frequency value. To do this it is useful to look at both reflectivity and seismic data at the same well location in the Fourier domain. In Figure 3-5 we plot the amplitude spectra of the seismic data from Figure 3-4 with a solid line. We take the amplitude spectra of Ricker wavelets of varying central frequency and multiply them by the amplitude spectra of the reflectivity until a good is fit is found to the seismic data. The best fitting result is shown in Figure 3-5 as the dashed line for a Ricker wavelet with center frequency of $28Hz$. The amplitude spectrum of the best fitting Ricker wavelet is shown along with the reflectivity amplitude spectrum in Figure 3-6. Inverse transforming the spectrum of the Ricker wavelet gives the zero phase wavelet itself, shown in Figure 3-7. It has negative polarity because of the polarity relationship between the seismic and reflectivity data.

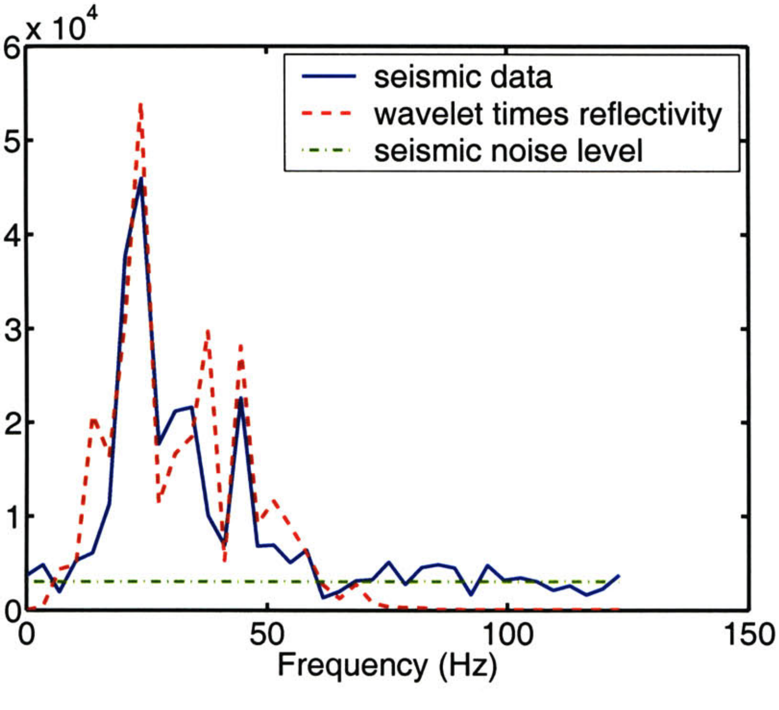

Figure 3-5: Amplitude spectra of seismic trace, best fitting wavelet convolved with reflectivity in well, and noise.

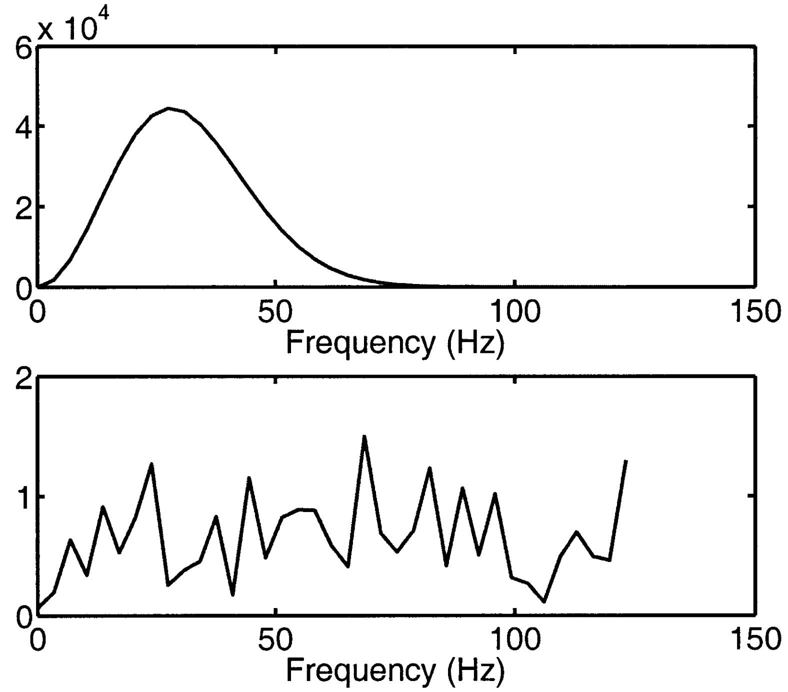

Figure 3-6: Amplitude spectra of best fitting Ricker wavelet (top), and reflectivity in well (bottom).

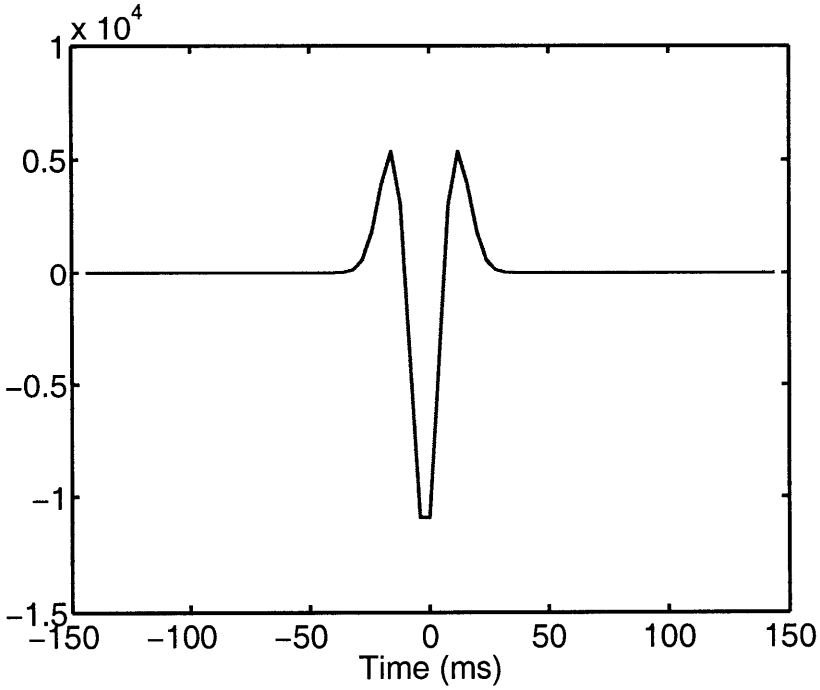

Figure 3-7: Ricker wavelet in time.

When we reconvolve our chosen wavelet from Figure 3-7 with the reflectivity shown in Figure 3-4, we do not reproduce the observed seismic trace exactly. This is shown in Figure 3-8. This highlights the inherent inaccuracy in our seismic convolutional model expressed by the $\mathbf{n_H}$ vector in Equation 4.1.

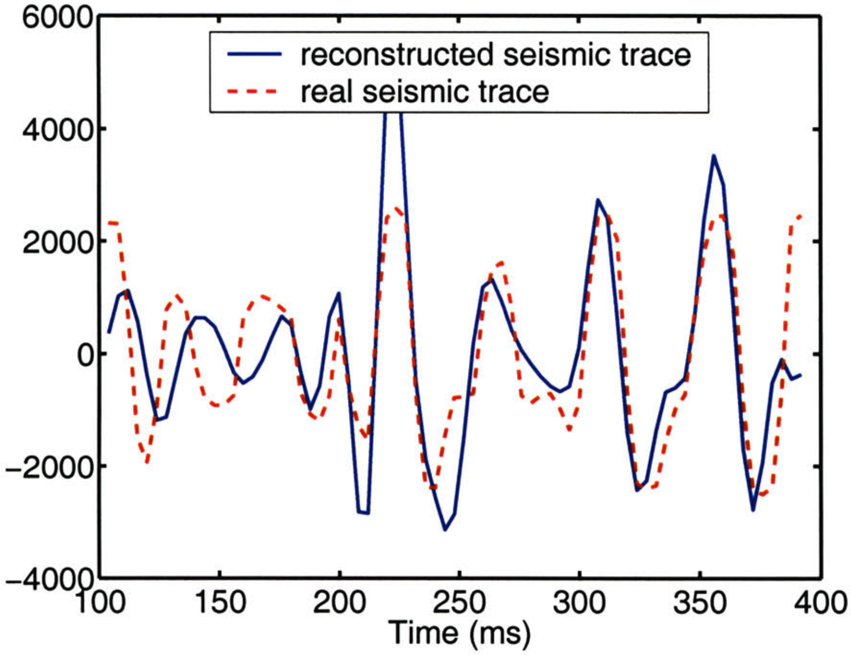

Figure 3-8: Reconvolved wavelet with reflectivity and true seismic trace.

3.4 Application to field data

We show the negative of the 2-D seismic data in Figure 3-9, and an alternate plot of the negative 3-D subvolume in Figure 3-10. Although the overall quality of this seismic is good, it was observed after joint inversion (below) that the seismic data needed further static corrections. The static errors were clearly observed after increasing the resolution in the inversion. Using a preliminary 2-D inversion run, statics corrections were manually calculated and then applied to the original 2-D seismic data. This is shown in Figure 3-11. One should compare Figures 3-9 and 3-11 to see the improvement from the static correction. The corrected 2-D seismic data set is used in all the solutions to the 2-D inverse problem below.

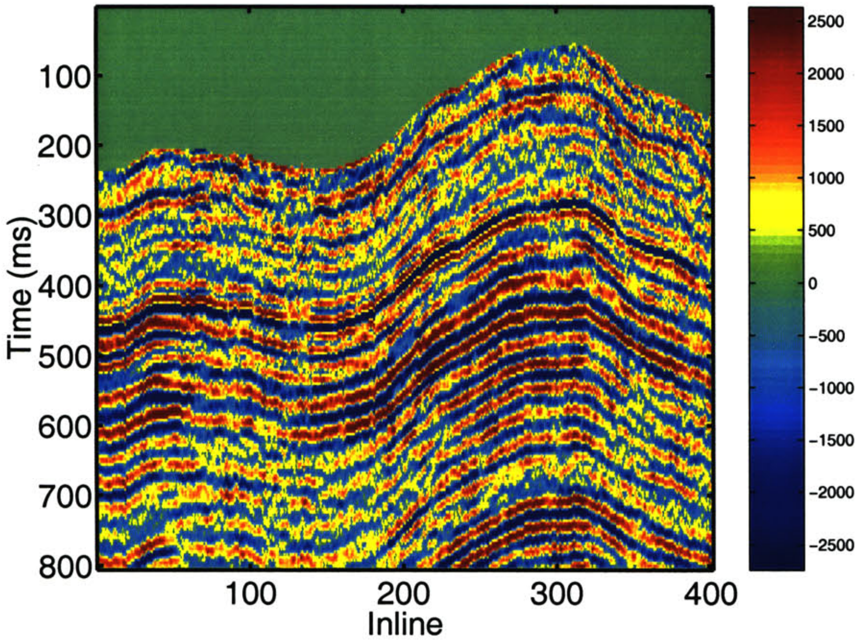

Figure 3-9: 2-D slice of field seismic data

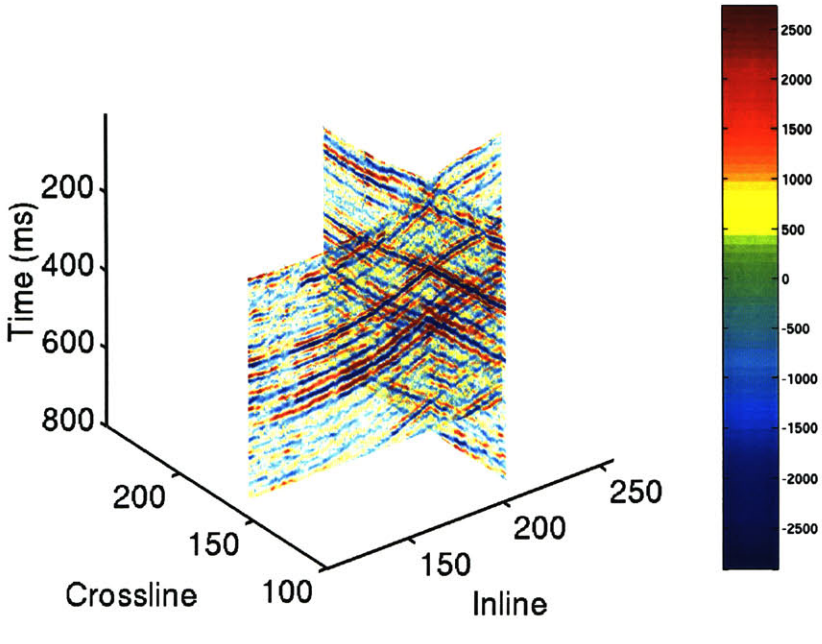

Figure 3-10: Alternate plot of 3-D seismic subvolume.

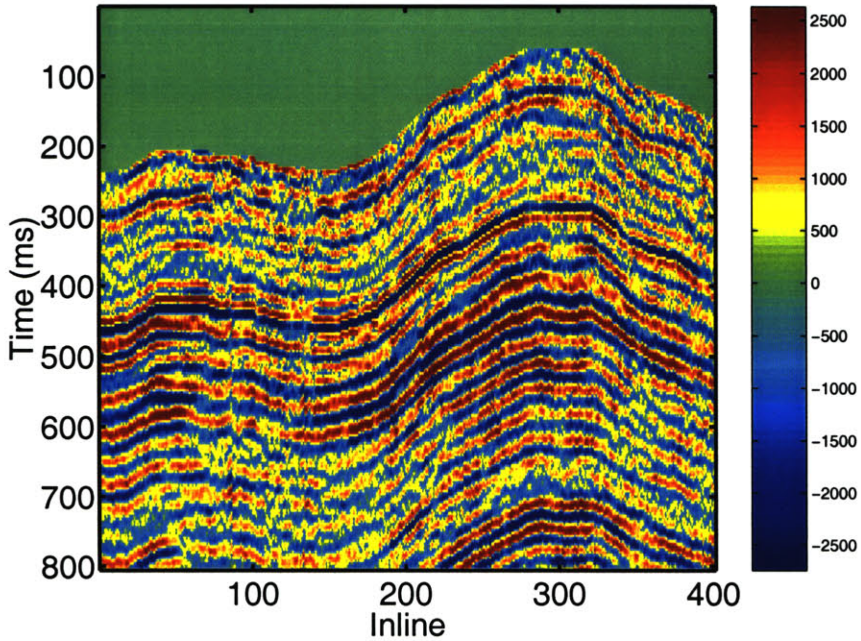

Figure 3-11: Static corrected 2-D seismic data.

We now examine the results of the joint inversion on the field data. Inversion is done for 4 sets of σ^2 parameters. The combination of parameters are shown in Table 3.1 and illustrate 4 important special cases of the inversion: kriging (with no use of seismic data), damped trace-by-trace deconvolution (with no use of well data), laterally smoothed deconvolution (with no use of wells but with geostatistical constraints), and optimal joint deconvolution/kriging (with optimal use of well and seismic data, *and* geostatistical constraints). We see that a different $\sigma^2_{\mathbf{L}}$ is used in 2-D and 3-D. The **L** operator takes horizontal differences in one direction in 2-D and in two directions in 3-D. Hence it is a different operator in 2-D than in 3-D and one is essentially solving a different inverse problem. Therefore we expect the value of $\sigma^2_{\mathbf{L}}$ to change with dimensionality. $\sigma^2_{\mathbf{L}}$ also changes with the a parameter. This can be

Inversion Type	$\sigma^2_{\mathbf{H}}$	$\sigma^2_{\mathbf{P}}$	$\sigma^2_{\mathbf{L}}$ in 2-D	$\sigma^2_{\mathbf{L}}$ in 3-D	a
Kriging Only	∞	7.65×10^{-7}	0.005	0.005	200
Damped Trace-By-Trace Decon.	1.05×10^{6}	∞	0.1	100	1
Laterally Smoothed Decon.	1.05×10^{6}	∞	0.0075	0.05	200
Optimal Joint Decon./Kriging	1.05×10^{6}	7.65×10^{-7}	0.005	0.15	200

Table 3.1: Parameter combinations for 4 different inversion schemes.

seen by looking at the form of the **L** matrix in Equation 3.11. As a changes, the norm of **L** changes also. This affects the solution of Equation 3.9 and explains the order of magnitude difference of $\sigma_{\mathbf{L}}^2$ between damped trace-by-trace deconvolution and the other inversion cases. We re-emphasize here that the optimal $\sigma_{\mathbf{L}}^2$ values for all cases are found by trial and error.

Using the well data shown in Figure 3-3, we perform kriging with the parameters from Table 3.1. The kriging results are shown in Figures 3-12 and 3-13 for 2-D and 3-D, respectively. From these figures we can see that the well data has been extrapolated a large distance because of the correlation length input with the **L** operator. Kriging can tell us nothing, however, about reflectivity at locations above and below the wells. It also cannot account for any variations in reflectivity at inter-well locations.

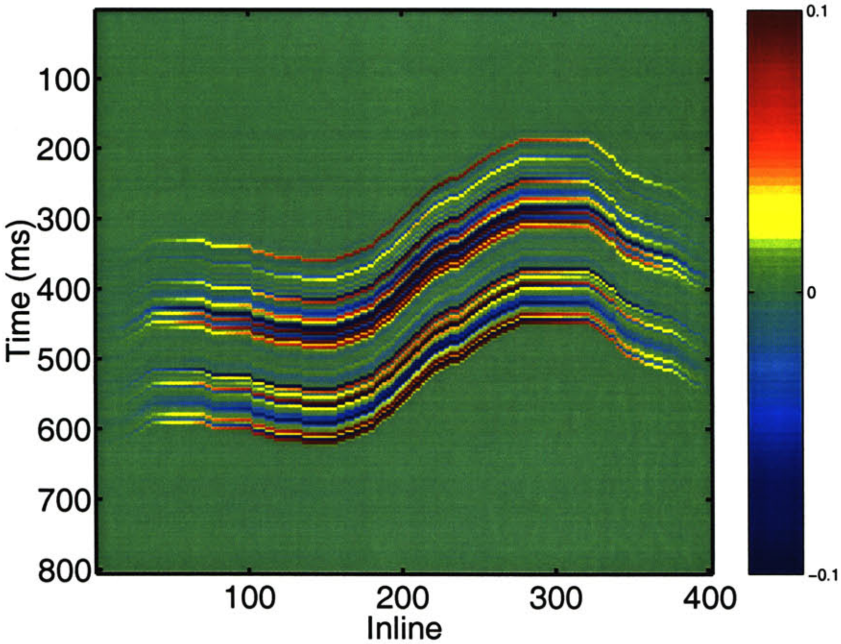

Figure 3-12: 2-D kriged reflectivity field.

Damped trace-by-trace deconvolution makes no use of the well data or geostatistical constraints. It does not force the result of the inversion to fit well data at well locations because $\sigma_{\mathbf{P}}^2$ is infinite. Nor does it enforce much horizontal correlation because $a = 1$. Thus it essentially only performs a vertical damped deconvolution at each seismic trace location. The results of such an inversion are shown in Figures

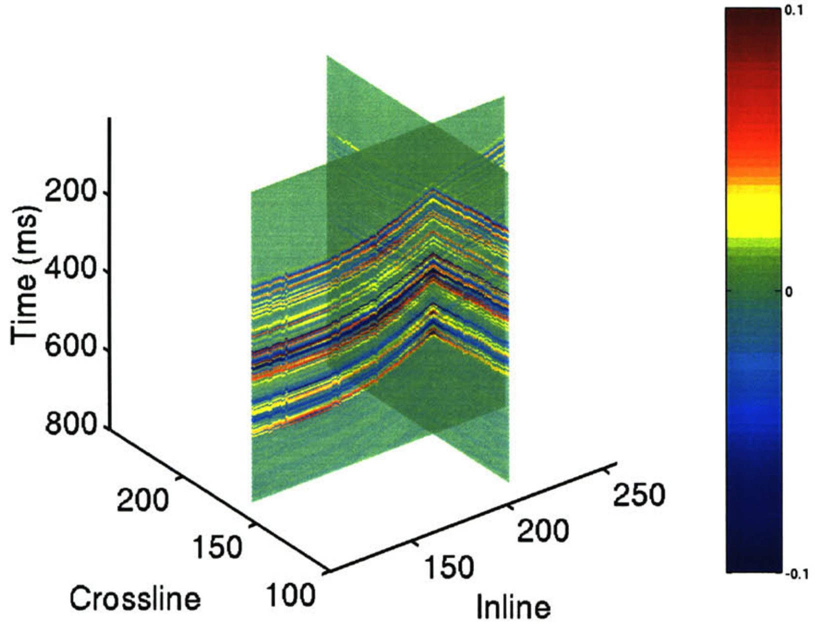

Figure 3-13: 3-D kriged reflectivity field.

3-14 and 3-15 for 2-D and 3-D, respectively. The lateral continuity is as bad as the original seismic but vertically the resolution has been improved because of deconvolution in that direction. We next combine the vertical resolution improvement of damped trace-by-trace deconvolution with the horizontal correlation provided by the **L** matrices.

Laterally smoothed deconvolution enforces horizontal correlation constraints on the estimated reflectivity field but does not make use of well data. We show the results of this in figures 3-16 and 3-17, for 2-D and 3-D respectively. It improves upon the damped trace-by-trace deconvolution by smoothing out the noise and connecting reflectors horizontally while still improving resolution vertically. Incorporating the well data can only improve this result, as we show next.

In Figures 3-18 and 3-19 we see the results of the optimal joint deconvolution/kriging in 2-D and 3-D, respectively. Away from well locations we see that the inversion produces a laterally consistent deconvolution; it sharpens the image and shows greater lateral continuity than the original seismic. Near the wells the inversion makes use of the high resolution well logs and produces a high resolution image of reflectivity not attainable with laterally smoothed deconvolution by itself.

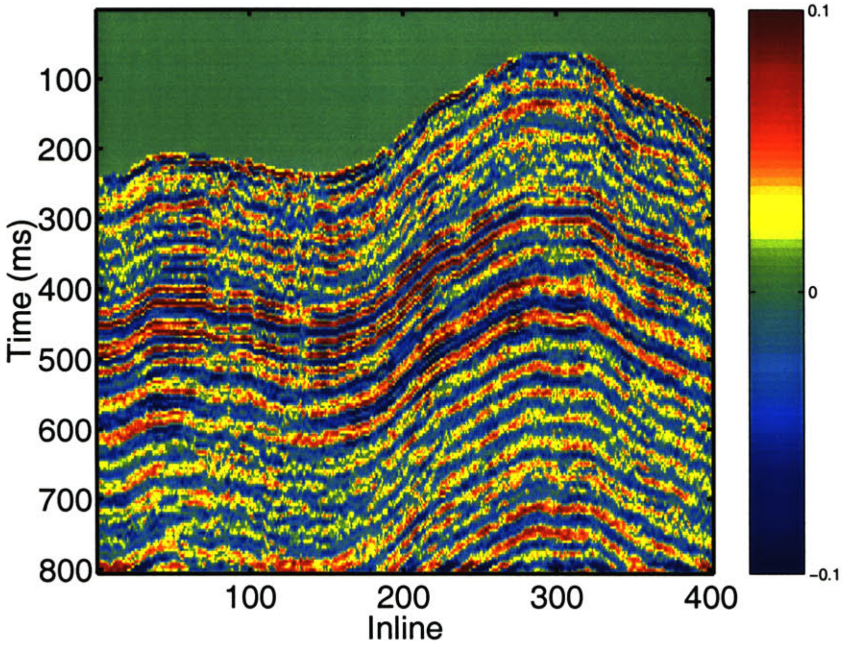

Figure 3-14: Damped trace-by-trace deconvolution of 2-D seismic data.

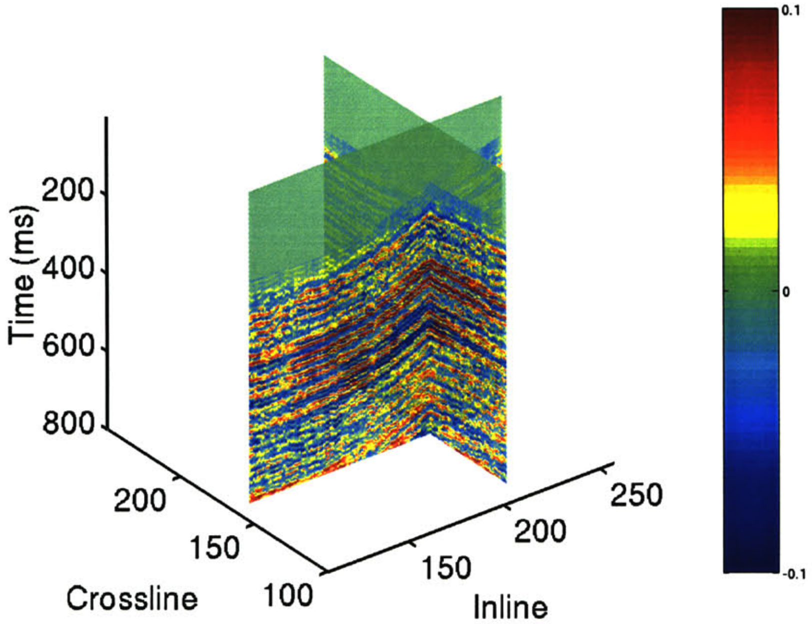

Figure 3-15: Trace-by-trace damped deconvolution of 3-D seismic data.

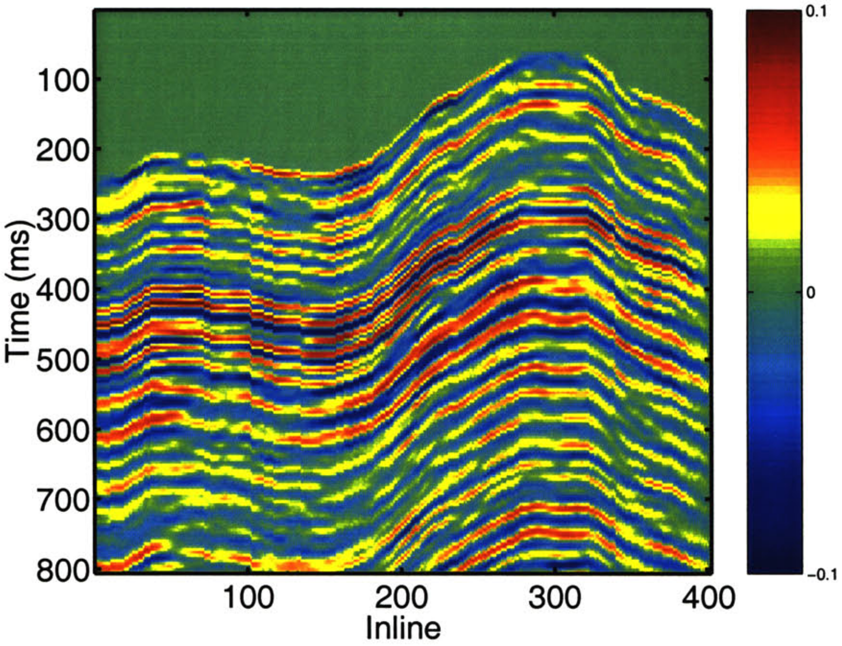

Figure 3-16: Laterally smoothed deconvolution of 2-D seismic data.

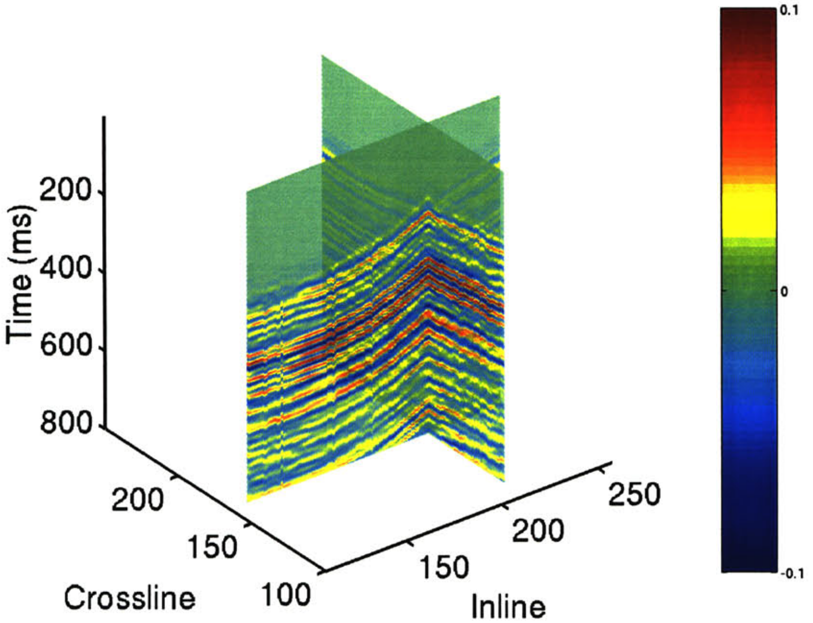

Figure 3-17: Laterally smoothed deconvolution of 3-D seismic data.

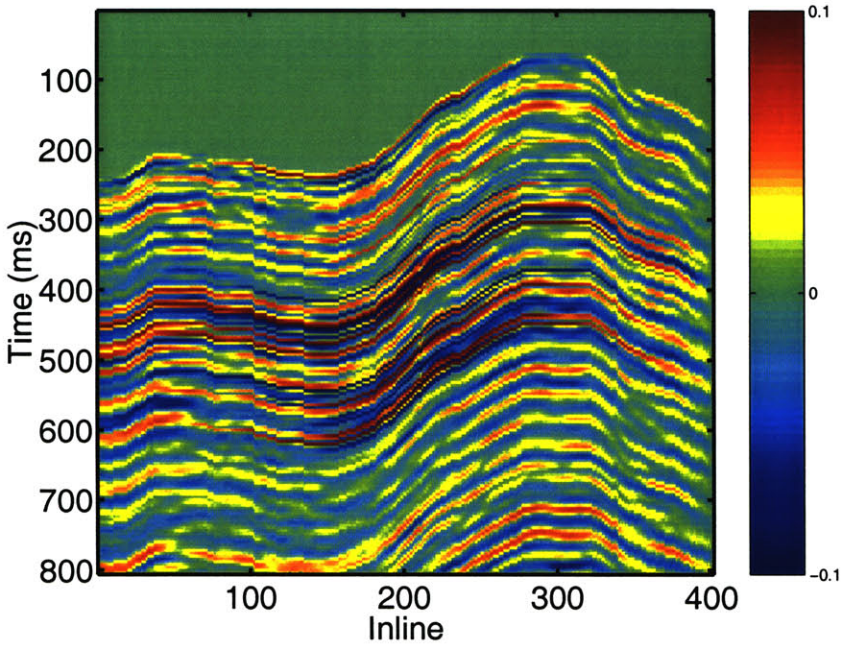

Figure 3-18: 2-D joint deconvolution/kriging of well and seismic data.

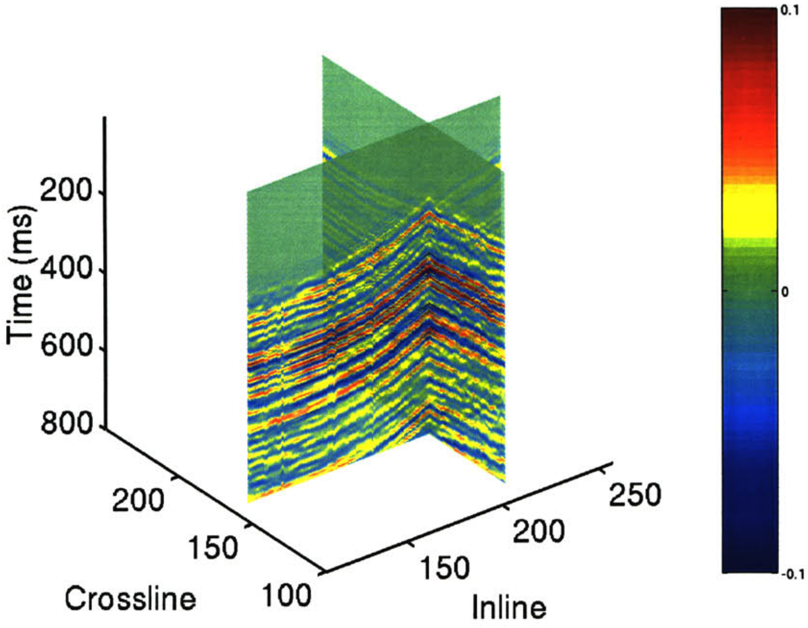

Figure 3-19: 3-D joint deconvolution/kriging of well and seismic data.

The reflectivity fields obtained above by joint inversion can be further inverted for other petrophysical parameters. Al-Moqbel et al. (2002) converts the reflectivity cube shown in Figure 3-19 into an acoustic impedance cube. He does this in a fast, trace-by-trace, fashion via a 1-D nonlinear inverse operator. Since correlation between adjacent reflectivity traces has already been imposed by the joint inversion process, the resulting impedance cube also exhibits horizontal correlation. This way of estimating impedance is fast because it breaks the problem into two steps: a 3-D linear inverse problem, and then a sequence of fast 1-D nonlinear operations.

3.5 Conclusions

In this chapter we applied a methodology that jointly deconvolves poststack seismic data while interpolating existing well data. By posing both deconvolution and kriging as a single joint inversion we could arrive at several different solutions as special cases including kriging, damped trace-by-trace deconvolution, laterally smoothed deconvolution, and joint deconvolution/kriging.

The kriging result simply extrapolated the available well data without constraining the estimate to the seismic data. Therefore it could not account for heterogeneity in the reflectivity field away from the wells. The damped trace-by-trace deconvolution greatly enhanced the vertical (temporal) resolution of the seismic, but resulted in a horizontally discontinuous field. This problem was corrected in the laterally smoothed deconvolution where geostatistical correlation constraints were imposed on the resulting reflectivity. Finally, by incorporating both the geostatistical constraints and well data along with the seismic, we arrived at joint deconvolution/kriging. It provided the best estimated field by using the well data to increase the resolution of the deconvolution near the wells and the seismic data to aid the extrapolation away from the wells. We summarize the results of all 4 inversion cases in Figure 3-20 for comparison.

There are a couple of obvious improvements that could be made on these results. We did not account for the vertical correlation structure of the reflectivity field. A

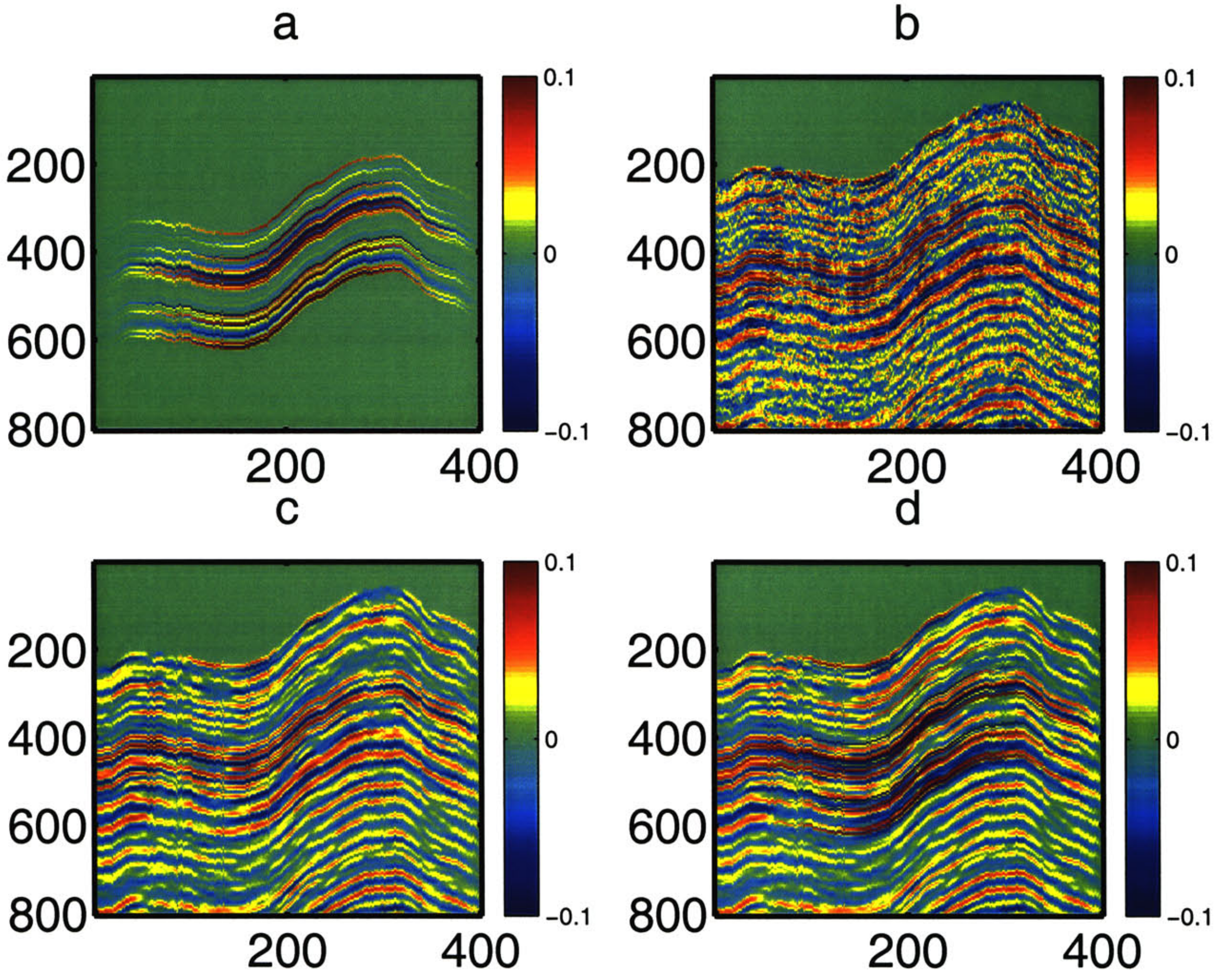

Figure 3-20: Inversion results for a) kriging, b) trace-by-trace deconvolution, c) laterally smoothed deconvolution, and d) joint deconvolution/kriging.

great deal of research has gone into this in recent years (Saggaf and Robinson, 2000) and the stochastic reflectivity models of other workers could be easily incorporated. Another improvement that could be made would be to calculate the estimation error variance of the estimated reflectivity field. There exist methods to do this in the same maximum likelihood framework using the conjugate gradients algorithm (Schneider, 2000).

3.6 Acknowledgments

We thank the ERL Founding Members Consortium for its support of this project.

Chapter 4

Joint deconvolution and interpolation of remote sensing data[††]

This chapter represents a different application of the same joint inversion methodology used in Chapter 3. We again combine accurate, but sparse, point samples of a field, with global, but low resolution and blurred, observations of the same field. The data sets involved in this case are digital elevation model (DEM) data and global positioning system (GPS) data. In practice the former is obtained via satellite measurements over large areas but is of low resolution, while the latter data set is of higher accuracy but only available at sparse locations. Using a real topographic model, synthetic DEM and GPS data are generated and a problem is constructed to show the superiority of the joint inversion method.

4.1 Introduction

Inverse problems can involve estimating a field of parameters given multiple data sets related to that field by different operators. These data sets might be of different

[††] This chapter to be submitted to *IEEE Transactions on Geoscience and Remote Sensing.*

resolution and provide different, but complementary, information about the parameter field of interest. It is intuitive that a combination of multiple data sets should result in a superior estimated field than by using just one of them. How to optimally combine these different data sets, however, is not necessarily a straight-forward operation.

In remote sensing a common source of data are aerial or satellite based sensors. An example of such a data set is a digital elevation model (DEM), which is usually acquired over a regular grid of discrete points and represents the topographical elevation of the Earth's surface. In many cases this data can be modeled as the result of a convolution of the unknown elevation model with a **point spread function** (PSF), representing the impulse response of the sensor. There is usually also additive noise in such a system. We show an example of such a data set in Figure 4-1. Another source of data representing elevation measurements are those obtained by a global positioning system (GPS). These data are noisy point samples of the elevation field obtained directly at ground sites. They tend to be of much higher accuracy than

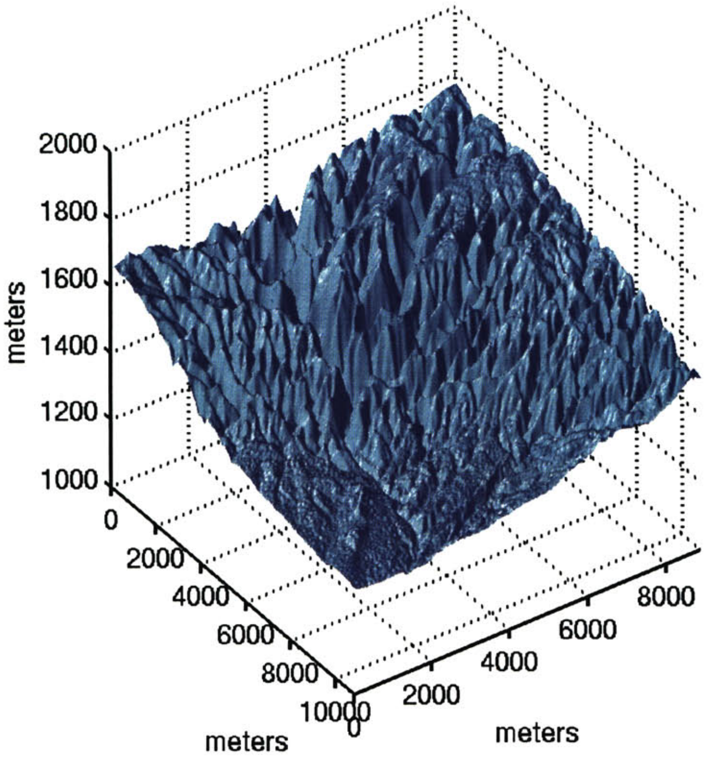

Figure 4-1: Digital elevation model.

DEM data, but sampled sparsely at irregular locations. The technique of inverting the former data set is known as **deconvolution**, a problem that has been well studied in a number of scientific fields over that last 50 years (Wiener, 1949; Robinson, 1954). It involves reversing the blurring effect caused by the PSF while simultaneously damping the noise in the system. Inverting the latter data set is known as either **statistical interpolation** (Leung et al., 2001) or, in the geostatistical community, as **kriging** (Deutsch and Journel, 1998).

By constructing a likelihood function associated with each data set and defining a prior model of the topography field we can use the Bayesian framework to define the joint inverse problem. The solution then becomes the model that maximizes the posterior probability distribution. Finding that model involves solving a very large system of equations. This cannot be done with standard matrix inversion techniques. By making use of the structure of the problem and the conjugate gradients algorithm, we show how the solution can be calculated accurately and efficiently.

The outline is as follows: we first present deconvolution and interpolation and derive likelihood functions associated with both problems. Combined with a prior model, these likelihood functions define the Bayesian maximum a-posteriori (MAP) solution. We next present the conjugate gradients algorithm as a method for overcoming computation difficulties associated with large dimensionality. Joint inversion is then performed on a synthetic data set mimicking DEM and GPS data.

We use the following notation conventions in this chapter:

vectors	$\mathbf{f}, \mathbf{g}, \mathbf{h} \ldots$
scalars	$\mathrm{f}, \mathrm{g}, \mathrm{h}, \ldots$
random vectors	$\boldsymbol{f}, \boldsymbol{h}, f, h \ldots$
random scalars	$f, h, f, h \ldots$
matrix operators	$\mathbf{F}, \mathbf{P}, \mathbf{H}, \boldsymbol{\Lambda}, \ldots.$
functions	$f(\cdot), p(\cdot), \ldots$
probabilistic events	$\mathfrak{a}, \mathfrak{b}, \mathfrak{c}, \ldots$
constants	$L, N, \alpha, \beta \ldots$

4.2 Deconvolution

We will denote our unknown model parameter as $\mathbf{f}$. For the purposes of this chapter it will be a discrete finite dimensional vector on a 2-D Cartesian spatial lattice representing topographical elevation. The PSF will be denoted by $\mathbf{h}$. It is the impulse response of the measuring instrument, and, for our purposes, will be an IIR 2-D filter. We will represent the forward convolution operator as a matrix/vector product:

$$\boldsymbol{s} = \mathbf{Hf} + \boldsymbol{n}, \tag{4.1}$$

where $\mathbf{H}$ is a square circulant convolution matrix. $\boldsymbol{n}$ is a random vector that represents the noise in the data gathering process as well as inaccuracy in the representation of the PSF. We constrain it to be Gaussian with covariance matrix $\mathbf{C}_{\boldsymbol{n}}$. This is done mainly because of the mathematical tractability of Gaussian random vectors and the fact that linear combinations of a large number of finite variance random variables converge to the Gaussian, providing theoretical justification for its usage.

Equation 4.1 implicitly describes the **likelihood function**, $p_{\boldsymbol{s}|\mathbf{f}}(\mathrm{s}|\mathbf{f})$, of the data $\boldsymbol{s}$ given a model $\mathbf{f}$. It is equivalent to the probability density function (PDF) of the noise: $p_{\boldsymbol{s}|\mathbf{f}}(\mathrm{s}|\mathbf{f}) = p_{\boldsymbol{n}}(\mathbf{n}) = p_{\boldsymbol{n}}(\mathrm{s} - \mathbf{Hf})$. Since the noise is Gaussian we have an explicit representation of this likelihood function:

$$p_{\boldsymbol{s}|\mathbf{f}}(\mathrm{s}|\mathbf{f}) = \frac{1}{\left(2\pi^{\frac{N_n}{2}} \ \det(\mathbf{C}_{\boldsymbol{n}})\right)} \ \exp\left[-\frac{1}{2}(\mathrm{s} - \mathbf{Hf})^T \mathbf{C}_{\boldsymbol{n}}^{-1}(\mathrm{s} - \mathbf{Hf})\right], \tag{4.2}$$

where $N_{\boldsymbol{n}}$ is the number of elements of $\boldsymbol{n}$.

If $\mathbf{H}$ is invertible it has a block diagonal inverse, $\mathbf{H}^{-1}$. Furthermore, if the noise, $\boldsymbol{n}$, is not too large, the inverse problem is well-posed and can be solved by maximizing Equation 4.2 over all possible vectors $\mathbf{f}$. This is known as the method of maximum likelihood estimation (Willsky et al., 1999; Rodi, 1989). If $\mathbf{H}$ is not invertible or the noise level is too large, the problem is ill-posed (Tikhonov and Arsenin, 1977) and some other constraint is required for a unique stable solution. In the Bayesian framework this constraint comes in the form of a PDF defined on the object being estimated. We examine the prior PDF below.

4.3 Interpolation

Statistically based interpolation of data is usually derived in the minimum variance framework (Deutsch and Journel, 1998; Leung et al., 2001). We will instead define the interpolation problem in the maximum likelihood framework (Fomel, 2001), as we did for deconvolution. The minimum variance framework leads to the same solution as the maximum likelihood framework, but to a different (yet equivalent) system of equations to solve (Willsky et al., 1999; Rodi, 1989). Which framework to use depends on the number of elements in the data and model vectors and the algorithm used to solve the linear system of equations. It can be shown (Daniel, 1997) that for direct inversion of full matrices the minimum variance solution requires less computations when there are fewer data than model parameters. The opposite is true when the data vector is larger than the model vector. In the joint inverse problem below we have the latter case so we use the maximum likelihood framework.

Posing interpolation as a discrete linear inverse problem in the maximum likelihood framework requires 1) a "picking" operator that picks isolated observations out of a lattice of values, and, 2) a constraint that forces continuity between adjacent estimated values. We examine the first of these requirements in this section and address the second in the next section.

We let $\mathbf{f}$ again be the unknown model. $\boldsymbol{d}$ will represent data extracted by the picking operator, $\mathbf{P}$. $\boldsymbol{d}$ is related to $\mathbf{f}$ via the following equation:

$$\boldsymbol{d} = \mathbf{P}\mathbf{f} + \boldsymbol{n}'. \tag{4.3}$$

$\mathbf{P}$ is a rectangular matrix comprised of zeros with a single value of 1 in each row at the location corresponding to the element of $\mathbf{f}$ that is observed. It can be seen that $\mathbf{P}$ is made up of a subset of the rows of the identity matrix. It "picks" the observed data out of the field $\mathbf{f}$ and adds some noise, $\boldsymbol{n}'$, to make the sampled data. $\boldsymbol{n}'$ is another Gaussian random vector that we assume to be uncorrelated with $\mathbf{f}$ and with covariance matrix $\mathbf{C}_{\boldsymbol{n}'}$.

As in Equation 4.2, we can explicitly write the likelihood function for the sampled data:

$$p_{\boldsymbol{d}|\mathbf{f}}(\mathbf{d}|\mathbf{f}) = \frac{1}{\left(2\pi^{\frac{N_{\boldsymbol{n}'}}{2}} \det(\mathbf{C}_{\boldsymbol{n}'})\right)} \exp\left[-\frac{1}{2}(\mathbf{d}-\mathbf{P}\mathbf{f})^T \mathbf{C}_{\boldsymbol{n}'}^{-1}(\mathbf{d}-\mathbf{P}\mathbf{f})\right], \tag{4.4}$$

where $N_{\boldsymbol{n}'}$ is the number of data in $\boldsymbol{n}'$.

The matrix $\mathbf{P}$ is wider than it is long, thus it does not have a unique inverse. The sampled data cannot be inverted to give inter-sample model estimates unless we add some kind of constraint to the problem. As with the ill-posed deconvolution problem above, this constraint will be a prior PDF defined on $\mathbf{f}$.

4.4 The prior model

In the deconvolution problem, a prior PDF may be needed to make the problem invertible, or to account for excessive noise in the system. In the interpolation problem, a prior PDF is absolutely required if any interpolation is to be done. Besides making inverse problems solvable, a prior PDF tells us something about the continuity of the estimated field.

We will impose a Gaussian PDF over $\mathbf{f}$ to make it a random field, $\boldsymbol{f}$. The Gaussian PDF is sufficient to uniquely specify the mean, variance, and covariance of each element of $\boldsymbol{f}$. The PDF will have the explicit form

$$p_{\boldsymbol{f}}(\mathbf{f}) = \frac{1}{\left(2\pi^{\frac{N_{\mathbf{f}}}{2}} \det(\mathbf{C}_{\boldsymbol{f}})\right)} \exp\left[-\frac{1}{2}\, \mathbf{f}^T\, \mathbf{C}_{\boldsymbol{f}}^{-1}\, \mathbf{f}\right]. \tag{4.5}$$

We established above that the individual elements of $\boldsymbol{f}$ are functions of 2-D space: $f_j = f_j(\mathrm{s}_j)$, where $\mathrm{s}_j = (\mathrm{x}_j, \mathrm{y}_j)$ are 2-D spatial coordinates. We will assume an isotropic exponential covariance structure between the points of $\boldsymbol{f}$. This covariance model is commonly used in geostatistical applications and has the form

$$C(f_j, f_k) = \sigma_{\mathbf{f}}^2 \exp\left[-\frac{||\mathbf{s}_j - \mathbf{s}_k||_{\ell_1}}{L}\right], \tag{4.6}$$

where $\sigma_{\mathbf{f}}^2$ is the variance of the prior random field and L is the isotropic correlation length. Since $\mathbf{C}_{\mathbf{f}}[j, k] = C(f_j, f_k)$, the values of the matrix $\mathbf{C}_{\mathbf{f}}$ can be filled in via Equation 4.6. The stationary structure of this random field ensures that the matrix $\mathbf{C}_{\mathbf{f}}$ will be a convolution matrix. We also force it to be a circular matrix for computational reasons below.

4.5 Joint inversion

The conditional probability of an event $\mathfrak{a}$ occurring given that two other events $\mathfrak{b}$ and $\mathfrak{c}$ have occurred is defined as

$$p(\mathfrak{a}|\mathfrak{b}, \mathfrak{c}) = \frac{p(\mathfrak{a}, \mathfrak{b}, \mathfrak{c})}{p(\mathfrak{b}, \mathfrak{c})}. \tag{4.7}$$

where $p(\cdot)$ is the probability functional returning a value between 0 and 1. Similarly we have

$$p(\mathfrak{b}, \mathfrak{c}|\mathfrak{a}) = \frac{p(\mathfrak{a}, \mathfrak{b}, \mathfrak{c})}{p(\mathfrak{a})}. \tag{4.8}$$

Combining Equations 4.7 and 4.8 we arrive at

$$p(\mathfrak{a}|\mathfrak{b}, \mathfrak{c}) = \frac{p(\mathfrak{b}, \mathfrak{c}|\mathfrak{a})p(\mathfrak{a})}{P(\mathfrak{b}, \mathfrak{c})}. \tag{4.9}$$

If we assume that events $\mathfrak{b}$ and $\mathfrak{c}$ are independent, we further have

$$p(\mathfrak{a}|\mathfrak{b}, \mathfrak{c}) = \frac{p(\mathfrak{b}|\mathfrak{a})p(\mathfrak{c}|\mathfrak{a})p(\mathfrak{a})}{p(\mathfrak{b})p(\mathfrak{c})}. \tag{4.10}$$

If we set the events $\mathfrak{a}$, $\mathfrak{b}$, and $\mathfrak{c}$ equal to the vectors $\mathbf{f}$, s, and $\mathbf{d}$, respectively, we arrive at the Bayesian posterior PDF:

$$p_{\boldsymbol{f}|\mathrm{s},\mathbf{d}}(\mathbf{f}|\mathrm{s},\mathbf{d}) = k\ p_{\boldsymbol{s}|\mathbf{f}}(\mathrm{s}|\mathbf{f})p_{\boldsymbol{d}|\mathbf{f}}(\mathbf{d}|\mathbf{f})p_{\boldsymbol{f}}(\mathbf{f}), \tag{4.11}$$

with $k = 1/(p_{\boldsymbol{s}}(\mathrm{s})p_{\boldsymbol{d}}(\mathbf{d}))$ being simply a re-normalization constant. The Bayesian solution of the joint inverse problem requires finding the vector $\mathbf{f}$ that maximizes Equation 4.11.

Inserting Equations 4.2, 4.4, and 4.5 into Equation 4.11 we have the explicit form of the posterior PDF:

$$p_{\boldsymbol{f}|\mathrm{s},\mathbf{d}}(\mathbf{f}|\mathrm{s},\mathbf{d}) = k\ \exp\left[-\epsilon(\mathbf{f}|\mathrm{s},\mathbf{d})\right], \tag{4.12}$$

where

$$\epsilon(\mathbf{f}|\mathrm{s},\mathbf{d}) = \frac{1}{2}\left((\mathrm{s}-\mathbf{Hf})^T\mathbf{C}_{\boldsymbol{n}}^{-1}(\mathrm{s}-\mathbf{Hf}) + (\mathbf{d}-\mathbf{Pf})^T\mathbf{C}_{\boldsymbol{n}'}^{-1}(\mathbf{d}-\mathbf{Pf}) + \mathbf{f}^T\ \mathbf{C}_{\boldsymbol{f}}^{-1}\ \mathbf{f}\right). \tag{4.13}$$

Maximizing the multi-Gaussian posterior PDF in Equation 4.12 corresponds to minimizing its negative exponent, Equation 4.13. To find the minimum we take the derivative of Equation 4.13 with respect to $\mathbf{f}$ and set the result equal to zero:

$$\frac{\partial\epsilon(\mathbf{f}|\mathrm{s},\mathbf{d})}{\partial\mathbf{f}} = \left(\mathbf{H}^T\mathbf{C}_{\boldsymbol{n}}^{-1}\mathbf{H} + \mathbf{P}^T\mathbf{C}_{\boldsymbol{n}'}^{-1}\mathbf{P} + \mathbf{C}_{\boldsymbol{f}}^{-1}\right)\mathbf{f} - \mathbf{H}^T\mathbf{C}_{\boldsymbol{n}}^{-1}\mathrm{s} - \mathbf{P}^T\mathbf{C}_{\boldsymbol{n}'}^{-1}\mathbf{d} = 0. \tag{4.14}$$

Rearranging, we have

$$\underbrace{\left(\mathbf{H}^T\mathbf{C}_{\boldsymbol{n}}^{-1}\mathbf{H} + \mathbf{P}^T\mathbf{C}_{\boldsymbol{n}'}^{-1}\mathbf{P} + \mathbf{C}_{\boldsymbol{f}}^{-1}\right)}_{\mathbf{A}}\mathbf{f} = \underbrace{\mathbf{H}^T\mathbf{C}_{\boldsymbol{n}}^{-1}\mathrm{s} + \mathbf{P}^T\mathbf{C}_{\boldsymbol{n}'}^{-1}\mathbf{d}}_{\mathbf{b}}. \tag{4.15}$$

Inverting the matrix $\mathbf{A}$ will result in a vector $\mathbf{f}_{est}$ maximizing the posterior probability:

$$\mathbf{f}_{est} = \mathbf{A}^{-1}\mathbf{b}. \tag{4.16}$$

We now examine the computational difficulties of inverting such a matrix.

4.6 Computational issues

If $\boldsymbol{f}$ is a 2-D Cartesian spatial lattice with dimensions $p \times q$, then $\boldsymbol{f}$ has $N_{\boldsymbol{f}} = p \times q$ elements and the matrix $\mathbf{A}$ has $N_{\boldsymbol{f}}^2$ elements. For a problem of realistic size $\mathbf{A}$ may be so large that it cannot be stored in computer RAM, let alone inverted. Standard algorithms for matrix inversion (Strang, 1986; Trefethen and Bau, 1997; Press et al., 1995) are not applicable and we must seek another way of solving Equation 4.15.

Certain iterative methods require the repeated application of the $\mathbf{A}$ matrix to obtain $\mathbf{f}_{est}$. If we can find a fast way to operate $\mathbf{A}$ without explicitly constructing it, we can still solve the problem. We show how this can be done quickly by looking individually at each component of $\mathbf{A}$. The first component of $\mathbf{A}$ is $\mathbf{H}^T\mathbf{C}_{\boldsymbol{n}}^{-1}\mathbf{H}$. Since $\mathbf{H}$ is a circulant convolution matrix, it is diagonalized by the Fourier transform. If $\boldsymbol{n}$ is a circular stationary random vector, $\mathbf{C}_{\boldsymbol{n}}^{-1}$ is also diagonalized by the Fourier transform. We represent the 2-D Fourier transform by the matrix $\mathbf{F}$. $\mathbf{F}^T$ is its unitary inverse such that $\mathbf{F}^T\mathbf{F} = \mathbf{I}$. We have

$$\begin{aligned} \mathbf{H}^T\mathbf{C}_{\boldsymbol{n}}^{-1}\mathbf{H} &= \mathbf{F}^T\mathbf{F}\ \mathbf{H}^T\ \mathbf{F}^T\mathbf{F}\ \mathbf{C}_{\boldsymbol{n}}^{-1}\ \mathbf{F}^T\mathbf{F}\ \mathbf{H}\mathbf{F}^T\mathbf{F} & (4.17) \\ &= \mathbf{F}^T\Lambda_{\mathbf{H}}^{*}\Lambda_{\boldsymbol{n}}^{-1}\Lambda_{\mathbf{H}}\mathbf{F}, & (4.18) \end{aligned}$$

where $\Lambda_{\mathbf{H}} = \mathbf{F}\mathbf{H}\mathbf{F}^T$ and $\Lambda_{\boldsymbol{n}}^{-1} = \mathbf{F}\mathbf{C}_{\boldsymbol{n}}^{-1}\mathbf{F}^T$ are diagonal matrices with the Fourier spectrum of $\mathbf{H}$ and $\mathbf{C}_{\boldsymbol{n}}^{-1}$ along the respective diagonals. $\Lambda_{\mathbf{H}}^{*}$ is the complex conjugate of $\Lambda_{\mathbf{H}}$.

We defined $\boldsymbol{f}$ above to be a circular stationary random vector, therefore $\boldsymbol{C}_{\boldsymbol{f}}^{-1}$ in Equation 4.15 is also diagonalized by the Fourier transform:

$$\begin{aligned} \boldsymbol{C}_{\boldsymbol{f}}^{-1} &= \mathbf{F}^T\mathbf{F}\boldsymbol{C}_{\boldsymbol{f}}^{-1}\mathbf{F}^T\mathbf{F} & (4.19) \\ &= \mathbf{F}^T\Lambda_{\boldsymbol{f}}^{-1}\mathbf{F}. & (4.20) \end{aligned}$$

We can combine Equations 4.18 and 4.20 together to get

$$\mathbf{H}^T\mathbf{C}_n^{-1}\mathbf{H} + C_f^{-1} = \mathbf{F}^T \underbrace{\left(\mathbf{\Lambda}_{\mathbf{H}}^*\mathbf{\Lambda}_n^{-1}\mathbf{\Lambda}_{\mathbf{H}} + \mathbf{\Lambda}_f^{-1}\right)}_{\tilde{\mathbf{\Lambda}}}\mathbf{F}. \tag{4.21}$$

The matrix $\tilde{\mathbf{\Lambda}} = \mathbf{\Lambda}_{\mathbf{H}}^*\mathbf{\Lambda}_n^{-1}\mathbf{\Lambda}_{\mathbf{H}} + \mathbf{\Lambda}_f^{-1}$ is a diagonal matrix. Its diagonal can be calculated and stored before any inversion is done. Altogether calculating the right hand side of Equation 4.21 is an $O(N_f \log(N_f))$ operation.

The final part of Equation 4.15 to examine is the $\mathbf{P}^T\mathbf{C}_{n'}^{-1}\mathbf{P}$ operator. This operator picks data points out of f, operates with $\mathbf{C}_{n'}^{-1}$, and then "unpicks" the resulting vector back to the same size as f. Equation 4.15 can be restated as

$$\left(\mathbf{F}^T\tilde{\mathbf{\Lambda}}\mathbf{F} + \mathbf{P}^T\mathbf{C}_{n'}^{-1}\mathbf{P}\right)\mathbf{f} = \mathbf{b}. \tag{4.22}$$

This equation could be simplified without the existence of $\mathbf{C}_{n'}^{-1}$. When picking data points it is typical that errors on the picked data are independent of each other with a stationary variance $\sigma_{\mathbf{P}}^2$. We assume that this is the case for our inversion, therefore we have that $\mathbf{C}_{n'}^{-1} = \frac{1}{\sigma_{\mathbf{P}}^2}\mathbf{I}$. This further reduces Equation 4.15 to

$$\left(\mathbf{F}^T\tilde{\mathbf{\Lambda}}\mathbf{F} + \frac{1}{\sigma_{\mathbf{P}}^2}\mathbf{P}^T\mathbf{P}\right)\mathbf{f} = \mathbf{b}. \tag{4.23}$$

The $\mathbf{P}^T\mathbf{P}$ operator is an extremely fast $O(N_f)$ operation, thus the forward operation of $\mathbf{F}^T\tilde{\mathbf{\Lambda}}\mathbf{F} + \frac{1}{\sigma_{\mathbf{P}}^2}\mathbf{P}^T\mathbf{P}$ remains $O(N_f \log(N_f))$. The vector $\mathbf{b}$ can be calculated quickly before inverting $\mathbf{A}$ and does not significantly contribute to computation load.

The iterative method we use to solve Equation 4.23 is the conjugate gradients (CG) algorithm. Rather than derive the CG algorithm here we recommend the interested reader to Golub and Van Loan (1996) for details. We list the steps of the algorithm in Figure 4-2. It essentially proceeds by generating several sequences of vectors, $\{\mathbf{f}_0, \mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_N\}$, $\{\mathbf{r}_0, \mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N\}$, and $\{\mathbf{d}_0, \mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_N\}$. The $\mathbf{f}_j$ vectors are a sequence of approximations to the vector $\mathbf{f}_{est} = \mathbf{A}^{-1}\mathbf{b}$ that will minimize the objective functions $\Phi(\mathbf{f}) = \frac{1}{2}\mathbf{f}^T\mathbf{A}\mathbf{f} - \mathbf{f}^T\mathbf{b}$. The $\mathbf{r}_j$ vectors are a sequence of residual vectors that

```
1. Set $\mathbf{f}_0$ = arbitrary, and $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{f}_0$
2. $\beta_j = \mathbf{r}_{j-1}^T \mathbf{r}_{j-1} / \mathbf{r}_{j-2}^T \mathbf{r}_{j-2}$    (except $\beta_1 = 0$)
3. $\mathbf{d}_j = \mathbf{r}_{j-1} + \beta_j \mathbf{d}_{j-1}$    (except $\mathbf{d}_1 = \mathbf{r}_0$)
4. $\alpha_j = \mathbf{r}_{j-1}^T \mathbf{r}_{j-1} / \mathbf{d}_j^T A \mathbf{d}_j$
5. $\mathbf{f}_j = \mathbf{f}_{j-1} + \alpha_j \mathbf{d}_j$
6. $\mathbf{r}_j = \mathbf{r}_{j-1} - \alpha_j A \mathbf{d}_j$
7. Repeat step 2.
```

Figure 4-2: The conjugate gradients algorithm.

measure the misfit between $\mathbf{b}$ and $\mathbf{A}\mathbf{f}_j$. The $\mathbf{d}_j$ vectors are $\mathbf{A}$-orthogonal (Golub and Van Loan, 1996) vectors pointing in the direction of the best $\mathbf{f}$ for the next iteration. The CG algorithm as we have listed it in Figure 4-2, along with our formulation of the $\mathbf{A}$ operator in Equation 4.23, only requires the storage of the current $\mathbf{f}_j$, $\mathbf{r}_j$, and $\mathbf{d}_j$ vectors, each with $N_{\boldsymbol{f}}$ elements, as well as the storage of the diagonal of $\tilde{\mathbf{\Lambda}}$, which also has $N_{\boldsymbol{f}}$ elements. This amounts to $O(N_{\boldsymbol{f}})$ memory requirements.

In the absence of numerical round-off error, the CG algorithm will reach the minimizing solution in $N_{\boldsymbol{f}}$ iterations. In practice it will only iteratively converge to $\mathbf{f}_{est}$ as $j \mapsto \infty$. However, as the algorithm runs, smaller and smaller details are added to the solution. This allows one to stop the algorithm early and still have a solution close to $\mathbf{f}_{est}$. This also has a negative effect that if stopped too early the solution will be effectively smoothed. The effect of this sort of smoothing and that produced by the prior covariance matrix, $\mathbf{C}_{\boldsymbol{f}}$, might be hard to distinguish. Therefore, after obtaining a satisfactory solution one should run a few more iterations to make sure that the smoothness is not an artifact of the CG algorithm. Another benefit of the algorithm is that it can be restarted where it left off by setting $\mathbf{f}_N$ from the last run to be $\mathbf{f}_0$ in next run.

How to establish convergence and automatically stop the CG algorithm is an area of ongoing research. We do not address this issue here but recommend the interested reader to Schneider (2000) for an overview of the subject and some new results. In this chapter we empirically establish a reasonable number of iterations by trial and error.

4.7 Application to data and discussion

Before inverting the data sets, we make one further assumption on Equation 4.15: that $\boldsymbol{n}$ is also identically distributed white noise. Thus $\mathbf{C}_{\boldsymbol{n}} = \sigma_{\boldsymbol{n}}^2\mathbf{I}$. Making this assumption does not take away from the generality of the method. Performing joint inversion in this case reduces to just setting the respective values of $\sigma_{\boldsymbol{n}}^2$, $\sigma_{\boldsymbol{n}'}^2$, $\sigma_{\boldsymbol{f}}^2$, and the correlation length L.

To examine the effectiveness of the joint inversion we create a synthetic data set. We take the real DEM shown in Figure 4-1 and blur it with a PSF, $\mathbf{h}$. The result of this blurring is shown in Figure 4-3. We add independent Gaussian noise with stationary variance of $\sigma_{\boldsymbol{n}}^2 = 2500m^2$ to the blurred field. The blurred and noisy data

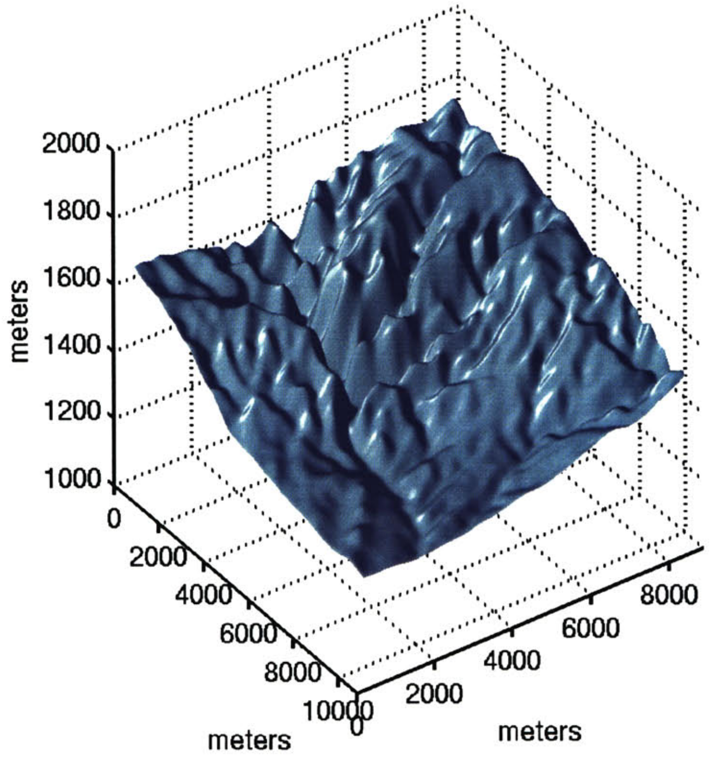

Figure 4-3: Blurred DEM.

set is shown in Figure 4-4. We see that the noise level is large and obscures the

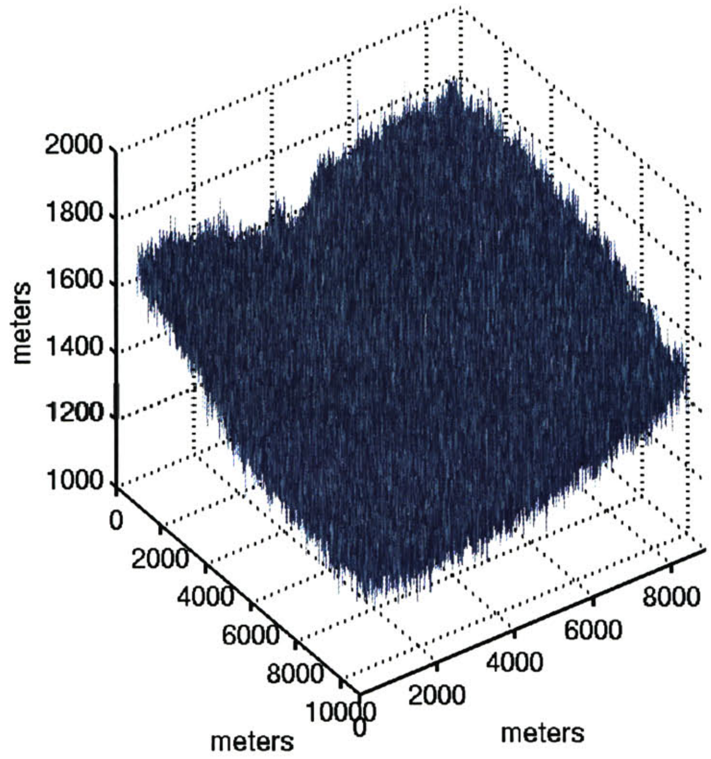

Figure 4-4: Blurred and noisy data.

image. We also extract samples of the DEM and add independent Gaussian noise with stationary variance $\sigma^2_{\boldsymbol{n}'} = 0.0001m^2$. This data is shown in Figure 4-5.

By changing the values of $\sigma^2_{\boldsymbol{n}}$, $\sigma^2_{\boldsymbol{n}'}$, and L one will get very different inversion results. We examine 4 special combination of these parameters: statistical interpolation, damped deconvolution, laterally correlated inversion, and optimal joint inversion. Statistical interpolation sets $\sigma^2_{\boldsymbol{n}} = \infty$, which effectively says the noise on the blurred DEM data set is infinite, and therefore removes that data set from the inversion. Both damped deconvolution and laterally correlated deconvolution set $\sigma^2_{\boldsymbol{n}'} = \infty$, which says that the noise on the sampled data is infinite, and removes that data set from the inversion. The former case sets the $L = 0$, which is equivalent to having $\mathbf{C}_{\boldsymbol{f}} = \sigma^2_{\boldsymbol{f}}\mathbf{I}$. The latter case imposes lateral correlation. For the optimal joint inversion case, we set both $\sigma^2_{\boldsymbol{n}}$ and $\sigma^2_{\boldsymbol{n}'}$ equal to the values we used to generate the respective noises. We intuitively expect this to yield the best result. We list these 4 inversion

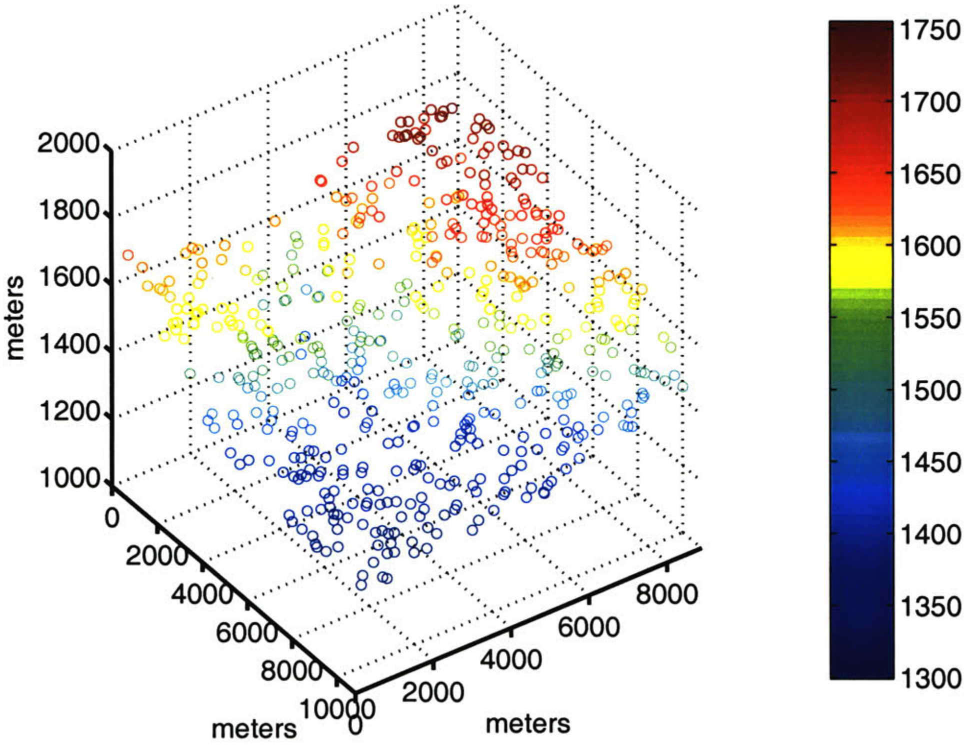

Figure 4-5: GPS data.

cases in Table 4.1.

Inversion Type	σ^2_{n}	$\sigma^2_{n'}$	L	"Optimal" σ^2_{f}
Statistical Interpolation	∞	0.0001	25	2000
Damped Deconvolution	2500	∞	0	4500
Laterally Correlated Decon.	2500	∞	25	2000
Optimal Joint Inversion	2500	0.0001	25	2000

Table 4.1: Parameter combinations for 4 different inversion runs.

We also need σ^2_{f} to invert the data. This, however, is a difficult parameter to set (Vogel, 2002). Setting it too high produces a noise corrupted result. Setting it too low smooths the solution too much. To show the effect on the solution by varying σ^2_{f}, the inversion is run for each of the 4 cases in Table 4.1 with σ^2_{f} ranging from 500 to 4500, in steps of 500. The ℓ_2 norm of the error between $\mathbf{f}_{est}$ and the true DEM shown in Figure 4-1 is calculated for each run. The results of these runs are shown in Figure 4-6. We also zoom in and show the same results for just the laterally correlated deconvolution and optimal joint inversion in Figure 4-7. This plot clearly shows a

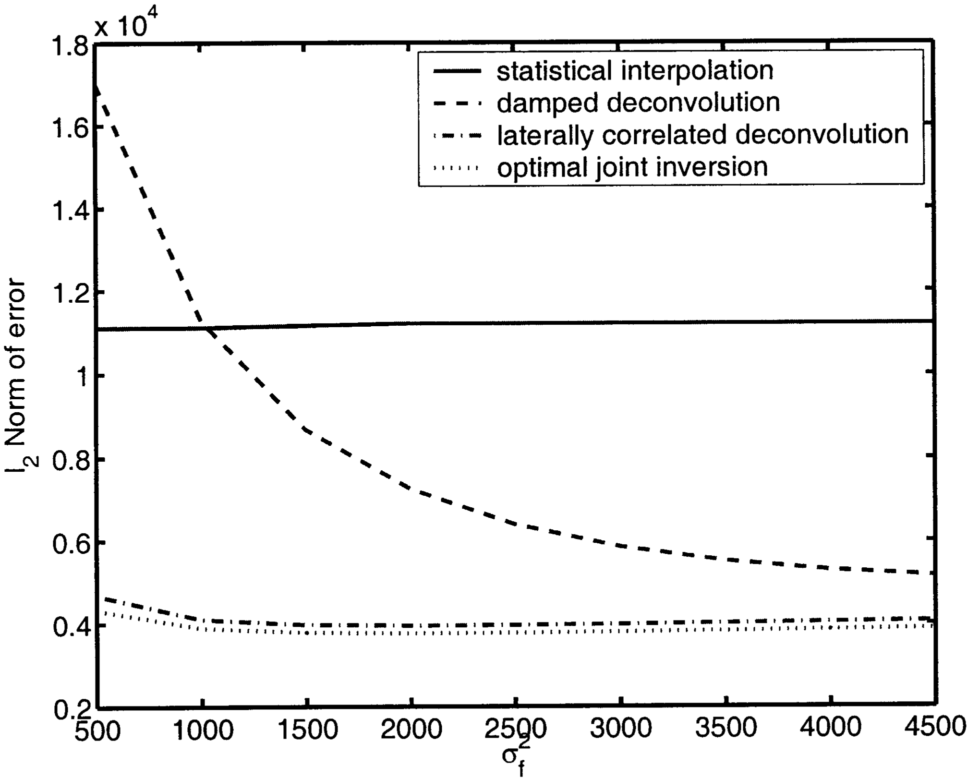

Figure 4-6: Comparison of ℓ_2 norm of error for different inversion cases across different values of σ_f^2.

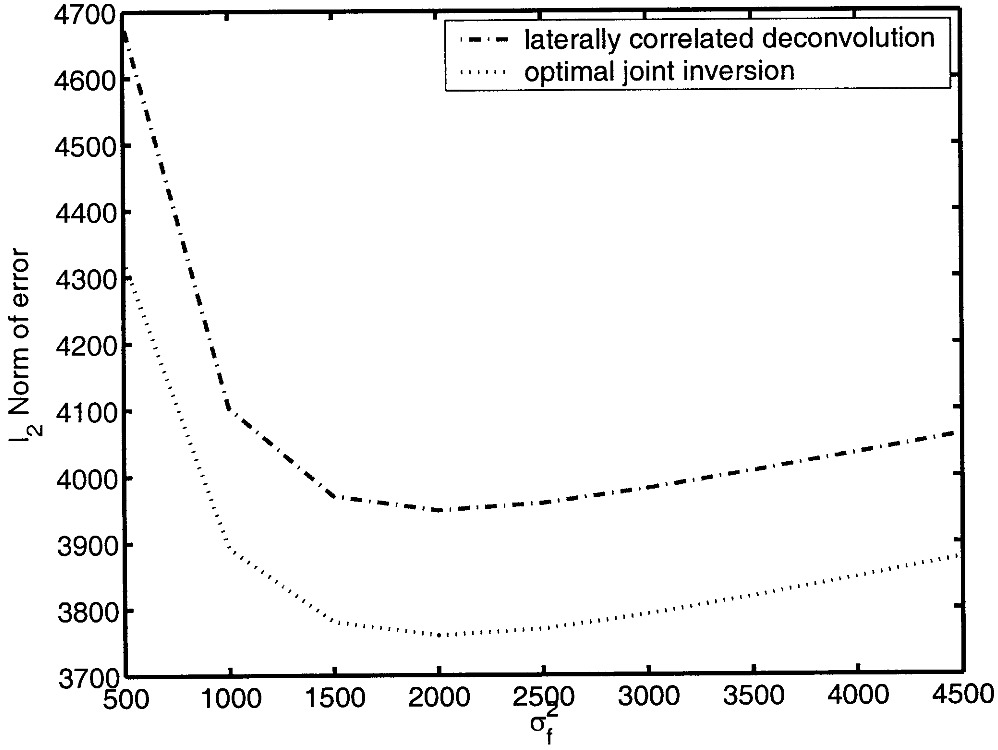

Figure 4-7: Comparison of ℓ_2 norm of error for 2 inversion cases across different values of σ_f^2.

minimum at the $\sigma_{\boldsymbol{f}}^2 = 2000$ on both curves. It should be emphasized that this sort of comparison cannot be done in practice, since the true solution is not known. It is only shown here to demonstrate the effect $\sigma_{\boldsymbol{f}}^2$ has in the various inversion cases, and which case tends to give the best (smallest error) solutions. It should also be noted that the ℓ_2 norm may not be the best criterion for comparing the inversion cases. It is a global measure and does not take into account local improvement of the inversion around sample locations.

We see that statistical interpolation by itself tends to perform the worst, since it only provides accurate estimates near sample locations. Damped deconvolution performs better but makes no use of the a-priori geostatistical structure of $\boldsymbol{f}$. Including this prior information improves the inversion in the laterally constrained case, but we still have not included the sampled data. Finally, including all data sets and the correlated prior we arrive at the optimal joint inversion solution, which gives the smallest error of all the cases across all values of $\sigma_{\boldsymbol{f}}^2$.

The final column of Table 4.1 lists the "optimal" prior variance. This is chosen by selecting the lowest error result from Figure 4-6 for each inversion case. Each of these "optimal" results is plotted in Figures 4-8 through 4-15. In these plots we do not plot inversion results within 15 units of the edges because of edge effects caused by circular convolutions.

In Figure 4-8 we see that statistical interpolation honors the data at the sample locations but is smooth between samples, as it cannot reconstruct greater detail there. In Figure 4-9 we plot of the difference between the true DEM and the estimate from statistical interpolation. The sample locations are overlaid as black circles. We notice that the error near sample locations is minimal but increases significantly away from the samples.

In Figure 4-10 the damped deconvolution gives suboptimal results uniformly across the whole model. This can be seen in both the ringy quality of the estimate and the apparently random distribution of error in Figure 4-11. Since the GPS data is not honored there is no error reduction around those locations.

In Figure 4-12 the laterally correlated inverse solution is slightly better than

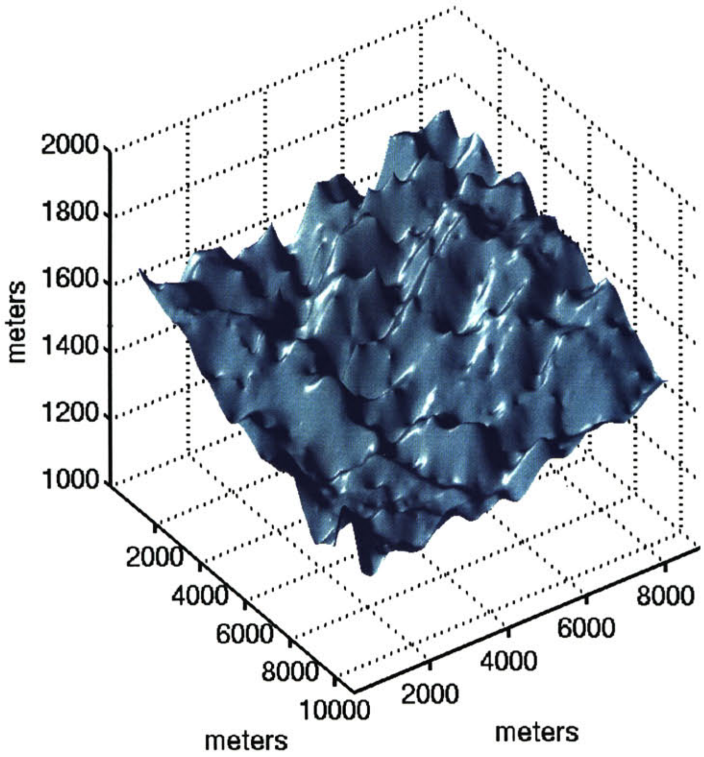

Figure 4-8: Statistical interpolation.

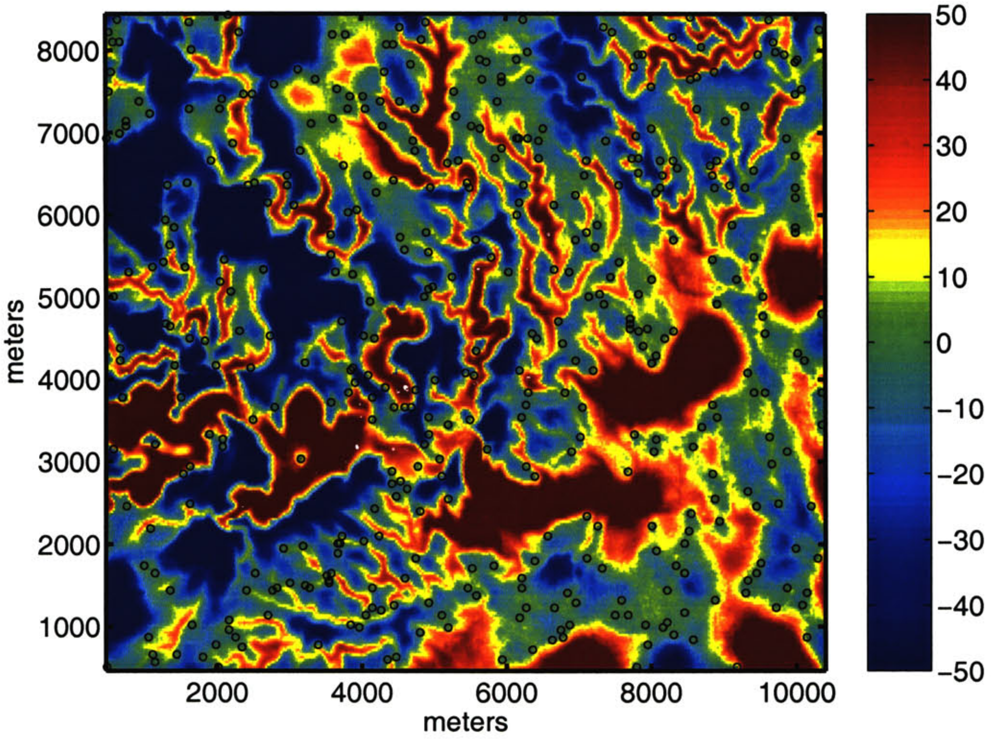

Figure 4-9: Inversion result minus true DEM (in meters).

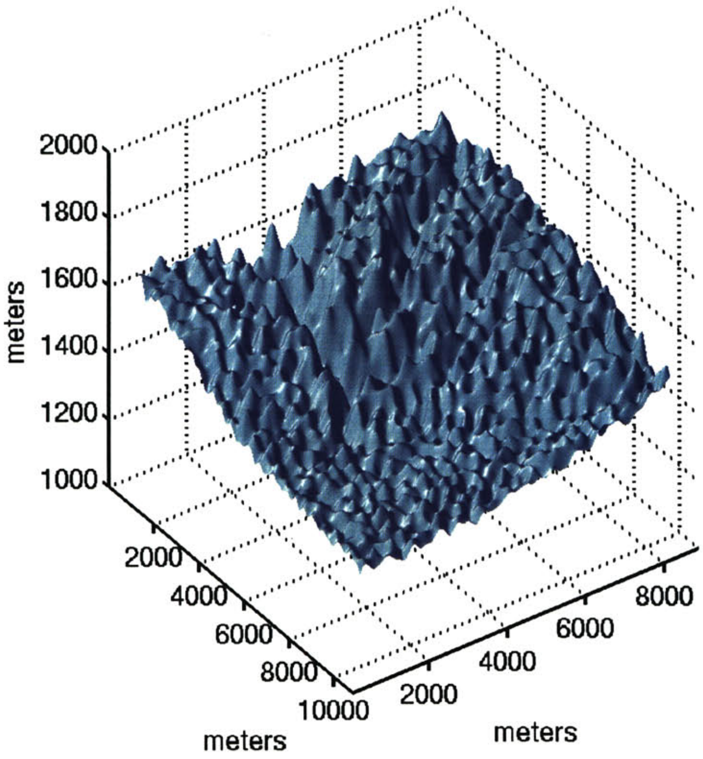

Figure 4-10: Damped deconvolution.

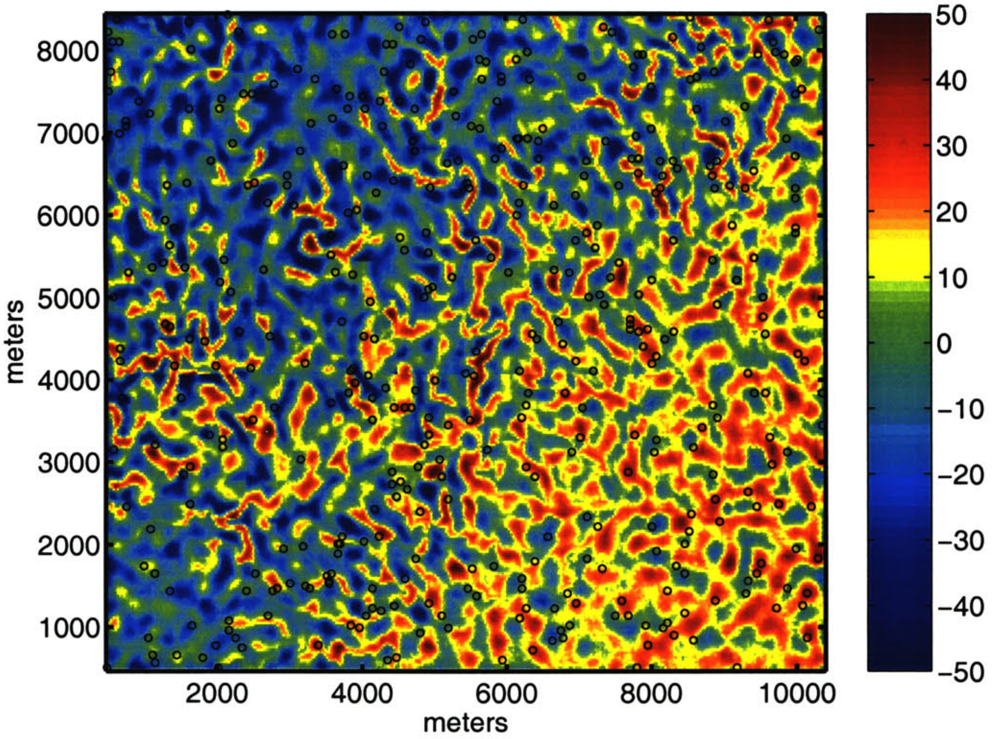

Figure 4-11: Inversion result minus true DEM (in meters).

damped deconvolution in that it produces an estimate with the proper correlation length. The overall error of the estimate is less but local GPS data is still not honored.

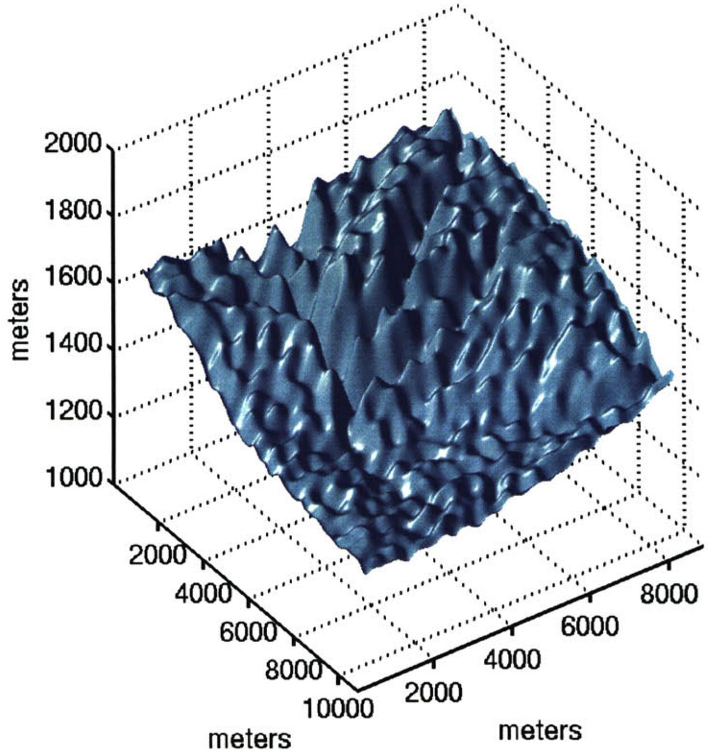

Figure 4-12: Laterally correlated deconvolution.

Finally, the optimal joint inversion estimate in Figure 4-14 gives a good solution uniformly across the model while providing an almost exact fit at the data locations. Although this is difficult to see in Figure 4-15, the estimate matches the sample data at the sample locations.

4.8 Conclusions

We have presented a method for jointly inverting two remote sensing data sets for an optimal model of topographic elevation. The maximum likelihood inversion framework generated a large system of equations to solve. Using the conjugate gradients method overcame the problem of large dimensionality.

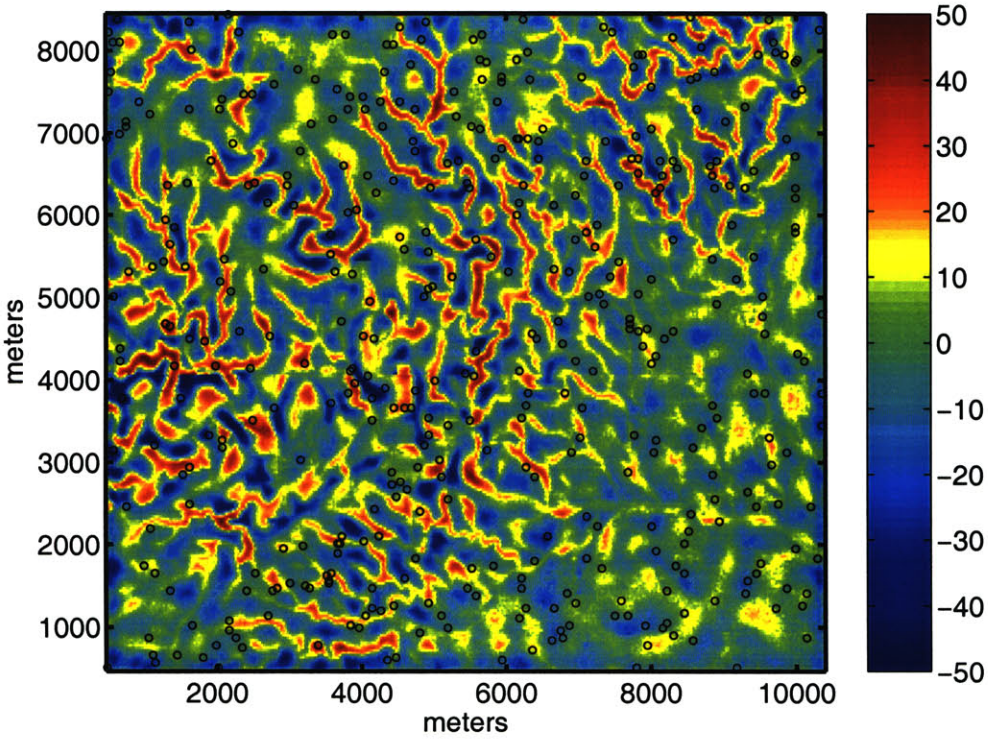

Figure 4-13: Inversion result minus true DEM (in meters).

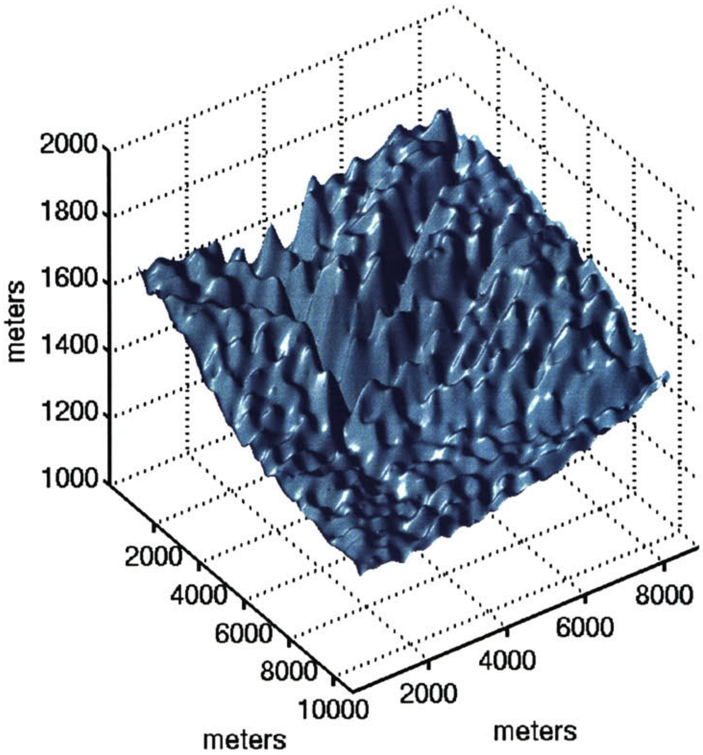

Figure 4-14: Optimal joint inversion.

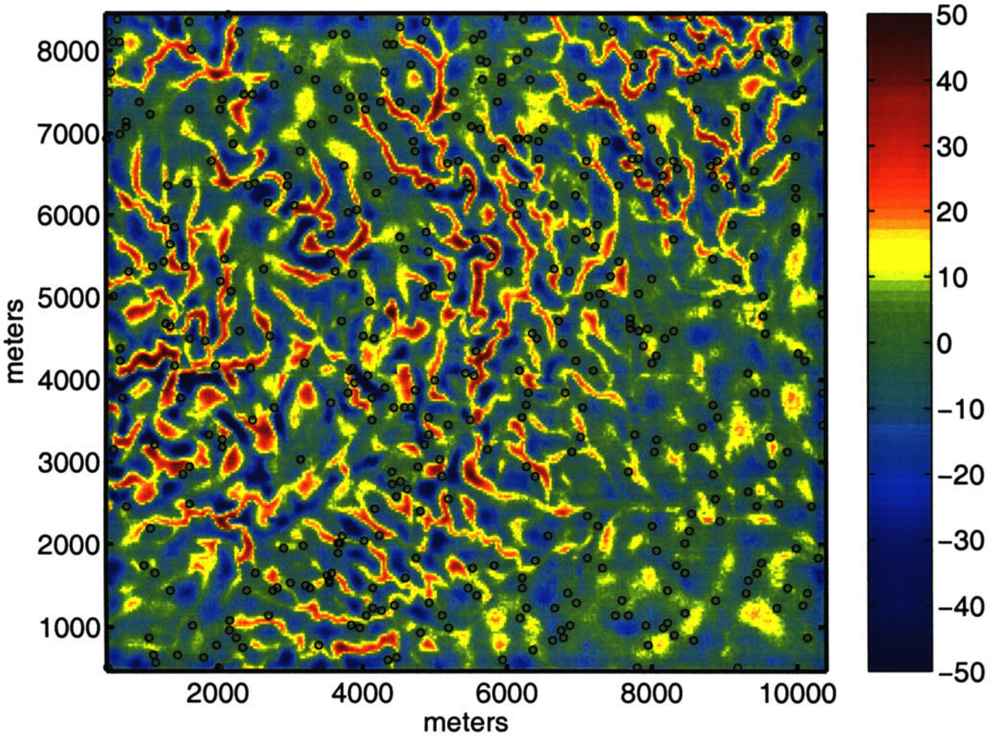

Figure 4-15: Inversion result minus true DEM (in meters).

The method was applied to synthetic data mimicking DEM and GPS data for 4 separate inversion cases. Statistical interpolation did not make use of the DEM data. Damped deconvolution consistently outperformed the statistical interpolation results but did not make use of geostatistical prior knowledge. Laterally correlated deconvolution used an exponential correlation function as a prior model and gave superior results to the damped deconvolution, but still did not use the GPS data. Finally, by using all data sets with correct assumptions of noise levels, the optimal joint inversion consistently gave the best results. In addition to having the smallest error compared to the true solution, it almost perfectly matched the GPS data at the sample locations. Thus we see that by posing deconvolution and interpolation as a single joint inversion we can still arrive at either interpolation or damped deconvolution as special cases, but incorporating both of the data sets gives the superior solution.

The methodology presented in this chapter is not limited to only deconvolution and interpolation problems. Any other data set can be incorporated into the Bayesian framework if it has a well defined likelihood function. This includes data arising from nonlinear forward operators. The conjugate gradients algorithm can still be applied in nonlinear inverse problems. Thus the method presented in this chapter might be ideal for general data fusion of multiple data sets. This remains an area for future research.

Recent work has attempted to use the CG algorithm to give the estimation error variance in addition to the estimate at no additional cost. There has been some success with this (Schneider, 2000) and it is a promising direction, but, due to instability in the CG algorithm, there remain many problems to be resolved. Also, the CG algorithm is not the only iterative method that could be applied to the joint problem. Other algorithms such QRMS, bi-conjugate gradients, Gauss-Seidel, etc. may solve the problems associated with the CG algorithm. This is an open area to investigate and it is not known what benefits these other methods might bring.

A final problem that needs to be addressed is how to set the prior variance. This parameter greatly affects the inversion result and is notoriously hard to optimize. Methods such as *generalized cross validation* or *L-curve method* (Vogel, 2002) may prove useful.

Chapter 5

Fractional splines and discrete data

Geophysicists deal with multidimensional petrophysical parameter fields. These fields are often idealized as continuous functions of three spatial dimensions and perhaps a fourth temporal dimension. Representing such a field of parameters on a digital computer is impossible, so the data sets dealt with must be discretized into a finite number of coefficients in order to be manipulated or analyzed. **Splines** provide a link between continuous and discrete representations of a function.

Splines are piecewise polynomial **approximations** of a true function. They are representable by linear combinations of **spline basis functions**, which possess useful mathematical properties and are the building blocks of any arbitrary spline. By manipulating the coefficients of such basis functions one can effectively perform operations on the spline.

More specifically, classical integer degree splines are defined to be piecewise continuous polynomials (Unser, 1999) indexed by an integer parameter α, indicating the number of bounded derivatives of the function. They can be conceptualized as being constructed from a basis of functions that also have α bounded derivatives. The α^{th} derivative of an α degree spline yields a function that is piecewise constant with discontinuities at the locations where the piecewise polynomials meet. These points are called **knots** and are usually constrained to lie at the nodes of a regularly spaced grid. Recently the integer degree spline framework has been generalized by Blu and Unser (2001a). The parameter α can take fractional, non-integer values resulting in

fractional splines. In this case the parameter α represents the **Hölder** degree of continuity, a generalization of the traditional concept of continuity to define functions with non-integer degrees of differentiability. The Hölder exponent, α, is also sometimes called the *Lipschitz* exponent (Mallat, 1998, page 166).

In the first section we present the concept of projecting a continuous function into a function space spanned by a set of basis functions. Linear combinations of these basis functions generate an approximation to the original function. The section introduces Riesz bases, which allow for the concept of a dual function. Associating the dual function with a physical tool used to obtain geophysical measurements, we construct a framework for optimally approximating the original function. Prefiltering is presented as a way to project a function into an arbitrary basis of our choice. The concept of accuracy of approximation of a basis is also introduced. Finally, a periodic basis is defined so that filtering operations can be defined on the approximation.

In the second section we present spline basis functions as an ideal basis for representing a function. There are several kinds, each possessing useful properties. Algebraic singularity functions are the arch-types of functions that possess a certain degree of Hölder continuity. From these we construct other types of spline basis functions that retain the Hölder continuity but possess much better mathematical properties. Such basis functions include B-splines, cardinal splines, and orthogonal splines.

Finally, in the third section, we show how the basis into which we have projected affects the form of linear operators in that basis. We present the particular example of a derivative operator acting on a discrete geophysical data set in order to illustrate the potential error introduced by careless manipulation of coefficients.

5.1 Projection onto a basis

Given a Hilbert space, $\mathcal{H}$, a **projector**, P, is a linear operator mapping $\mathcal{H}$ into a subspace $\mathcal{U}$ such that (Mallat, 1998, Appendix A.4):

1. $P_{\mathcal{U}}$ is orthogonal,

2. $P_{\mathcal{U}}$ is self-adjoint,

3. $||P_{\mathcal{U}}|| = 1$,

4. $\forall f \in \mathcal{H}, ||f - P_{\mathcal{U}} f|| = \min_{f' \in \mathcal{U}} ||f - f'||$.

For our purposes the space $\mathcal{H}$ will be $\mathcal{L}_2$, the space of square integrable functions. $\mathcal{U}$ will a subspace spanned by a set of basis functions $g_k = g(x - k)$ of our choice. The **projection**, $f_{\mathcal{U}}$, of f into $\mathcal{U}$ will then have the form

$$\begin{aligned} P_{\mathcal{U}} f &= f_{\mathcal{U}} & (5.1) \\ &= \sum_k c[k] g(x - k). & (5.2) \end{aligned}$$

Property 4 essentially says that $f_{\mathcal{U}}$ is the function in $\mathcal{U}$ that is closest to f in an $\mathcal{L}_2$ sense.

To make the concept of a projection more concrete we show in Figure 5-1 a well log measuring the rock velocity as a function of depth. In Figure 5-2 we show a close-up view of the area enclosed in the box of Figure 5-1. We see that the log is

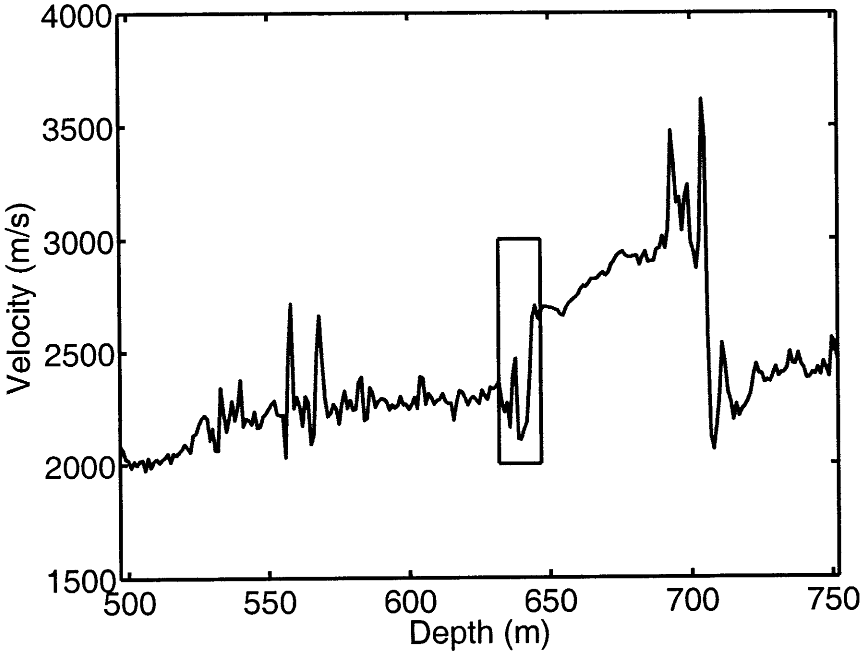

Figure 5-1: Velocity as a function of depth.

not a continuous function, but rather a discrete sequence of coefficients. What *are* these coefficients? One may wish them to be point samples of the true function (as many geophysicists are in the habit of doing) but that is usually not true. These are the coefficients, $c[k]$, in Equation 5.2. The well logging tool that took these samples probably took a weighted average of values around the sample point. It is reasonable to assume that the tool took the same average around each sample point and also that the samples were evenly spaced[1]. We can then view the well logging process as a convolution of f with $\bar{g}(-x)$, which models the well logging tool, followed by a discretization:

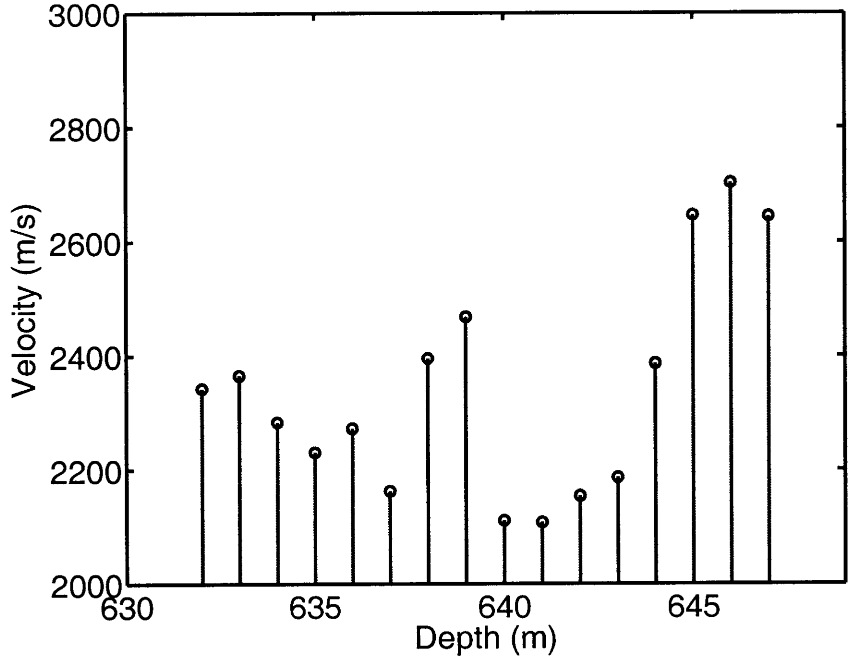

Figure 5-2: Samples of velocity log in black box of Figure 5-1.

$$c[k] = \delta_c(x)\left[\bar{g}(-x) * f(x)\right]. \tag{5.3}$$

$\delta_c(x)$ is the *Dirac comb* function (Mallat, 1998, page 29) that performs the action of sampling the continuous function it is multiplied with. Unser (1999) would call $\bar{g}$ a **prefilter**.

[1] The spacing of the samples can be any distance Δx. For simplicity we set $\Delta x = 1$ in all equations unless otherwise noted.

Given coefficients and a basis, we can reconstruct the projection, $f_{\mathcal{U}}$, according to Equation 5.2. We reconstruct an approximation of f using cubic B-splines (Section 5.2) in Figure 5-3. We notice that the reconstructed approximation does not equal the coefficients at the sample locations. In general, this will be true. The coefficients will equal samples of the approximation function only when our basis functions are *interpolating* functions, such as cardinal splines (Section 5.2).

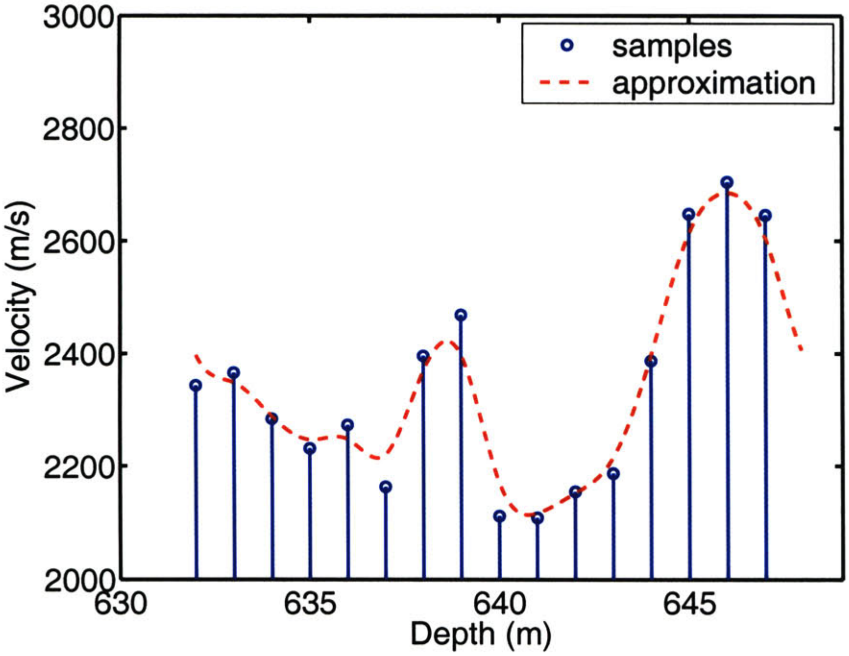

Figure 5-3: Samples of velocity log and reconstructed approximation, $f_{\mathcal{U}}$.

5.1.1 Riesz bases

Mallat (1998, Appendix A.3) gives a concise definition of Riesz bases. In the Hilbert space $\mathcal{U} \subset \mathcal{L}_2$ the existence of such a basis says that any function, $f_{\mathcal{U}}$, can be represented in terms of a sequence of linearly independent functions $g_k = g(x - k) \in \mathcal{U}$, with constants $A > 0$ and $B > 0$, such that

$$\begin{aligned} f_{\mathcal{U}}(x) &= \sum_k c[k] g(x-k), \quad (5.4) \\ &\text{and} \\ \frac{1}{B}\|f_{\mathcal{U}}\|^2 &\leq \sum_k |c[k]|^2 \leq \frac{1}{A}\|f_{\mathcal{U}}\|^2. \quad (5.5) \end{aligned}$$

Equation 5.5 expresses a partial $\ell_2 - \mathcal{L}_2$ norm equivalence between $f_{\mathcal{U}}$ and $c[k]$. If the basis functions, g were orthogonal, A would equal B and the norm of $c[k]$ would equal the norm of $f_{\mathcal{U}}$. The Riesz Representation Theorem (Naylor and Sell, 1982) proves that there exist **dual** functions $\bar{g}$ such that

$$c[k] =< \bar{g}(x-k), f >, \ \forall k \in \mathbb{Z}. \quad (5.6)$$

Both g and $\bar{g}$ span the same space of functions. Therefore $f_{\mathcal{U}}$ could equivalently be represented by the dual functions with a similar norm equivalence property:

$$\begin{aligned} f_{\mathcal{U}}(x) &= \sum_k \bar{c}[k] \bar{g}(x-k), \quad (5.7) \\ &\text{and} \\ A\|f_{\mathcal{U}}\|^2 &\leq \sum_k |\bar{c}[k]|^2 \leq B\|f_{\mathcal{U}}\|^2. \quad (5.8) \end{aligned}$$

In this case the coefficients are

$$\bar{c}[k] =< g(x-k), f >, \ \forall k \in \mathbb{Z}. \quad (5.9)$$

The logic of Riesz bases says that, given a Hilbert space, one can represent any element of it by linear combinations of a convenient basis function, g. How to weight those basis functions is given by Equation 5.6, which says that one should take the inner product of the dual functions with $f_{\mathcal{U}}$ to give us the c's. Relating these concepts to the well log data of Figures 5-1 and 5-2 we see that we wish to apply the logic in reverse. That is, we start with known dual basis functions and coefficients, and wish to know the correct basis function with which to construct our projection.

To see that the dual of the dual is, in fact, the obvious basis function to use, we note that the approximation error of an orthogonal projection, $f_{\mathcal{U}}$, must be orthogonal to $\mathcal{U}$, which implies that

$$\langle f - f_{\mathcal{U}}, \bar{g}(x-l)\rangle = 0. \tag{5.10}$$

We immediately see that

$$\langle f, \bar{g}(x-l)\rangle = \langle f_{\mathcal{U}}, \bar{g}(x-l)\rangle\,. \tag{5.11}$$

This tells us that the coefficients arising from operating $\bar{g}_k$ on f are equal to the coefficients of operating $\bar{g}_k$ on its projection, $f_{\mathcal{U}}$. We could potentially choose any basis of functions, ϕ, spanning some function space, $\mathcal{V} \subset \mathcal{L}_2$, to associate with the coefficients $c[k]$. We, however, would not have any way of knowing how good of an approximation $\sum_k c[k]\phi(x-k)$ would be. If we use g, we know that our approximation is the optimal one possible (in an $\mathcal{L}_2$ sense) in the subspace $\mathcal{U}$.

From the preceding we see that our knowledge of the measurement process plus the coefficients resulting from an experiment should determine the appropriate basis function to represent the data[2]. The reconstruction formula in Equation 5.4 would then tell us how to evaluate $f_{\mathcal{U}}$ at arbitrary locations. To do this one simply evaluates each g_k times its coefficient, $c[k]$, at the location and sums the values.

5.1.2 Prefiltering

We may not wish to represent f in a space spanned by g, as was done in Equation 5.4. Indeed, g may not be a very well behaved function and there are many reasons to select a different basis ϕ to approximate f. For instance, certain linear operators may become very sparse when represented as matrices in a certain basis (e.g., see

[2] As a side note, we mention that these arguments only apply when the data gathering process can be modeled as a linear functional for each data point. If the functional is nonlinear it is not clear how to choose a basis.

Section 5.3). Therefore one basis may lead to greater computational efficiency than another. To project f into a space spanned by ϕ requires **prefiltering**, in which we pass our coefficients through a discrete filter to obtain a new set of coefficients.

To illustrate how this works we present an equation similar to 5.4:

$$f_{\mathcal{V}}(x) = \sum_k a[k]\phi(x-k). \tag{5.12}$$

The difference here is that ϕ_k are the basis functions approximating f, and they span a different space than the g_k's. $f_{\mathcal{V}}(x)$ is a different function than $f_{\mathcal{U}}(x)$. We have $c[k]$, what we want are the coefficients $a[k]$. To calculate these we desire that $f_{\mathcal{U}}(x)$ and $f_{\mathcal{V}}(x)$ are as close to each other as possible. Therefore we evaluate Equations 5.4 and 5.12 at the knots to give us

$$f_{\mathcal{U}}[l] = \sum_k c[k]g[l-k] \tag{5.13}$$

$$f_{\mathcal{V}}[l] = \sum_k a[k]\phi[l-k]. \tag{5.14}$$

We can represent the convolutions in Equations 5.13 and 5.14 in matrix/vector notation as

$$\mathbf{f}_{\mathcal{U}} = \mathbf{Gc} \tag{5.15}$$

$$\mathbf{f}_{\mathcal{V}} = \mathbf{\Phi a}, \tag{5.16}$$

where $\mathbf{G}$ and $\mathbf{\Phi}$ are convolution matrices.

We next minimize the difference between $\mathbf{f}_{\mathcal{U}}$ and $\mathbf{f}_{\mathcal{V}}$ over all possible $\mathbf{a}$ in the ℓ_2 norm:

$$\min_{\mathbf{a}} ||\mathbf{f}_{\mathcal{U}} - \mathbf{f}_{\mathcal{V}}||_{\ell_2}. \tag{5.17}$$

To find this minimum we expand the norm, take its derivative with respect to $\mathbf{a}$, and set this equal to $\mathbf{0}$:

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{a}}\left[\mathbf{c}^T\mathbf{G}^T\mathbf{G}\mathbf{c}+\mathbf{a}^T\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{a}-2\mathbf{a}^T\mathbf{\Phi}^T\mathbf{G}\mathbf{c}\right] &= \mathbf{0} && (5.18)\\
2\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{a}-2\mathbf{\Phi}^T\mathbf{G}\mathbf{c} &= \mathbf{0} && (5.19)\\
\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{a} &= \mathbf{\Phi}^T\mathbf{G}\mathbf{c} && (5.20)\\
\mathbf{a} &= (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{G}\mathbf{c}. && (5.21)
\end{aligned}$$

If $\mathbf{\Phi}$ is invertible, we can reduce Equation 5.21 to

$$\mathbf{a} = \underbrace{\mathbf{\Phi}^{-1}\mathbf{G}}_{\mathbf{V}}\mathbf{c}, \tag{5.22}$$

which shows the precise form of the proper prefilter:

$$\mathbf{V} = \mathbf{\Phi}^{-1}\mathbf{G}. \tag{5.23}$$

Rewriting Equation 5.22 as discrete convolutions of filters, we have

$$a[k] = \phi[k]^{-1} * g[k] * c[k]. \tag{5.24}$$

The convolution of $g[k]$ with $c[k]$ converts the c's into actual point samples of $f_{\mathcal{U}}$. Subsequent convolution with $\phi^{-1}[k]$ converts these point samples into coefficients in the ϕ_k basis.

Although prefiltering is advocated by many researchers including Strang (1986), Mallat (1998), and Choi and Baraniuk (1999), we note that it is rarely done in practice for a couple of reasons:

1. The filters $g[k]$ and $\phi[k]$ may be infinite in length (IIR) and slow computations. Therefore one may wish to stay in the suboptimal basis g_k.

2. In dealing with data such as the well log data in Figure 5-1 we often *don't know* the sampling function $\bar{g}$. Without this knowledge one might as well treat the samples as coefficients in a convenient basis of our choice, such as splines

(Section 5.2).

5.1.3 Accuracy of approximation

Now that we have seen how to project a function f into an arbitrary basis the question becomes "which basis is best?". We want our approximate function to be as close to the original function as possible. Therefore we examine how to quantitatively measure the error of approximation. **Approximation theory** (Strang and Nguyen, 1997; Mallat, 1998; Unser, 1999; Thévanaz et al., 2000) provides a framework for doing this.

We let $\phi(\frac{x}{\Delta x} - k)$ be a Riesz basis in $\mathcal{V}_j \subset \mathcal{L}_2$ with a variable discretization level $\Delta x = 2^{-j}$. $f_{\mathcal{V}_j}$ is the projection of f onto this basis of functions. Unser (1999, page 28) gives the following error bound for approximating a function in this basis:

$$\|f - f_{\mathcal{V}_j}\| \leq C(\Delta x)^L \|f^{(L)}\|, \qquad \forall f \in \mathcal{H}_2^L. \tag{5.25}$$

L is the *order of the approximation.* It and the constant, C, are functions of $\mathcal{V}_j$ and the **Sobolev space** to which f belongs. A Sobolev space, $\mathcal{H}_2^L$, contains functions whose L^{th} derivative lies in $\mathcal{L}_2$. $\mathcal{H}_2^0 = \mathcal{L}_2$ is the previously defined Hilbert function space of square integrable functions.

The way to interpret Equation 5.25 is as follows. We assume prior knowledge of the largest possible L' such that $f \in \mathcal{H}_2^{L'}$. We then pick a subspace $\mathcal{V}_j$, the elements of which are L times differentiable. Equation 5.25 then holds for all $L \leq L'$. The best basis to use is the one with the largest L because the term $(\Delta x)^L$ in Equation 5.25 becomes the smallest. This classical *Strang-Fix* theory has been generalized by Unser and Blu (2000) to include non-integer orders of approximation, enabling one to exactly match basis functions to the Hölder continuity of the function being approximated.

What, then, does one do in a real application if one is unsure of the Hölder continuity of the true function, f? Or, even worse, what if the Hölder coefficient of that function changes from point to point?[3] We put forth a couple of suggestions:

1. If one is unsure of the Hölder degree of continuity of the true function, an estimate could be obtained from prior data. A method is presented by Li et al. (1996) that does this by examining the variances of wavelet transform coefficients as a function of scale.

2. If the Hölder degree changes from point to point the best one can do, without a-priori knowledge of point-wise continuity, is to use one global value. Complex functions such as these are known as **multi-fractals**. One could again use the wavelet transform coefficient variances method of Li et al. (1996). Another method is suggested in the **singularity spectrum** used by Herrmann (1997). The singularity spectrum shows how much of the support of a continuous function appears to scale with a certain Hölder exponent. One could simply pick either the most commonly occurring, or the average, Hölder exponent for specifying the smoothness of a basis.

5.1.4 Periodic bases

We have seen how to take an infinite dimensional continuous function, approximate it in a subspace, and represent the approximation with a discrete sequence of coefficients. In most applications the data comes in the form of finite length discrete vectors. The basis that spans the Hilbert space then has a finite number of basis functions. Finite dimensional vectors present a problem when one performs filtering, which requires knowledge of vector values beyond the end points. There are essentially two ways to handle this problems. The first constructs a filter such that it handles edge values with special "boundary filters". The second, and more common, way is to artificially extend the vector. The effect of the filter will still cause problems by "smearing" the edges. Following Strang and Nguyen (1997), we examine 3 methods mitigating this effect:

[3] Herrmann (1997) has shown this to be the case in well data.

1. Padding the vector with zeros (or some other constant).

2. Assuming the observed signal periodically repeats itself off to infinity.

3. Assuming that a mirrored version of the function repeats itself off to infinity.

The first two methods are often used in conjunction; that is, the vector is padded with zeros and then conceptualized as repeating periodically to allow for Fourier domain filtering. This method says that the values of the function at locations outside the interval repeat those within the interval infinitely. Applying this concept to the function f gives

$$f(x - nT) = f(x),\ n \in \mathbb{Z}, \tag{5.26}$$

where the support of f is $[0, T]$. Letting this periodization extend also to the approximation of f in Equation 5.4, we arrive at

$$\begin{aligned} f_U(x - nT) &= f_U(x) & (5.27)\\ \sum_k c[k]g(x - k - nT) &= \sum_k c[k]g(x - k), & (5.28) \end{aligned}$$

which says that

$$g(x - k - nT) = g(x - k). \tag{5.29}$$

From this we see that periodizing the projection will also periodize its basis functions. Although we technically have an infinite number of coefficients now, they repeat themselves infinitely and only one finite set needs to be retained and manipulated. In this way we have modeled a continuous function of infinite support by a finite vector of coefficients.

Method 3 is similar to method 2. f is first mirrored across its end points so that

$$f(x + T) = f(T - x), \forall x \in [0, T]. \tag{5.30}$$

This defines a new function f' over the interval $[0, 2T]$. This new function is then repeated periodically over the entire real line. The basis functions for f' are then also a periodic basis. Whereas the periodic approximation function from method 2 usually has discontinuities at its edges, that from method 3 is continuous at the edges (but not continuous in its derivative). If a convolutional filter is applied to a periodic function, method 2 will affect the result near the edges of the image more than method 3. In applying a 2-D wavelet transform to an image, Strang and Nguyen (1997, p. 340) shows that mirrored extension results in the least distortion at the edges.

In this thesis we will mostly stick to the combination of methods 1 and 2 for simplicity, although we will apply method 3 in chapter 7. We do not use method 1 by itself.

5.2 Spline basis functions

Up to this point we have only spoken of generic basis functions that we might use to represent a function f. In this section we present certain candidate basis functions that form what are known as **spline spaces**, $\mathcal{S} \subset \mathcal{L}_2$. These spaces contain functions that are piece-wise polynomials with an adjustable degree of differentiability. Among many useful properties, spline basis functions possess optimal approximation properties according to Equation 5.25 (Strang and Nguyen, 1997; Thévanaz et al., 2000). There are many possible bases for a spline space. We examine four of them, each with useful and different mathematical properties.

5.2.1 Algebraic singularity functions

One-sided algebraic singularity functions[4], denoted $\chi_+^\alpha(x)$, have α bounded derivatives and a singularity at the origin with Hölder exponent α:

$$\chi_+^\alpha(x) = \begin{cases} \frac{x^\alpha}{\Gamma(\alpha+1)}, & x \geq 0 \\ 0, & \text{otherwise.} \end{cases} \tag{5.31}$$

The Fourier transform of such a function (in a distributional sense) is

$$\hat{\chi}_+^\alpha(\omega) = \frac{1}{(i\omega)^{\alpha+1}}. \tag{5.32}$$

This is identical to Zemanian (1987, page 348) with the exception of the placement of the Γ term. We present the function in this form to make it the inverse of the causal derivative operator. It should be noted that these functions are not normal functions but rather **generalized functions** (Gel'fand and Shilov, 1964). They are unbounded and of infinite support and require care when manipulated mathematically.

The anti-causal one-sided algebraic singularity function is simply defined in terms of the causal one as:

$$\chi_-^\alpha(x) = \chi_+^\alpha(-x) \tag{5.33}$$

We will require symmetric versions of these functions later so we define them here as (Unser and Blu, 1999):

$$\chi_\star^\alpha(x) = \begin{cases} \frac{|x|^\alpha}{-2sin(\pi/2\alpha)\,\Gamma(\alpha+1)}, & \alpha \text{ not even} \\ \frac{x^{2n}\log x}{(-1)^{1+n}\pi\,\Gamma(\alpha+1)}, & \alpha = 2n \text{ (even)} \end{cases} \tag{5.34}$$

[4] Also known as one-sided **power functions** or **homogeneous distributions** (Gel'fand and Shilov, 1964).

These functions possess Fourier transforms

$$\hat{\chi}_{\star}^{\alpha}(\omega) = \frac{1}{|\omega|^{\alpha+1}}. \tag{5.35}$$

The subscripts "+", "−", and "⋆" imply causality, anti-causality, and symmetry of the algebraic singularity functions, respectively.

In Figure 5-4 we show an $\alpha = 1$ causal algebraic singularity function. Linear combinations of translations of such a function can construct a piecewise linear function as shown in Figure 5-5. Similarly, in Figure 5-6 we plot an $\alpha = 2$ singularity function. A linear combination of its translations is shown in Figure 5-7.

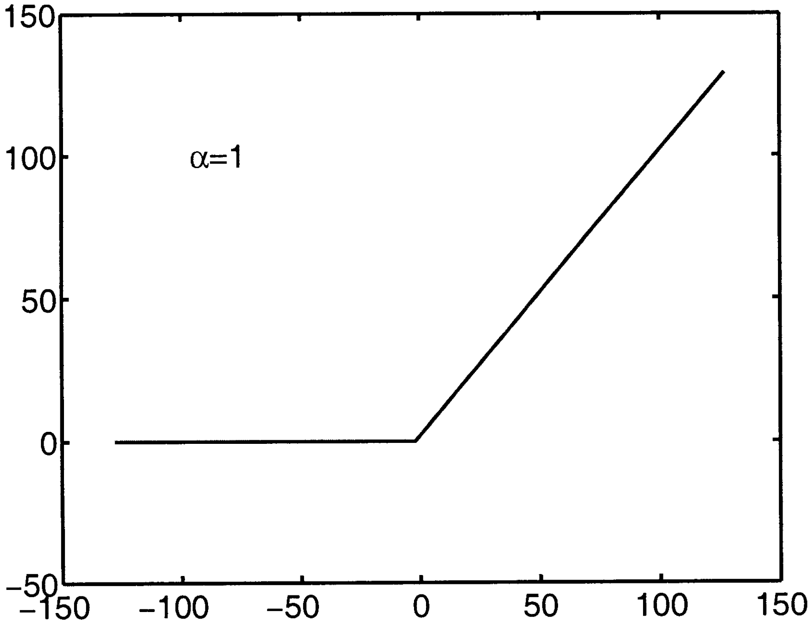

Figure 5-4: Causal algebraic singularity function for $\alpha = 1$.

The big drawback to using algebraic singularity functions as a basis is that they unbounded and of infinite support. It is not possible to periodize them. Linear operators projected into such a basis lead to full matrices. They are useful, though, for creating other spline basis functions.

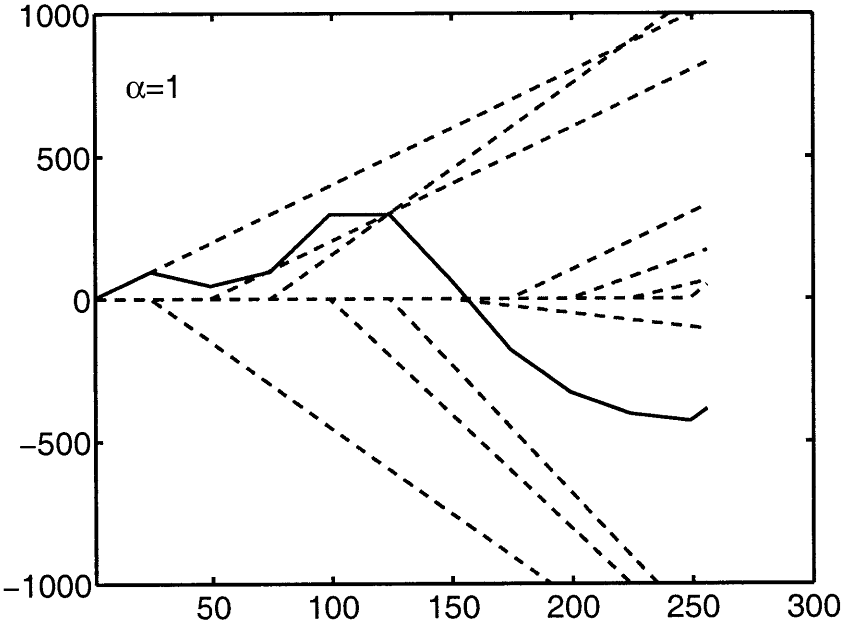

Figure 5-5: Combination of $\alpha = 1$ algebraic singularity functions.

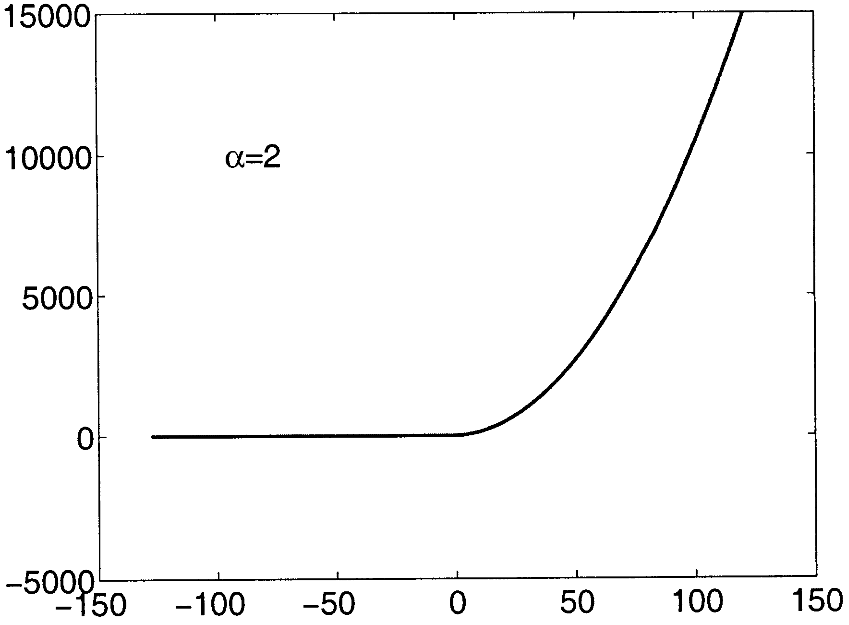

Figure 5-6: Causal algebraic singularity function for $\alpha = 2$.

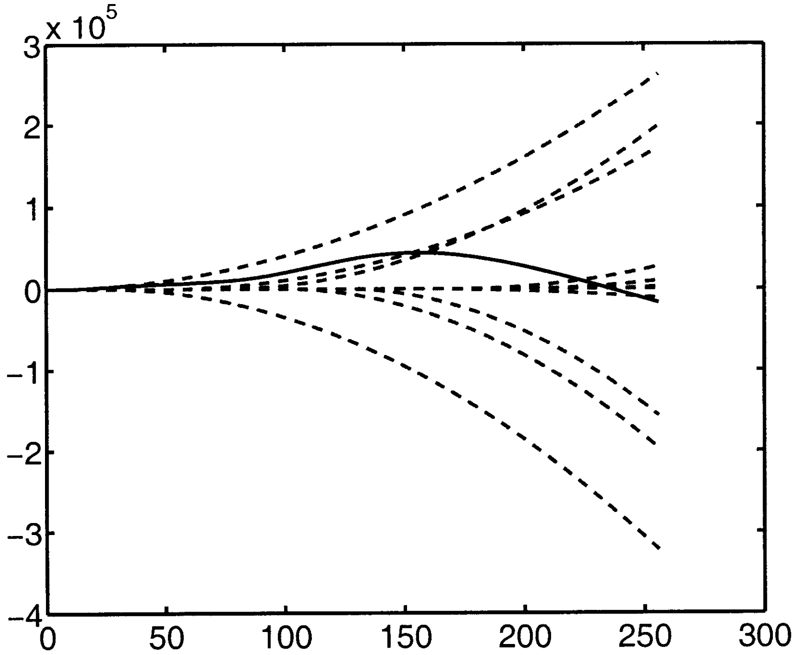

Figure 5-7: Combination of $\alpha = 2$ algebraic singularity functions.

5.2.2 B-splines

A classical result of integer degree splines, $\alpha \in \mathbb{Z}^{+} \cup \{0\}$, is that they are representable in a basis of B-splines (B for basic), $\beta^{\alpha}(x)$, which are bell shaped functions of compact support. These functions can be causal, anti-causal, or symmetric. There are two ways to define B-splines. The first is in terms of finite differences of the algebraic singularity functions defined in Equations 5.31 through 5.35 (Unser and Blu, 2000). For causal algebraic singularity functions we have:

$$\beta_{+}^{\alpha} = \Delta_{+}^{\alpha+1} \chi_{+}^{\alpha}. \tag{5.36}$$

For anti-causal singularity functions we have:

$$\beta_{-}^{\alpha} = \Delta_{-}^{\alpha+1} \chi_{-}^{\alpha}. \tag{5.37}$$

The differencing operator $\Delta^{\alpha+1}$ is defined as:

$$\Delta_{\pm}^{\alpha+1} = \sum_{k\geq 0}(-1)^k \binom{\alpha}{k} f(x \mp k) \tag{5.38}$$

We can also define the symmetric B-spline in terms of a convolution of a causal and an anti-causal B-spline:

$$\beta_{\star}^{\alpha} = \beta_{+}^{\frac{\alpha-1}{2}} * \beta_{-}^{\frac{\alpha-1}{2}}. \tag{5.39}$$

There is, in fact, a corresponding symmetric differencing operator defined in (Unser and Blu, 2000). We keep things simpler by defining the symmetric splines via a convolution of causal and anti-causal functions.

From Equations 5.31, 5.36, and 5.38 we can derive that the 0^{th} degree causal B-spline is in fact the boxcar function on the interval from 0 to 1. In the Fourier domain this becomes the Fourier transform of unity from 0 to 1:

$$\hat{\beta}_{+}^{0}(\omega) = \int_0^1 e^{-i\omega x} dx = \left(\frac{1-e^{-i\omega}}{i\omega}\right). \tag{5.40}$$

The 0^{th} degree anti-causal B-spline is the complex conjugate of the causal one:

$$\hat{\beta}_{-}^{0}(\omega) = \left[\hat{\beta}_{+}^{0}(\omega)\right]^{*}. \tag{5.41}$$

The symmetric 0^{th} degree B-spline is the absolute value of the 0^{th} degree causal B-spline:

$$\hat{\beta}_{\star}^{0}(\omega) = \left|\hat{\beta}_{+}^{0}(\omega)\right|. \tag{5.42}$$

We use the same notation for causality, anti-causality, and symmetry in the B-splines as we did for the algebraic singularity functions. If no subscript is supplied to a spline basis function in an equation in this thesis, it is assumed that the equation applies equally to spline basis functions of any symmetry. We plot the causal 0^{th}

degree B-spline in one and two dimensions in Figures 5-8 and 5-9. We plot symmetric versions of the same function in one and two dimensions in Figures 5-10 and 5-11. The anti-causal version of these functions is just the mirror image of the causal ones.

We see that the symmetric 0^{th} degree B-splines have a much stranger appearance than the causal functions, but they have the same degree of Hölder continuity.

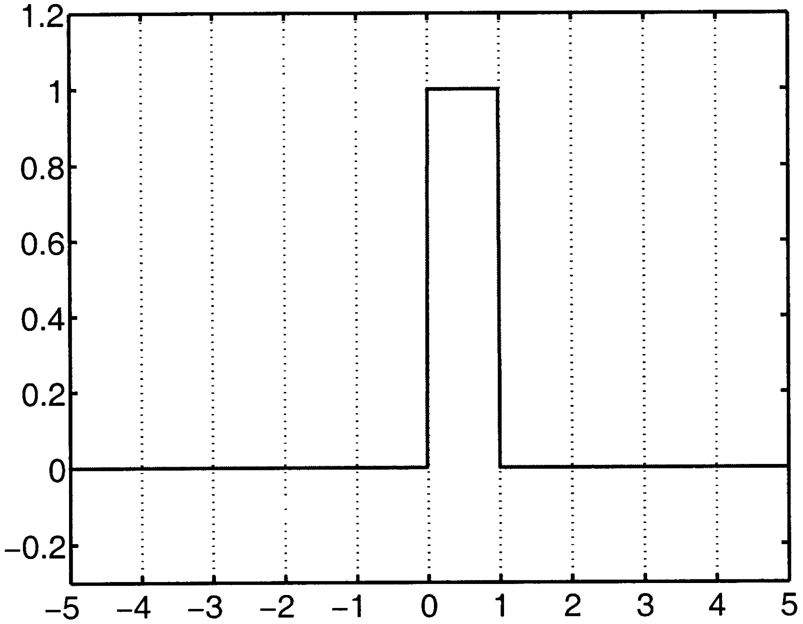

Figure 5-8: Causal 0^{th} degree B-spline.

From Equations 5.40 through 5.42 we can see that higher degree B-splines are created easily in the Fourier domain by exponentiating the 0^{th} degree B-spline to the $\alpha + 1$ power. This corresponds to α auto-convolutions of the 0^{th} degree B-spline with itself in the state domain.[5] We see the 1^{st} degree B-spline constructed in this way from the 0^{th} degree B-spline in Figure 5-12. In the causal case we have the following expression for integer degree B-splines:

$$\hat{\beta}_{+}^{\alpha}(\omega) = \left(\frac{1 - e^{-i\omega}}{i\omega}\right)^{\alpha+1}, \tag{5.43}$$

[5] *State domain* refers to the non-transformed domain.

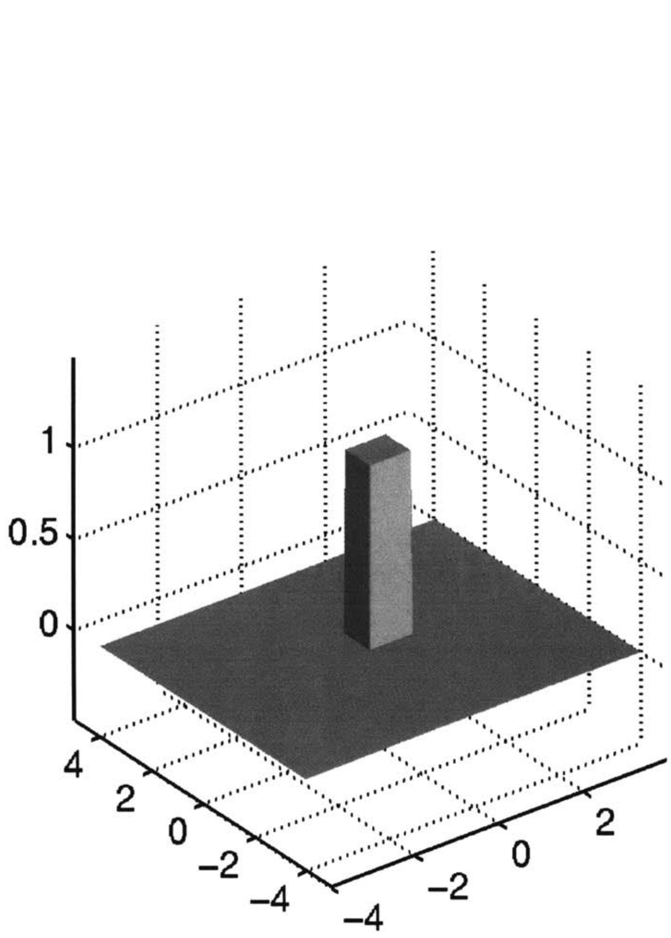

Figure 5-9: 2-D Causal 0^{th} degree spline outer-product basis function.

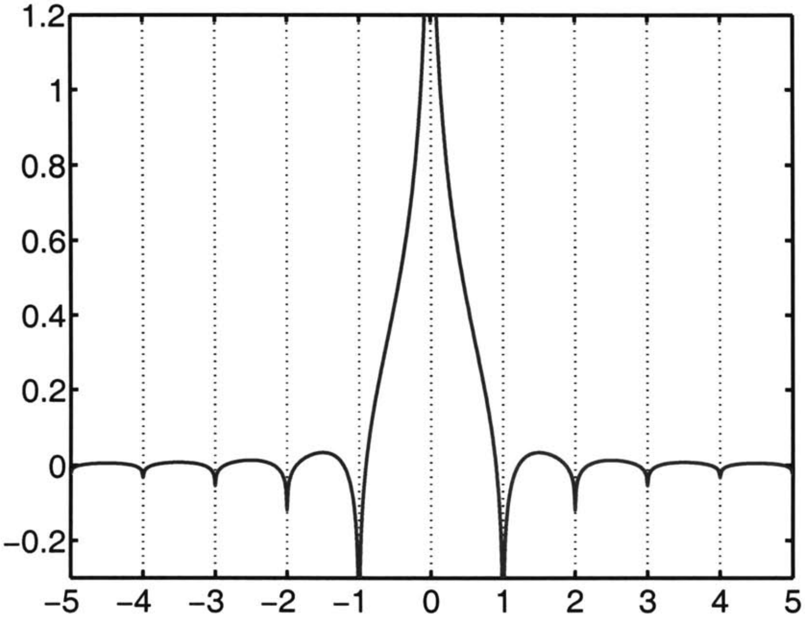

Figure 5-10: Symmetric 0^{th} degree B-spline.

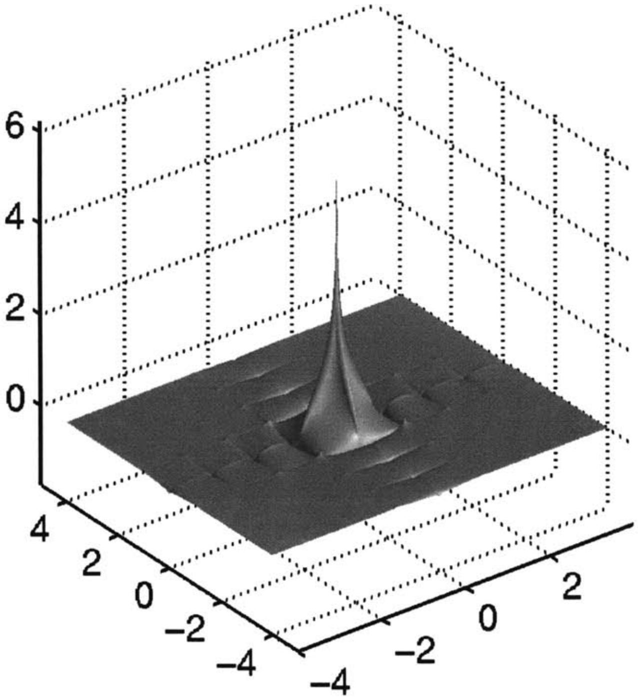

Figure 5-11: 2-D Symmetric 0^{th} degree spline outer-product basis function.

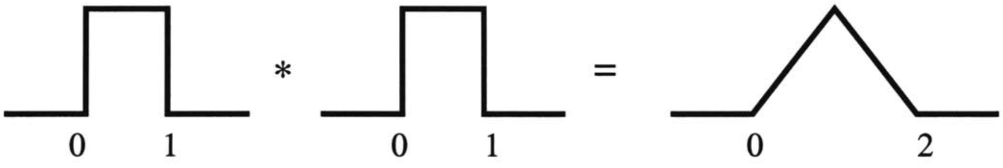

Figure 5-12: Convolution of two causal box car functions to produce a hat function.

in the anti-causal case:

$$\hat{\beta}_{-}^{\alpha}(\omega) = \left(\left[\frac{1-e^{-i\omega}}{i\omega}\right]^{*}\right)^{\alpha+1}, \tag{5.44}$$

and, for the symmetric case:

$$\hat{\beta}_{\star}^{\alpha}(\omega) = \left|\frac{1-e^{-i\omega}}{i\omega}\right|^{\alpha+1}. \tag{5.45}$$

By letting α assume fractional, non-integer values, $\alpha \in \{\mathbb{R} : \alpha > -1/2\}$, we arrive at **fractional B-splines**. The properties of B-splines do not change when α is not an integer with the exception of compact support; they become infinitely supported but decay exponentially rapidly to zero. If computations with the basis are done in the Fourier domain, this does not present any computational difficulties. However, if computations are done in the state domain an integer degree B-spline basis will lead to FIR filters (and sparse matrices) while the non-integer B-splines will lead to IIR filters (full matrices) (see Section 5.3). Thus we see that adjusting the α for the smallest approximation error may lead to much slower computations. This is a trade-off that must be balanced when dealing with discretized data.

We show examples in 1-D of causal and symmetric fractional B-splines for fractionally varying α in Figures 5-13 and 5-14, respectively. Again, although the appearance of the causal and symmetric B-splines differs for non-integer $\alpha \geq 0$, for a given α they have the same degree of Hölder differentiability.

5.2.3 Cardinal splines

Another widely used spline basis function is the **cardinal spline**. Like the B-spline, this basis function is more simply defined in the Fourier domain by (Unser, 1999)

$$\hat{\eta}^{\alpha}(\omega) = \frac{\hat{\beta}^{\alpha}(\omega)}{\sum_k \hat{\beta}^{\alpha}(\omega + 2\pi k)}. \tag{5.46}$$

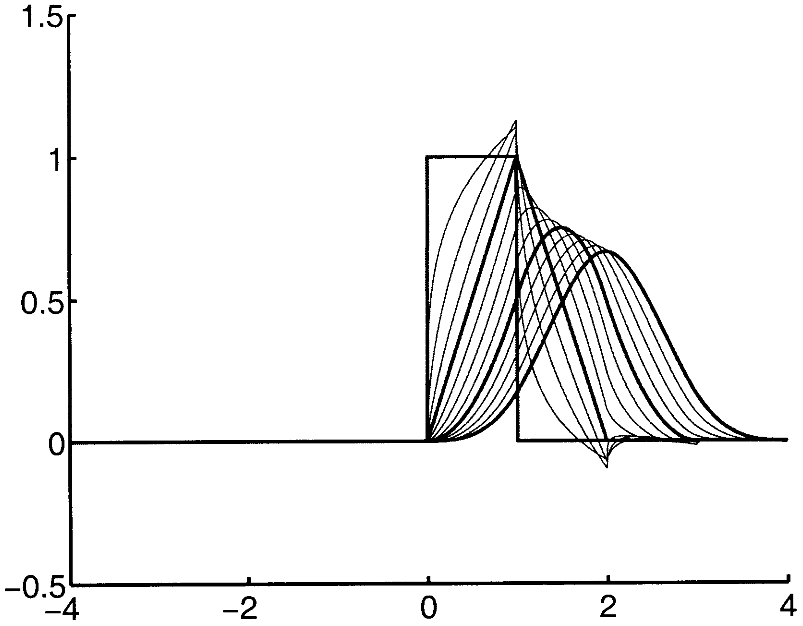

Figure 5-13: Causal B-splines for $\alpha = 0$ to $\alpha = 3$ in steps of 0.25.

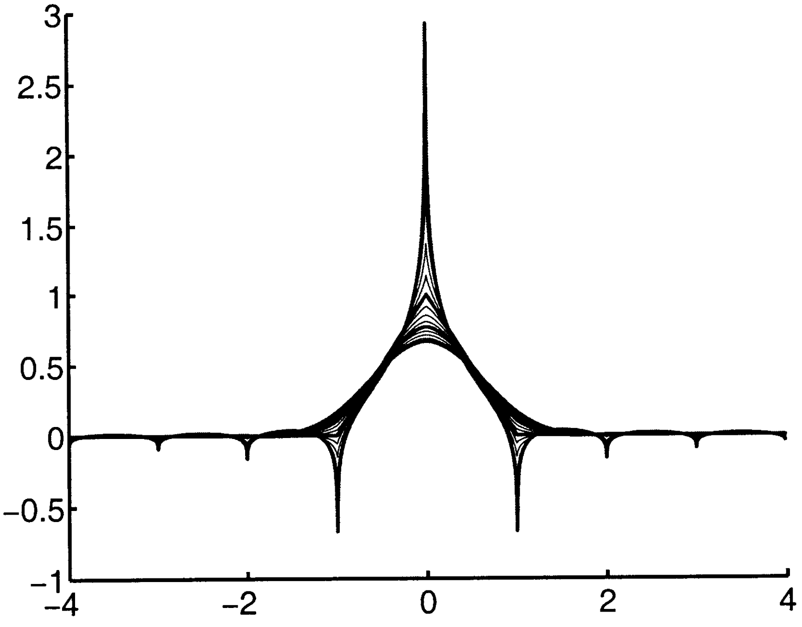

Figure 5-14: Symmetric B-splines for $\alpha = 0$ to $\alpha = 3$ in steps of 0.25.

The denominator of this equation is the sampled version of the B-spline in the frequency domain. By sampling a function we make its Fourier transform 2π-periodic (Mallat, 1998, page 40). Thus we see that a cardinal spline is a B-spline that has been filtered by the inverse of its sampled version.

Cardinal splines are *interpolating functions.* They have the interesting property that, when sampled on a regular grid, the sampled values equal zero everywhere except at a single node, where it equals one. In the limit as $\alpha \mapsto \infty$ the cardinal spline becomes the infinitely smooth sinc function of classical interpolation theory (Aldroubi et al., 1992). Equation 5.46 becomes the boxcar function in the case of the $\alpha = \infty$ sinc function. In Figure 5-15 we show the famous cubic B-spline and in Figure 5-16 we plot its corresponding cardinal spline, obtained by Equation 5.46. We notice that it is zero at all the integers.

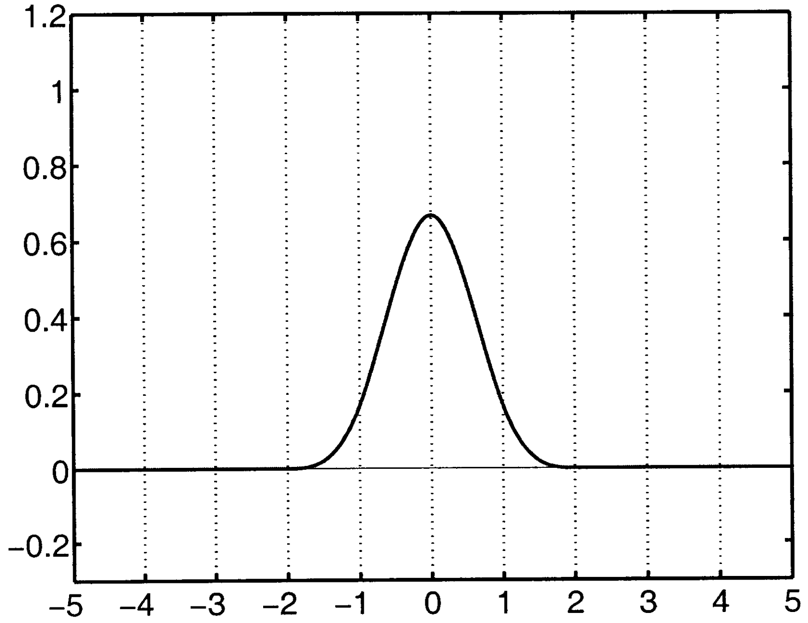

Figure 5-15: Cubic B-spline.

When using interpolation functions as a basis, the coefficients in Equation 5.4 are truly point samples of the approximation functions. Since most scientists prefer their discrete data to represent point samples of the approximation function, cardinal

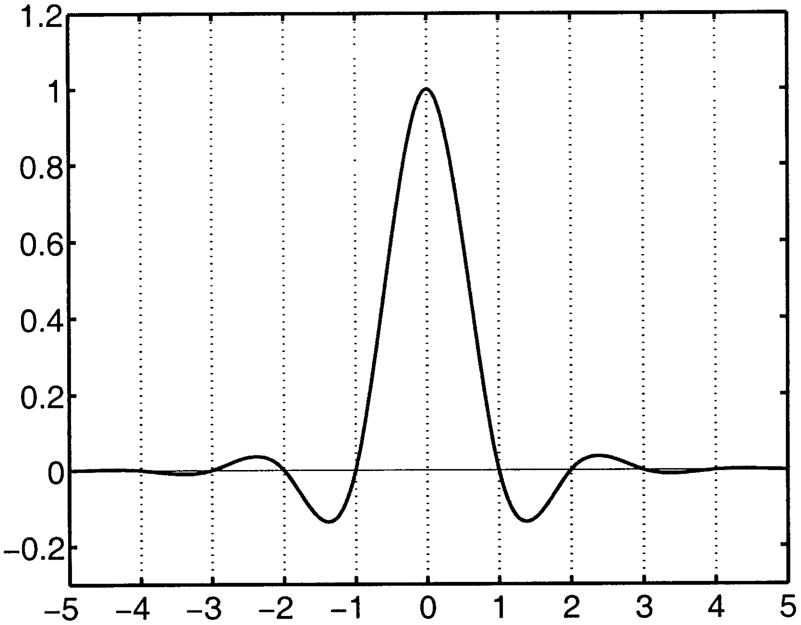

Figure 5-16: Cubic cardinal spline.

splines are a useful basis. For cardinal splines, Equation 5.4 becomes

$$f_S = \sum_k f_S[k]\eta^\alpha(x), \tag{5.47}$$

where $f_S[k]$ are the samples of f_S at the integers. For $\alpha \neq \{0,1\}$ cardinal splines share the drawback with non-integer degree B-splines of not having compact support - leading to IIR filters (non-sparse matrices).

It should be noted that the classical method for handling discrete data was to construct a band-limited approximation function, leading to $\alpha = \infty$ spline basis functions. There are no frequencies above the Nyquist frequency in this case, and Shannon's sampling theorem (Strang, 1986, page 325) proves that the discrete samples of a function are sufficient to characterize the function at all points. Looking at the problem from an approximation theoretic point of view, we see that splines can improve upon the band limited assumption if one has a-prior knowledge of the Hölder continuity of the true function, f. Then approximation theory tells one that a non-band-limited $\alpha < \infty$ degree cardinal spline would be a better basis than the sinc function.

5.2.4 Orthogonal splines

Another spline basis function is the **orthogonal spline** (Unser, 1999). It is defined as

$$\hat{\gamma}^{\alpha}(\omega) = \frac{\hat{\beta}^{\alpha}(\omega)}{\left(\sum_k |\hat{\beta}^{\alpha}(\omega + 2\pi k)|^2\right)^{1/2}} \tag{5.48}$$

Here the B-spline is filtered by the inverse of the square root of the periodic sampled power spectrum[6].

Orthogonal splines have the property that they are orthogonal to integer shifts of themselves. This means that the ℓ_2 energy in the coefficients of such a basis equals the $\mathcal{L}_2$ energy in the approximation function f_S. This will turn out to be a very desirable property when constructing wavelets in chapter 7, where we may desire to make an orthogonal transform. Orthogonal splines share the drawback with non-integer B-splines and cardinal splines of not being compactly supported.

We show a cubic orthogonal spline in Figure 5-17. This should be compared to the cubic B-spline and cubic cardinal spline in Figures 5-15 and 5-16. It looks very similar to the cardinal spline but does not exactly equal zero at the integers, or unity at the center node position.

5.2.5 Dual Splines

For any of the spline basis functions derived above (but not the algebraic singularity functions), one can derive the dual basis function. This dual function spans the same spline space as its associated basis function, and is also integer shift biorthogonal to the basis function. If we let $\phi(x)$ be any spline basis function, its unique dual is

[6] It should be noted that the denominators in both equations 5.46 and 5.48 cannot be evaluated exactly on a digital computer and are instead truncated to two hundred terms when used to generate cardinal or orthogonal splines in this thesis.

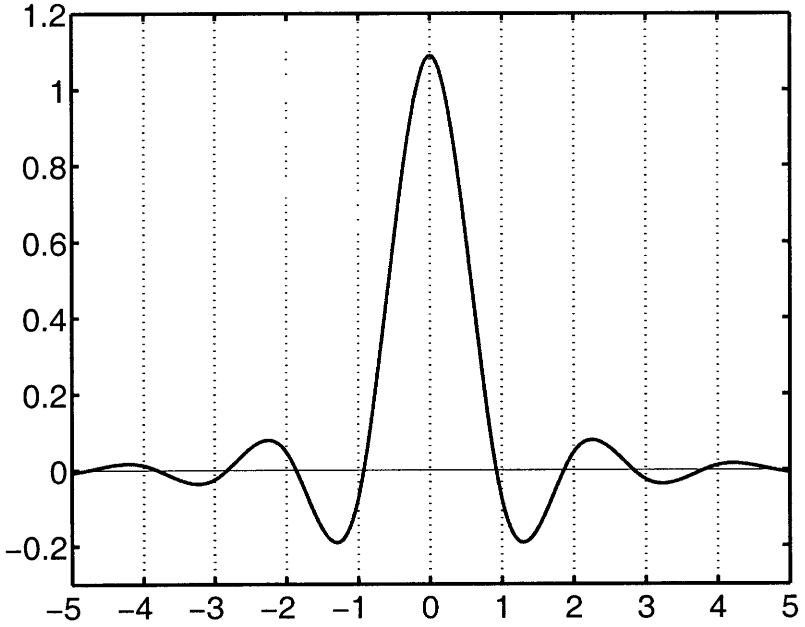

Figure 5-17: Cubic orthogonal spline.

defined in the Fourier domain by

$$\hat{\bar{\phi}}(\omega) = \frac{\hat{\phi}(\omega)}{\sum_k \left[\hat{\phi}(\omega + 2\pi k)\right]^2}. \tag{5.49}$$

It is easily seen that multiplying this function by $\hat{\phi}(\omega)$ and then making it 2π periodic (which is sampling in the Fourier domain) will give unity, thus showing orthogonality. That the dual spans the same spline space is seen by the fact that it is constructed by a discrete filter (linear combination) of the original basis function.

Plugging in β for ϕ we have the following equation for dual B-splines:

$$\hat{\bar{\beta}}(\omega) = \frac{\hat{\beta}(\omega)}{\sum_k \left[\hat{\beta}(\omega + 2\pi k)\right]^2}. \tag{5.50}$$

Representing the dual cardinal spline in terms of B-splines gives

$$\hat{\bar{\eta}}(\omega) = \frac{\hat{\beta}(\omega)\sum_k \hat{\beta}(\omega + 2\pi k)}{\sum_k \left[\hat{\beta}(\omega + 2\pi k)\right]^2}. \tag{5.51}$$

We express this equation in terms of B-splines because of their simple analytical expression. To see that it is orthogonal to $\hat{\eta}(\omega)$, simply multiply by Equation 5.46 and 2π periodize.

Orthogonal functions are their own duals, so we do not have a separate expression for $\hat{\mathring{\gamma}}(\omega)$.

5.3 Discretizing an operator

We have seen how to represent a continuous function with discrete coefficients via projection onto a spline subspace. We now examine how to project linear operators onto the same subspace. We will show that the action of the linear operator D on a function f_S then can be equivalently represented by a discrete matrix operating on a discrete vector. Understanding this topic is of extreme importance to all scientists and mathematicians since one can only represent and manipulate discrete vectors and operators on a digital computer. We will limit ourselves to spline subspaces of $\mathcal{L}_2$, and the case where D is a circular convolutional operator for simplicity.

We let ϕ represent an arbitrary spline basis function. We then have the representation

$$f_S(x) = \sum_k c[k]\phi(x-k). \tag{5.52}$$

Operating on both sides of Equation 5.52 with the operator D we have

$$\begin{aligned} z(x) &= Df_S(x) && (5.53)\\ &= \sum_k c[k]D\phi(x-k). && (5.54)\end{aligned}$$

We immediately see a problem with this equation. z is not a member of the same space as f_S. It is instead spanned by a strange new (perhaps generalized) function $\breve{\phi}(x-k) = D\phi(x-k)$. We may want to stay in the space $\mathcal{S}$ and calculate the coefficients of the projection of z onto $\mathcal{S}$; that is, we want to calculate the coefficients

$d[l]$ in the representation

$$z_S(x) = \sum_l d[l]\phi(x - l). \tag{5.55}$$

Calculating $d[l]$ involves convolving the dual of ϕ with z followed by sampling:

$$\begin{aligned} d[l] &= \delta_c(x') \int \bar{\phi}(x - x') \left[\sum_k c[k] D\phi(x - k) \right] dx & (5.56) \\ &= \sum_k c[k] \left[\delta_c(x') \int \bar{\phi}(x - x') D\phi(x - k) dx \right]. & (5.57) \end{aligned}$$

It is clearer to see this process in the Fourier domain. Since D is a circular convolution operator, it can be operated in this domain by multiplication of its spectrum, $\hat{D}(\omega)$:

$$\sum_k c[k] \left[\delta_c(x') \int \bar{\phi}(x - x') D\phi(x - k) dx \right] \Longleftrightarrow \sum_n \hat{D}_S(\omega + 2\pi n)\hat{c}(\omega), \tag{5.58}$$

where

$$\hat{D}_S(\omega) = \hat{\bar{\phi}}(\omega)\hat{D}(\omega)\hat{\phi}(\omega) \tag{5.59}$$

This representation needs explaining. In Equation 5.57 there is a convolution of D with the basis functions, and then with the dual functions. This gives rise to the 3 terms in Equation 5.59. The resulting operator is then convolved with the coefficients, $c[k]$, and then sampled, which makes the 2π periodic Fourier domain operator, $\sum_n \hat{D}_S(\omega + 2\pi n)$, of Equation 5.58. The coefficients, $c[k]$, were already a sampled function in the state domain, so $\hat{c}(\omega) = \hat{c}(\omega + 2\pi n)$. $\sum_n \hat{D}_S(\omega + 2\pi n)$ represents a discrete filtering operation, a.k.a., a circular convolution matrix which we denote as $\mathbf{D}_S$. We can now express the infinite dimensional operation

$$z_S = D_S f_S \tag{5.60}$$

equivalently by the finite dimensional operation

$$\mathbf{d} = \mathbf{D}_{\mathcal{S}}\mathbf{c}. \tag{5.61}$$

$\mathbf{c}$ are the coefficients of $f_{\mathcal{S}}$ in the spline basis, ϕ_k; $\mathbf{d}$ are the coefficients of $z_{\mathcal{S}}$ in the same basis; $\mathbf{D}_{\mathcal{S}}$ are the coefficients of the linear operator K mapping $f_{\mathcal{S}}$ to $z_{\mathcal{S}}$.

We end this section with an example of the above process. The operator, D, that we use will be the derivative operator. $\mathbf{D}_{\mathcal{S}}$ will then be a differencing matrix. What the coefficients of this matrix are will depend on the spline space we are in. We will let the coefficient vector $\mathbf{c}$ be velocity log shown in figure 5-1. We don't know what degree α to use - that is, we don't know what spline space is optimal for representing this data. This doesn't matter for our example. We simply want to show the effect of making different assumptions. If different assumptions lead to drastically different vectors $\mathbf{d}$ in Equation 5.61, then we know that a correct modeling of the discrete data is crucial to obtaining accurate results.

We generate 3 different matrices $\mathbf{D}_{\mathcal{S}}$. One assuming $\alpha = 0$, the next $\alpha = 1$, and, finally, $\alpha = \infty$. In Figures 5-18, 5-19, and 5-20 we show the impulse responses of each of these matrices, respectively. Their appearances are significantly different. The first is the first differencing operator commonly used to approximate the derivative of discrete data. The third is basically the Fourier transform of the derivative operator $\frac{d}{dx} \longleftrightarrow i\omega$. The second vector is somewhere in between. To better illustrate the difference between these operators we plot the absolute value of their Fourier transforms in Figure 5-21 for all three values of α.

Letting $\mathbf{c}$ be the well log data shown in Figure 5-1, we show the $\mathbf{d}$ vector resulting from Equation 5.61 for both $\alpha = 0$ and $\alpha = \infty$ operators in Figure 5-22. Figure 5-23 shows the difference between the $\mathbf{d}$'s plotted in Figure 5-22. From this last plot we see that the difference is of roughly the same magnitude as the vectors themselves. This is a substantial difference. If one spline space is optimal and the discrete data is treated as though it belonged to a different spline space, large error can be unnecessarily incurred.

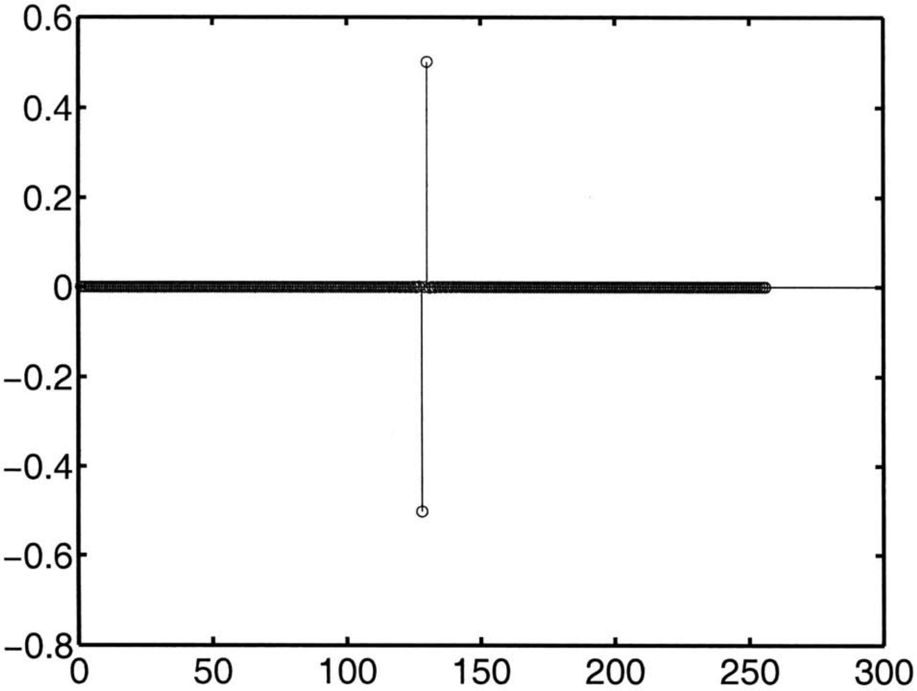

Figure 5-18: Impulse response of derivative matrix, $\mathbf{D}_{\mathcal{S}}$, in spline basis for $\alpha = 0$.

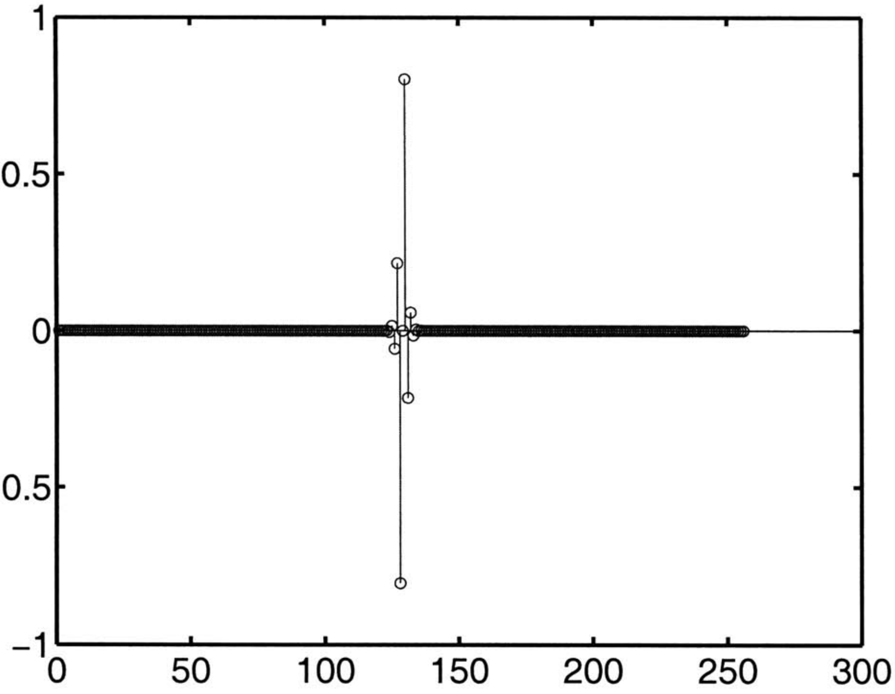

Figure 5-19: Impulse response of derivative matrix, $\mathbf{D}_{\mathcal{S}}$, in spline basis for $\alpha = 1$.

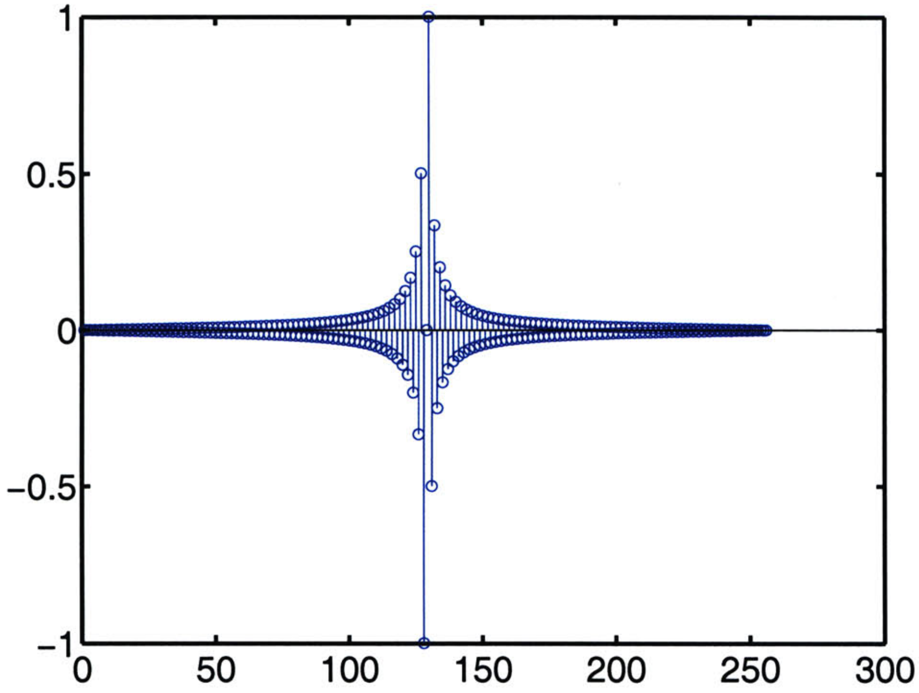

Figure 5-20: Impulse response of derivative matrix, $\mathbf{D}_{\mathcal{S}}$, in spline basis for $\alpha = \infty$.

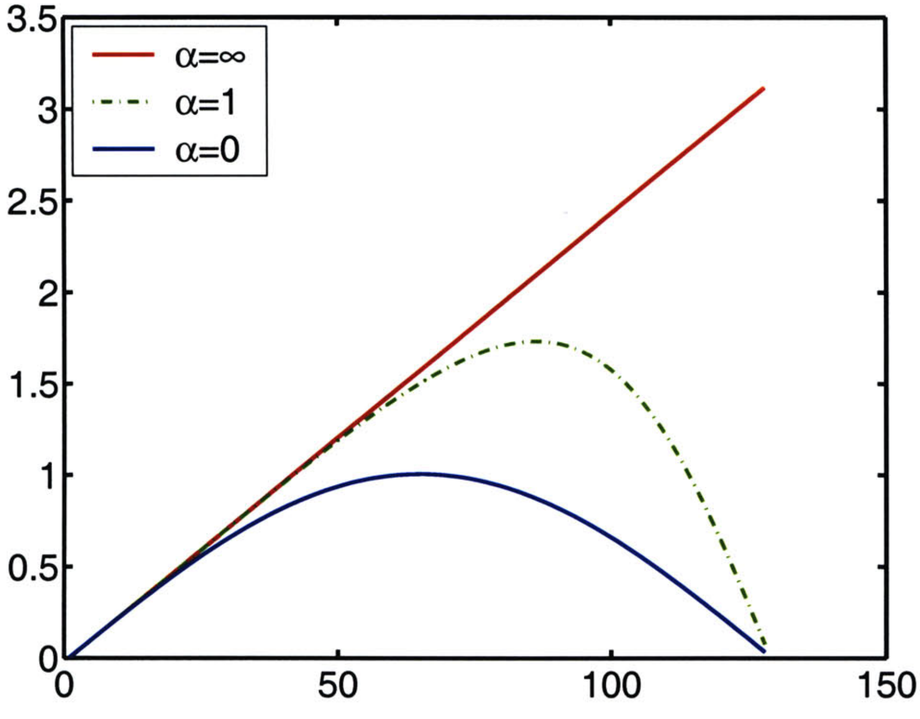

Figure 5-21: Frequency domain derivative operator, $\hat{D}_{\mathcal{S}}$, for all α's.

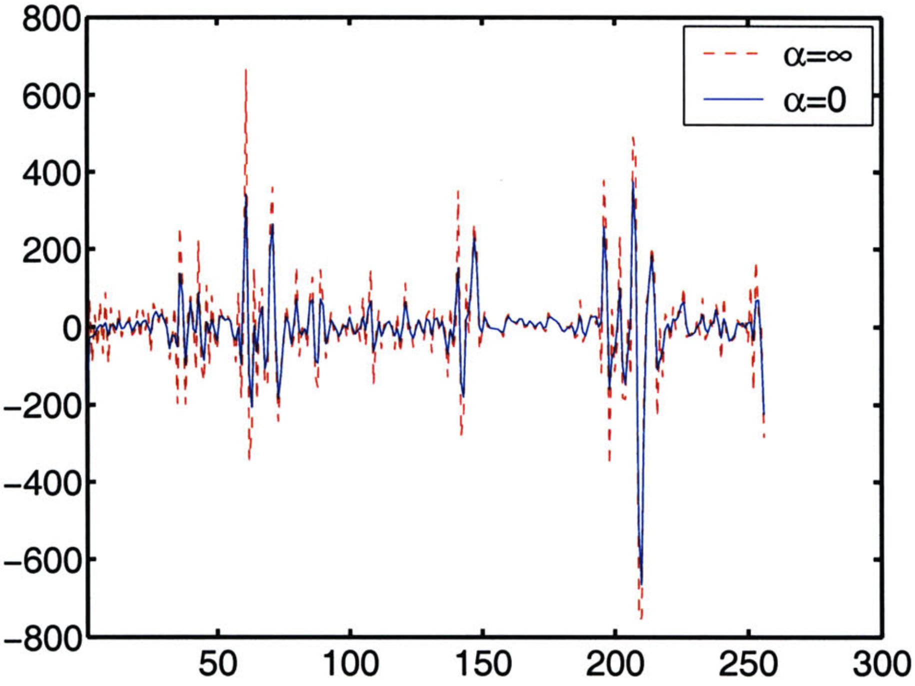

Figure 5-22: Results of operating $\mathbf{D}_{\mathcal{S}}\mathbf{c}$ for $\alpha = 0$ and $\alpha = \infty$.

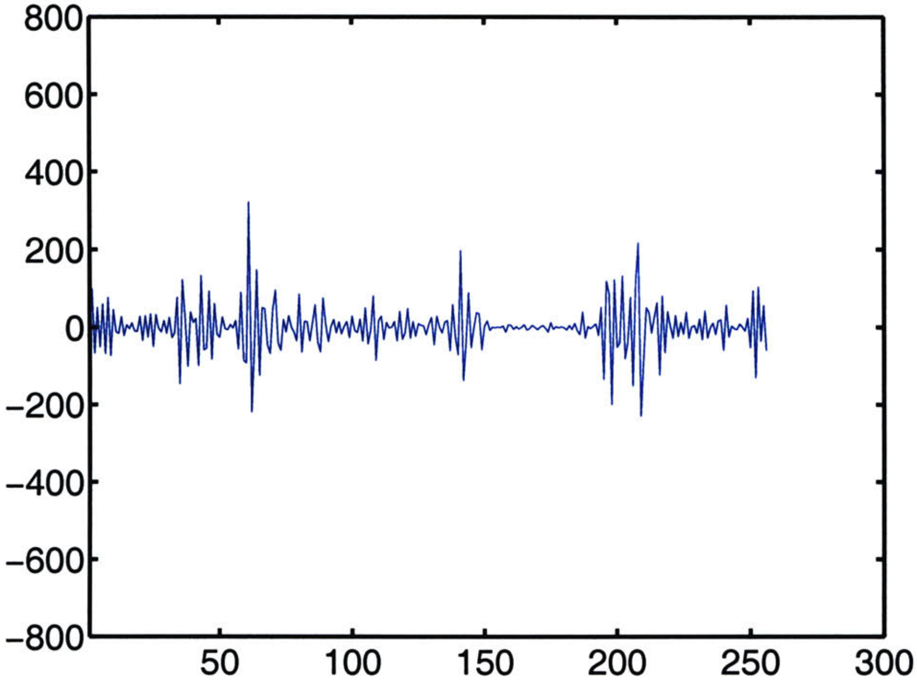

Figure 5-23: Difference in the results of operating $\mathbf{D}_{\mathcal{S}}\mathbf{c}$ for both α's.

Chapter 6

Wavelets

Wavelets are localized oscillatory functions, i.e. "wiggles" that have most of their energy at a specific point in space[1]. They are also localized in the frequency domain. They must integrate to zero to be a proper wavelet, and, depending on the wavelet, have higher order moments equal to zero also.

Many real world signals consist of a few isolated jumps (edges) separated by larger smooth regions. Wavelets represent a new and more efficient way to analyze and synthesize such non-stationary functions than classical bases (Donoho, 1995). They have proved useful in applications ranging from image compression (DeVore et al., 1992) to inverse problems (Kolaczyk, 1994, 1996; Donoho, 1995).

To better understand the nature of wavelets we must first define **multiresolutions** (Mallat, 1998).

6.1 Multiresolutions

Let $\{\mathcal{V}_j\}, j \in \mathbb{Z}$, be a sequence of closed subspaces of $\mathcal{L}_2$. According to Mallat (1998), the $\mathcal{V}_j$'s are a multiresolution approximation if the following properties are satisfied:

1. $\forall (j,k) \in \mathbb{Z} \times \mathbb{Z},\ f(x) \in \mathcal{V}_j \longleftrightarrow f(x - 2^j k) \in \mathcal{V}_j,$

[1] Since the independent variable could be either space or time we will just refer to it as space for simplicity.

2. $\forall j \in \mathbb{Z},\ \mathcal{V}_{j+1} \subset \mathcal{V}_j$,

3. $\forall j \in \mathbb{Z},\ f(x) \in \mathcal{V}_j \longleftrightarrow f(\frac{x}{2}) \in \mathcal{V}_{j+1}$,

4. $\lim_{j \to +\infty} \mathcal{V}_j = \bigcap_{-\infty}^{+\infty} \mathcal{V}_j = \{0\}$,

5. $\lim_{j \to -\infty} \mathcal{V}_j = \text{Closure}(\bigcup_{-\infty}^{+\infty} \mathcal{V}_j) = \mathcal{L}^2(\mathbb{R})$.

6. There exists a function ϕ such that $\phi(x-k)$, $k \in \mathbb{Z}$, is a Riesz basis of $\mathcal{V}_0$.

Blu and Unser (2001b) have shown that if the function ϕ satisfies the *partition of unity* constraint

$$\sum_{k \in \mathbb{Z}} \phi(x-k) = 1, \tag{6.1}$$

in addition to properties 3 and 6, then all the properties of a multiresolution are satisfied, and ϕ is a valid **scaling function**. This allows for a simpler description of a multiresolution. The scaling function can be used to approximate continuous functions as was done in Chapter 5, and, in fact, we will use spline basis functions as our scaling functions below.

Multiresolutions gives a mathematically precise way of defining the intuitive concepts of scale and resolution. We will follow Mallat (1998) and define scale to be the number 2^j and resolution, its inverse, the number 2^{-j}. We will call j itself the *scale level*[2]. Scale $2^0 = 1$ usually corresponds to the scale at which a regular grid is defined. An arbitrary function in $\mathcal{L}_2$ may have details at very small scales and the Riesz basis at scale 1 might be inadequate to completely represent the function. Therefore we usually refer to a representation of a function at a certain scale as an **approximation**.

[2] There are at least two conventions in the literature as to the definition of **scale**. Some follow Choi and Baraniuk (1999), Strang and Nguyen (1997), and Donoho (1995) who use the opposite definitions of scale and resolution than Mallat (1998) uses. Some workers (Li et al., 1996) define scale by j, which we call *scale level.* All these different conventions leads to conflicting and confusing notations in equations involving wavelets. We believe that the notation of Mallat (1998) is the most intuitive and easy to use and thus follow it in this work.

We are now ready to make a precise definition of a wavelet. Let $\mathcal{W}_j$ be the *orthogonal complement* of $\mathcal{V}_j$ on $\mathcal{V}_{j-1}$ (Naylor and Sell, 1982):

$$\mathcal{V}_j = \mathcal{V}_{j+1} \oplus \mathcal{W}_{j+1}. \tag{6.2}$$

The symbol $\oplus$ implies a direct sum of two linear spaces (Naylor and Sell, 1982). We interpret this as saying that $\mathcal{V}_j$ has approximations of functions by a Riesz basis at level j and $\mathcal{W}_j$ contains the details necessary to represent the function at level $j-1$. Equation 6.2 implies a recursive relationship if we expand the $\mathcal{V}$'s in terms of $\mathcal{W}$'s. Starting at the smallest scale of interest, $\mathcal{V}_0$, we have:

$$\mathcal{V}_0 = \mathcal{W}_1 \oplus \mathcal{W}_2 \oplus \cdots \oplus \mathcal{W}_N \oplus \mathcal{V}_N, \tag{6.3}$$

where $\mathcal{V}_N$ is the largest scale space we care to define. It is in the spaces $\mathcal{W}_j$ that wavelets reside. They can be interpreted as carrying the details of functions between consecutive scales.

We can define an analyzing function, $\bar{\phi}$, that is biorthogonal to integer shifts of $\bar{\phi}$:

$$\delta[n-k] = \left\langle \phi(x-n), \bar{\phi}(x-k) \right\rangle. \tag{6.4}$$

Since ϕ_k is a Riesz basis, $\bar{\phi}$ could be the dual function (Section 5.1.1), but it does not have to be. Any function satisfying the biorthogonality constraint in Equation 6.4 will do. If $\bar{\phi}$ is the dual, then $\bar{\phi}$ is also in $\mathcal{V}_0$, and the relationships between subspaces in Equations 6.2 and 6.3 apply to it.

If $\bar{\phi}$ is not the dual of ϕ but satisfies the necessary requirements for a scaling function, it will have its own dual multiresolution spaces, $\bar{\mathcal{V}}_j$ and $\bar{\mathcal{W}}_j$, defined by the recursion

$$\bar{\mathcal{V}}_j = \bar{\mathcal{V}}_{j+1} \oplus \bar{\mathcal{W}}_{j+1}. \tag{6.5}$$

If expanded, this leads to

$$\bar{\mathcal{V}}_0 = \bar{\mathcal{W}}_1 \oplus \bar{\mathcal{W}}_2 \oplus \cdots \oplus \bar{\mathcal{W}}_N \oplus \bar{\mathcal{V}}_N. \tag{6.6}$$

It can be shown (Mallat, 1998) that $\mathcal{V}_j$ is orthogonal to $\bar{\mathcal{W}}_j$, but not to $\mathcal{W}_j$, and that $\bar{\mathcal{V}}_j$ is orthogonal to $\mathcal{W}_j$, but not to $\bar{\mathcal{W}}_j$.

By the above partition of function spaces we assume an arbitrary $f_{\mathcal{V}_j}(x) \in \mathcal{V}_j$ to be the sum of two functions, $f_{\mathcal{V}_{j+1}}(x) \in \mathcal{V}_{j+1}$ and $f_{\mathcal{W}_{j+1}}(x) \in \mathcal{W}_{j+1}$. $f_{\mathcal{V}_{j+1}}$ is representable in a basis of dilated scaling functions $\frac{1}{\sqrt{2^{j+1}}}\phi(\frac{x}{2^{j+1}} - k) \in \mathcal{V}_{j+1}$, while $f_{\mathcal{W}_{j+1}}$ has as a basis another set of dilated functions $\frac{1}{\sqrt{2^{j+1}}}\psi(\frac{x}{2^{j+1}} - k) \in \mathcal{W}_{j+1}$. We thus have the following equivalent decompositions:

$$\begin{aligned}
f_{\mathcal{V}_j}(x) &= \sum_n a_j[n]\frac{1}{\sqrt{2^j}}\phi(\frac{x}{2^j} - n) && (6.7)\\
&= f_{\mathcal{V}_{j+1}}(x) + f_{\mathcal{W}_{j+1}}(x) && (6.8)\\
&= \sum_k a_{j+1}[k]\frac{1}{\sqrt{2^{j+1}}}\phi(\frac{x}{2^{j+1}} - k) + \sum_k d_{j+1}[k]\frac{1}{\sqrt{2^{j+1}}}\psi(\frac{x}{2^{j+1}} - k). && (6.9)
\end{aligned}$$

The functions ψ will be our wavelets. We will call $a_j[k]$ the *approximation* coefficients, and $d_j[k]$ the *detail* coefficients, at scale level j. To determine these coefficients we must decompose $f_{\mathcal{V}_j}(x)$ with the analyzing functions $\frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}} - l)$ and $\frac{1}{\sqrt{2^{j+1}}}\bar{\psi}(\frac{x}{2^{j+1}} - l)$. Taking the inner product of Equation 6.9 with $\frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}} - l)$, and observing that functions in $\bar{\mathcal{V}}_{j+1}$ are orthogonal to functions in $\mathcal{W}_{j+1}$, we have:

$$\begin{aligned}
&\left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}} - l), f_{\mathcal{V}_j}(x) \right\rangle\\
&= \sum_k a_{j+1}[k]\left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}} - l), \frac{1}{\sqrt{2^{j+1}}}\phi(\frac{x}{2^{j+1}} - k) \right\rangle\\
&\quad + \sum_k d_{j+1}[k]\left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}} - l), \frac{1}{\sqrt{2^{j+1}}}\psi(\frac{x}{2^{j+1}} - k) \right\rangle && (6.10)\\
&= \sum_k a_{j+1}[k]\delta[l - k] && (6.11)\\
&= a_{j+1}[l]. && (6.12)
\end{aligned}$$

By the same line of reasoning we take the inner product of Equation 6.9 with $\frac{1}{\sqrt{2^{j+1}}}\bar{\psi}(\frac{x}{2^{j+1}} - l)$ to obtain the detail coefficients:

$$\left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\psi}(\frac{x}{2^{j+1}} - l), f_{V_j}(x) \right\rangle = d_{j+1}[l]. \tag{6.13}$$

If we take the inner product of Equation 6.7 with $\frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}} - l)$ we have:

$$\begin{aligned} &\left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}} - l), f_{V_j}(x) \right\rangle \\ &= \sum_n a_j[n] \left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}} - l), \frac{1}{\sqrt{2^j}}\phi(\frac{x}{2^j} - n) \right\rangle . \end{aligned} \tag{6.14}$$

Making the change of variables $x' = \frac{x}{2^j} - 2l$ and writing the inner product as an explicit integral we obtain

$$\left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}} - l), f_{V_j}(x) \right\rangle$$

$$= \sum_n a_j[n] \int_{-\infty}^{\infty} \frac{1}{\sqrt{2^{j+1}}}\bar{\phi}^*(\frac{x'}{2})\frac{1}{\sqrt{2^j}}\phi(x' - n + 2l)2^j dx' \tag{6.15}$$

$$= \sum_n a_j[n] \left\langle \frac{1}{\sqrt{2}}\bar{\phi}(\frac{x'}{2}), \phi(x' - n + 2l) \right\rangle \tag{6.16}$$

$$= \sum_n a_j[n]\bar{h}[n - 2l]. \tag{6.17}$$

The filter

$$\bar{h}[n] = \left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}}), \frac{1}{\sqrt{2^j}}\phi(\frac{x}{2^j} - n) \right\rangle \tag{6.18}$$

is extremely important to wavelet theory and is known as the (analysis) **refinement filter**. Using these coefficients, the optimal representation of $\frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}})$ in terms of $\frac{1}{\sqrt{2^j}}\bar{\phi}(\frac{x}{2^j})$ is given by

$$\frac{1}{\sqrt{2^{j+1}}}\bar{\phi}(\frac{x}{2^{j+1}}) = \sum_n \bar{h}[n]\frac{1}{\sqrt{2^j}}\bar{\phi}(\frac{x}{2^j} - n). \tag{6.19}$$

We can see that this equation expresses a scaling function at a scale level $j+1$ in terms of linear combinations of the same function at scale level j. It can also be shown that there exists a synthesis refinement filter defined by the relationship

$$\frac{1}{\sqrt{2^{j+1}}}\phi(\frac{x}{2^{j+1}}) = \sum_n h[n]\frac{1}{\sqrt{2^j}}\phi(\frac{x}{2^j}-n). \tag{6.20}$$

If we now take the inner product of Equation 6.7 with $\bar{\psi}(\frac{x}{2^{j+1}}-l)$ and follow the same line of reasoning as for Equations 6.14 through 6.17 we obtain

$$\begin{aligned}
&\left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\psi}(\frac{x}{2^{j+1}}-l), f_{V_j}(x)\right\rangle \\
&= \sum_n a_j[n]\left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\psi}(\frac{x}{2^{j+1}}-l), \frac{1}{\sqrt{2^j}}\phi(\frac{x}{2^j}-n)\right\rangle &\quad (6.21)\\
&= \sum_n a_j[n]\left\langle \frac{1}{\sqrt{2}}\bar{\psi}(\frac{x'}{2}), \phi(x'-n+2l)\right\rangle &\quad (6.22)\\
&= \sum_n a_j[n]\bar{g}[n-2l]. &\quad (6.23)
\end{aligned}$$

The new analysis filter

$$\bar{g}[n] = \left\langle \frac{1}{\sqrt{2^{j+1}}}\bar{\psi}(\frac{x}{2^{j+1}}), \frac{1}{\sqrt{2^j}}\phi(\frac{x}{2^j}-n)\right\rangle \tag{6.24}$$

is what actually defines the analysis wavelets. Using these coefficients we can represent $\frac{1}{\sqrt{2^{j+1}}}\bar{\psi}(\frac{x}{2^{j+1}})$ by $\frac{1}{\sqrt{2^j}}\bar{\phi}(\frac{x}{2^j}-n)$:

$$\frac{1}{\sqrt{2^{j+1}}}\bar{\psi}(\frac{x}{2^{j+1}}) = \sum_n \bar{g}[n]\frac{1}{2^j}\bar{\phi}(\frac{x}{2^j}-n). \tag{6.25}$$

There is a similar synthesis filter $g[n]$ that defines synthesis wavelets via the equation

$$\frac{1}{\sqrt{2^{j+1}}}\psi(\frac{x}{2^{j+1}}) = \sum_n g[n]\frac{1}{2^j}\phi(\frac{x}{2^j}-n). \tag{6.26}$$

This shows that wavelets at scale level $j+1$ are linear combinations of the scaling functions at scale level j.

The scaling functions determine the filters h and $\bar{h}$. We will derive g and $\bar{g}$ from these filters in Section 6.3, below.

6.2 The discrete wavelet transform

In the last section we concentrated on defining scaling functions and wavelets and showing their relationships across consecutive scales. Here we concentrate on analyzing a function $f(x)$ by inner products with dilated and translated analysis wavelets $\bar{\psi}_j \in \bar{\mathcal{W}}_j$ at all scales, and translated analysis scaling functions $\bar{\phi}_J \in \bar{\mathcal{V}}_J$ at a large scale level, J. This analysis procedure is called the discrete wavelet transform[3] (DWT). The analysis part of this transform lies in Equations 6.10 and 6.13. Re-writing the inner products explicitly as integrals we have:

$$a_J[k] = \int f(x) \left[\frac{1}{\sqrt{2^J}}\bar{\phi}(\frac{x}{2^J} - k)\right]^* dx, \tag{6.27}$$

$$d_j[k] = \int f(x) \left[\frac{1}{\sqrt{2^j}}\bar{\psi}(\frac{x}{2^j} - k)\right]^* dx, \tag{6.28}$$

where $a_J[k]$ and $d_j[k]$ are vectors of discrete wavelet coefficients. The complex conjugation of the analysis functions applies to the general case that they have imaginary components (i.e. they are complex scaling functions and wavelets). We saw in Section 5.1.4 that, if $f(x)$ is a periodically repeating function, only a finite number of coefficients are needed to represent the continuous function. The same is true with the DWT. If one period of the function is of length 2^J there will be 2^{J-j} wavelet coefficients at each scale level, and only one scaling function coefficient at scale level J.

We can reconstruct the projection of $f(x)$ onto $\mathcal{V}_0$ via Equation 6.9:

$$f_{\mathcal{V}_0}(x) = \sum_k a_J[k]\frac{1}{\sqrt{2^J}}\phi(\frac{x}{2^J} - k) + \sum_{j=1}^{J}\sum_k d_j[k]\frac{1}{\sqrt{2^j}}\psi(\frac{x}{2^j} - k). \tag{6.29}$$

[3] What is *discrete* about the discrete wavelet transform is that it takes continuous functions and equivalently represents them in terms of a discrete series of coefficients (although possibly infinitely many of them).

As the scale 2^j gets larger, the wavelet coefficients measure lower and lower frequency content in the signal. The periodized dilated scaling function at the largest scale is the constant function and measures the zero frequency of $f(x)$. It is constant due to the partition of unity property of the scaling function; i.e., at the largest scale the sum of the periodic scaling functions sums to a constant number according to Equation 6.1.

We saw in Chapter 5 that operations on a continuous function spanned by a basis functions ϕ_k can be equivalently done via operations on the discrete coefficients of the basis functions. This also holds for the DWT. If we project the wavelets, scaling functions, and $f(x)$ all into the same basis ϕ_k, then the DWT involves discrete vector multiplications and summation. For a periodic function with $N = 2^J$ coefficients, the DWT produces N coefficients and can be represented by a square matrix. We plot such a matrix in Figure 6-1. This particular DWT uses the Haar basis as ϕ_k (Strang and Nguyen, 1997). Each row is the projection of a continuous Haar wavelet into the Haar basis. The first $N/2$ rows contain double shifted discretized wavelets at scale level $j = 1$; the next $N/4$ rows contain quadruple shifted wavelets at scale level $j = 2$, and so on. The very last row contains the discretized constant function.

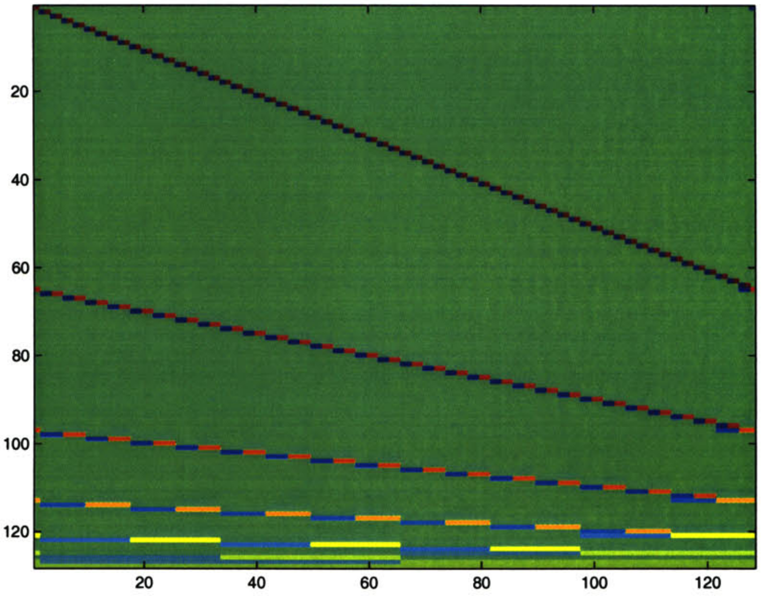

Figure 6-1: Wavelet transform matrix.

If one has the approximation coefficients of $f(x)$ in a basis other than ϕ_k and wishes to perform the DWT with this matrix, *prefiltering* (Section 5.1.2) must first be applied to the vector, otherwise one is committing a "wavelet crime"(Strang and Nguyen, 1997, page 232).

In the next section we see that one does not need to construct a matrix to perform the analysis portion of the DWT. It can be equivalently done by an $O(N)$ discrete filtering and downsampling operation. To do this we need the discrete filters h, $\bar{h}$, g, and $\bar{g}$.

6.3 Filter Banks

Combining Equations 6.12 and 6.17 we see the interesting relationship

$$a_{j+1}[l] = \sum_n a_j[n]\bar{h}[n-2l]. \tag{6.30}$$

From this equation we see that the approximation coefficients at scale level $j+1$ are equal to a correlation of the the approximation coefficients at scale level j with the analysis refinement filter, $\bar{h}$, followed by a downsampling. If we combine Equations 6.13 and 6.23 we get a similar relationship for detail coefficients:

$$d_{j+1}[l] = \sum_n a_j[n]\bar{g}[n-2l]. \tag{6.31}$$

This recursive formulation provides us with the approximation and detail coefficients at scale $j+1$ without ever having to evaluate the inner product of a wavelet or scaling function with the function being analyzed. One only need the approximation coefficients of the scaling functions in $\mathcal{V}_j$ to apply the algorithm.

A similar reconstruction formula can also be derived in terms of discrete coefficients. Returning to Equations 6.20 and 6.26 we observe that functions in $\mathcal{V}_{j+1}$ and $\mathcal{W}_{j+1}$ both also have a basis in $\frac{1}{\sqrt{2^j}}\phi(\frac{x}{2^j}-k)$. We can therefore decompose $f_{\mathcal{V}_{j+1}}(x)$

and $f_{W_{j+1}}(x)$ via inner products with the analyzing function $\frac{1}{\sqrt{2^j}}\bar{\phi}(\frac{x}{2^j}-k)$:

$$\begin{aligned}
a_j[n] &= \left\langle \frac{1}{\sqrt{2^j}}\bar{\phi}(\frac{x}{2^j}-k), f(x) \right\rangle && (6.32)\\
&= \sum_k a_{j+1}[k] \left\langle \frac{1}{\sqrt{2^j}}\bar{\phi}(\frac{x}{2^j}-k), \frac{1}{\sqrt{2^{j+1}}}\phi(\frac{x}{2^{j+1}}-k) \right\rangle + \\
&\quad \sum_k d_{j+1}[x] \left\langle \frac{1}{\sqrt{2^j}}\bar{\phi}(\frac{x}{2^j}-k), \frac{1}{\sqrt{2^{j+1}}}\psi(\frac{x}{2^{j+1}}-k) \right\rangle . && (6.33)
\end{aligned}$$

Making the same change of variables as in Equations 6.15 and 6.21 we have

$$\begin{aligned}
a_j[n] &= \sum_k a_{j+1}[k] \left\langle \frac{1}{\sqrt{2}}\phi(\frac{x'}{2}), \bar{\phi}(x'-n+2k) \right\rangle + \\
&\quad \sum_k d_{j+1}[k] \left\langle \frac{1}{\sqrt{2}}\psi(\frac{x'}{2}), \bar{\phi}(x'-n+2k) \right\rangle . && (6.34)
\end{aligned}$$

Plugging 6.18 and 6.24 into Equation 6.34 we have

$$a_j[n] = \sum_k a_{j+1}[k]h[n-2k] + \sum_k d_{j+1}[k]g[n-2k]. \qquad (6.35)$$

Whereas Equations 6.30 and 6.31 analyze by *correlation* with a filter followed by downsampling, Equation 6.35 synthesizes by summing two vectors, each of which result from upsampled filters *convolved* with approximation and detail coefficients, respectively. We have arrived at what is known as a **filter bank** (Strang and Nguyen, 1997): a mathematical construction that is capable of breaking a discrete signal into two separate downsampled signals via Equations 6.30 and 6.31 and then reconstructing the original with Equation 6.35.

It can be shown (Mallat, 1998) that a filter bank is an $O(N)$ process when the filters are short length FIR. If the filters are long, or IIR, the it can still be operated in the Fourier domain with $O(N \log N)$ operations.

The recursive filters h, $\bar{h}$, g, and $\bar{g}$ are the key to filter banks and in order to guarantee an analysis and synthesis without loss of information we must pose some constraints on these filters.

6.3.1 Relationship between filters

In order to understand the relationship between the various filters in a filter bank, we first observe that downsampling a function $f(x)$ in the Fourier domain has the following representation (Strang and Nguyen, 1997, chapter 3):

$$\hat{f}_{\downarrow 2}(\omega) = \frac{1}{2}[\hat{f}(\frac{\omega}{2}) + \hat{f}(\frac{\omega}{2} + \pi)], \tag{6.36}$$

where $\hat{f}_{\downarrow 2}(\omega)$ signifies the downsampled function. Transforming Equations 6.30 and 6.31 into the Fourier domain we then have, respectively,

$$\hat{a}_{j+1}(\omega) = \frac{1}{2}[\hat{a}_j(\frac{\omega}{2})\hat{\bar{h}}^*(\frac{\omega}{2}) + \hat{a}_j(\frac{\omega}{2} + \pi)\hat{\bar{h}}^*(\frac{\omega}{2} + \pi)], \tag{6.37}$$

and,

$$\hat{d}_{j+1}(\omega) = \frac{1}{2}[\hat{a}_j(\frac{\omega}{2})\hat{\bar{g}}^*(\frac{\omega}{2}) + \hat{a}_j(\frac{\omega}{2} + \pi)\hat{\bar{g}}^*(\frac{\omega}{2} + \pi)]. \tag{6.38}$$

Doubling all frequencies in these equations gives

$$\hat{a}_{j+1}(2\omega) = \frac{1}{2}[\hat{a}_j(\omega)\hat{\bar{h}}^*(\omega) + \hat{a}_j(\omega + \pi)\hat{\bar{h}}^*(\omega + \pi)], \tag{6.39}$$

and,

$$\hat{d}_{j+1}(2\omega) = \frac{1}{2}[\hat{a}_j(\omega)\hat{\bar{g}}^*(\omega) + \hat{a}_j(\omega + \pi)\hat{\bar{g}}^*(\omega + \pi)]. \tag{6.40}$$

Transforming Equation 6.35 into the Fourier domain yields

$$\hat{a}_j(\omega) = \hat{a}_{j+1}(2\omega)\hat{h}(\omega) + \hat{d}_{j+1}(2\omega)\hat{g}(\omega). \tag{6.41}$$

Substituting Equations 6.41 into 6.39, expanding, and grouping terms by $\hat{a}_{j+1}(2\omega)$ and $\hat{d}_{j+1}(2\omega)$, we arrive at

$$\begin{aligned}\hat{a}_{j+1}(2\omega) = {} & \frac{1}{2}[\hat{h}(\omega)\hat{\bar{h}}^*(\omega) + \hat{h}(\omega + \pi)\hat{\bar{h}}^*(\omega + \pi)]\hat{a}_{j+1}(2\omega) + \\ & \frac{1}{2}[\hat{g}(\omega)\hat{\bar{h}}^*(\omega) + \hat{g}(\omega + \pi)\hat{\bar{h}}^*(\omega + \pi)]\hat{d}_{j+1}(2\omega).\end{aligned} \tag{6.42}$$

To preserve equality between the left and right sides of Equation 6.42 we explicitly see two requirements for perfect reconstruction in a biorthogonal filter bank. They are

$$\hat{h}(\omega)\hat{\hat{h}}^*(\omega) + \hat{h}(\omega + \pi)\hat{\hat{h}}^*(\omega + \pi) = 2 \tag{6.43}$$

$$\hat{g}(\omega)\hat{\hat{h}}^*(\omega) + \hat{g}(\omega + \pi)\hat{\hat{h}}^*(\omega + \pi) = 0 \tag{6.44}$$

We can represent these equations by a 2 × 2 matrix acting on a vector:

$$\begin{pmatrix} \hat{h}(\omega) & \hat{h}(\omega + \pi) \\ \hat{g}(\omega) & \hat{g}(\omega + \pi) \end{pmatrix} \times \begin{pmatrix} \hat{\hat{h}}^*(\omega) \\ \hat{\hat{h}}^*(\omega + \pi) \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}. \tag{6.45}$$

In solving this system of equations we obtain

$$\begin{pmatrix} \hat{\hat{h}}^*(\omega) \\ \hat{\hat{h}}^*(\omega + \pi) \end{pmatrix} = \frac{2}{\Delta(\omega)} \begin{pmatrix} -\hat{g}(\omega + \pi) \\ \hat{g}(\omega) \end{pmatrix}, \tag{6.46}$$

where $\Delta(\omega)$ is the determinant:

$$\Delta(\omega) = \hat{g}(\omega + \pi)\hat{h}(\omega) - \hat{g}(\omega)\hat{h}(\omega + \pi). \tag{6.47}$$

We unfortunately cannot evaluate this determinant because it involves a function we do not know, namely, $\hat{g}(\omega)$. It is interesting to notice, however, that Equation 6.47 obeys the relationship

$$\Delta(\omega + \pi) = -\Delta(\omega). \tag{6.48}$$

Mallat (1998, page 261) shows that a function satisfying this constraint is

$$\Delta(\omega) = -2ae^{i(2l+1)\omega}, \tag{6.49}$$

where $a \in \mathbb{R}$ and $l \in \mathbb{Z}$. It is a convention to set $a = 1$ and $l = 0$. This equation combined with Equation 6.45 gives us our high pass filter $\hat{g}$:

$$\hat{g}(\omega) = -e^{i\omega}\hat{\bar{h}}^*(\omega + \pi). \tag{6.50}$$

To derive the relationship between $\hat{\bar{g}}$ and $\hat{h}$ we substitute Equation 6.41 into Equation 6.40 and group terms by $\hat{a}_{j+1}(2\omega)$ and $\hat{d}_{j+1}(2\omega)$:

$$\begin{aligned} \hat{d}_{j+1}(2\omega) &= \frac{1}{2}[\hat{h}(\omega)\hat{\bar{g}}^*(\omega) + \hat{h}(\omega + \pi)\hat{\bar{g}}^*(\omega + \pi)]\hat{a}_{j+1}(2\omega) + \\ & \frac{1}{2}[\hat{g}(\omega)\hat{\bar{g}}^*(\omega) + \hat{g}(\omega + \pi)\hat{\bar{g}}^*(\omega + \pi)]\hat{d}_{j+1}(2\omega). \end{aligned} \tag{6.51}$$

Preserving equality of the two sides leads to another two requirements for perfect reconstruction:

$$\hat{g}(\omega)\hat{\bar{g}}^*(\omega) + \hat{g}(\omega + \pi)\hat{\bar{g}}^*(\omega + \pi) = 2 \tag{6.52}$$

and,

$$\hat{h}(\omega)\hat{\bar{g}}^*(\omega) + \hat{h}(\omega + \pi)\hat{\bar{g}}^*(\omega + \pi) = 0. \tag{6.53}$$

We can again construct a matrix expressing these two equations:

$$\begin{pmatrix} \hat{g}(\omega) & \hat{g}(\omega + \pi) \\ \hat{h}(\omega) & \hat{h}(\omega + \pi) \end{pmatrix} \times \begin{pmatrix} \hat{\bar{g}}^*(\omega) \\ \hat{\bar{g}}^*(\omega + \pi) \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}. \tag{6.54}$$

Inverting this matrix leads to

$$\begin{pmatrix} \hat{\bar{g}}^*(\omega) \\ \hat{\bar{g}}^*(\omega + \pi) \end{pmatrix} = \frac{2}{-\Delta(\omega)} \begin{pmatrix} \hat{h}(\omega + \pi) \\ -\hat{h}(\omega) \end{pmatrix}, \tag{6.55}$$

where $\Delta(\omega)$ is defined in Equation 6.49. Combining Equation 6.55 with Equation

6.49 gives us

$$\hat{\bar{g}}(\omega) = -e^{i\omega}\hat{h}^*(\omega + \pi). \tag{6.56}$$

Equations 6.50 and 6.56 establish a link between the high-pass filters, $\hat{g}$, $\hat{\bar{g}}$, and the low-pass filters, $\hat{h}$ and $\hat{\bar{h}}$. We typically start with only the scaling function ϕ from which we derive $\hat{h}$. We then need to establish a relationship between $\hat{h}$ and either $\hat{\bar{h}}$ or $\hat{g}$ in order completely specify all four filters. We show the connection between the four filters in the schematic of Figure 6-2.

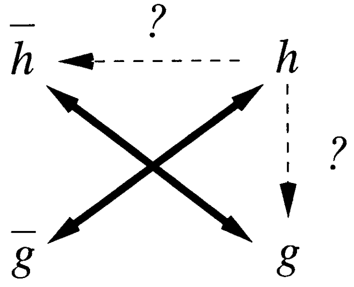

Figure 6-2: Relationship between biorthogonal filters.

Before we finish linking the filters, we mention that workers often start with a discrete FIR, $\hat{h}$, and from it construct a filter bank. Going in this direction is more difficult because one must show that the filter defines a valid scaling function. Proving this involves analyzing the convergence of

$$\hat{\phi}(\omega) = \prod_{p=1}^{\infty} \frac{\hat{h}(2^{-p}\omega)}{\sqrt{2}}. \tag{6.57}$$

This product converges to a valid scaling function when following properties hold true (Mallat, 1998, page 229):

1. $|\hat{h}(\omega)|^2 + |\hat{h}(\omega + \pi)|^2 = 2 \quad \forall \omega \in \mathbb{R}$.

2. $\hat{h}(0) = \sqrt{2}$.

3. $\hat{h}(\omega)$ is 2π periodic and continuously differentiable in a neighborhood of $\omega = 0$.

4. $\inf_{\omega \in [-\frac{\pi}{2}, \frac{\pi}{2}]} |\hat{\omega}| > 0$.

Starting with a valid scaling function in the first place, and from it deriving the filter h alleviates this difficulty.

Biorthogonal filter banks

It is common practice to apply the method of Smith and Barnwell (1986) to derive the high-pass filter $\hat{g}$ directly from $\hat{h}$

$$\hat{g}(\omega) = -e^{-i\omega}\hat{h}^*(\omega + \pi). \tag{6.58}$$

From $\hat{g}$ and $\hat{h}$ the other filters can be derived through Equations 6.50 and 6.56. The filters and the related scaling functions are indeed biorthogonal but, in general, they span different multiresolution spaces.

Dual filter banks

In order to span the same multiresolution, one starts with a valid Riesz basis, ϕ, which specifies a unique dual function, $\bar{\phi}$, that spans the same space. These functions are automatically biorthogonal to each other, and if one is a valid scaling function, so is the other. From these functions valid low-pass filters, $\hat{h}$ and $\hat{\bar{h}}$, can be derived. One can then calculate the related high-pass filters for both scaling functions from Equations 6.50 and 6.56. This gives what we call **dual filter bank**: a biorthogonal filter bank with the added property of having synthesis and analysis scaling functions that span the same multiresolution.

Whichever way we derive a valid filter bank, the g's finally give us the wavelet via Equation 6.26. In the Fourier domain, for a synthesis wavelet at $j = 1$, this is

$$\begin{aligned}\hat{\psi}(2\omega) &= \frac{1}{\sqrt{2}}\hat{g}(\omega)\hat{\phi}(\omega) \\ &= \frac{1}{\sqrt{2}}\hat{\bar{h}}^*(\omega + \pi)\hat{\phi}(\omega)\end{aligned} \tag{6.59}$$

6.4 Fractional spline wavelets

In chapter 5 we defined fractional spline basis functions. These functions are a Riesz basis for $\mathcal{L}_2$ and fulfill all the requirements of a scaling function. We therefore use them to generate wavelets. These **fractional spline wavelets** inherit all the beneficial properties of the spline basis functions, including an adjustable parameter α exactly fitted to the Hölder continuity of the functions they analyze or synthesize.

Spline wavelets are not new to the wavelet community and are known by the names of their creators: Battle (1987) and Lemarié (1988). What is new is the generalization of these wavelets to non-integer degree splines.

To construct fractional spline wavelets we must first construct the refinement filter $h[k]$, as described in Section 6.1. To distinguish between refinement filters generated by the different spline basis functions we put two subscripts: 1) $+, -,$ or $\star$ signifying causal, anti-causal, or symmetric basis functions, and, 2) $\beta, \eta,$ or γ signifying whether the spline is a B-spline, cardinal spline, or orthogonal spline, respectively. The subscripts will be dropped when they are irrelevant or implied by the text.

The (synthesis) refinement filter associated with B-spline basis functions is implied by

$$\frac{1}{\sqrt{2}}\beta^{\alpha}(x/2) = \sum_k h^{\alpha}_{\beta}[k]\beta^{\alpha}(x-k) \tag{6.60}$$

or, in the Fourier domain,

$$\hat{\beta}^{\alpha}(2\omega) = \frac{1}{\sqrt{2}}\hat{h}^{\alpha}_{\beta}(\omega)\hat{\beta}^{\alpha}(\omega). \tag{6.61}$$

Rearranging, we have

$$\hat{h}^{\alpha}_{\beta}(\omega) = \sqrt{2}\frac{\hat{\beta}^{\alpha}(2\omega)}{\hat{\beta}^{\alpha}(\omega)}. \tag{6.62}$$

By combining Equations 5.43 and 6.62 we obtain the following expression for the

B-spline refinement filter in the causal, anti-causal, and symmetric cases, respectively:

$$\hat{h}^{\alpha}_{+,\beta}(\omega) = \sqrt{2}\left(\frac{1+e^{-i\omega}}{2}\right)^{\alpha+1}, \tag{6.63}$$

$$\hat{h}^{\alpha}_{-,\beta}(\omega) = \sqrt{2}\left(\left[\frac{1+e^{-i\omega}}{2}\right]^{*}\right)^{\alpha+1}, \tag{6.64}$$

$$\hat{h}^{\alpha}_{\star,\beta}(\omega) = \sqrt{2}\left|\frac{1+e^{-i\omega}}{2}\right|^{\alpha+1}. \tag{6.65}$$

Similarly, if we are dealing with cardinal splines, the refinement filter in the Fourier domain is

$$\hat{h}^{\alpha}_{\eta}(\omega) = \sqrt{2}\frac{\hat{\eta}^{\alpha}(2\omega)}{\hat{\eta}^{\alpha}(\omega)}. \tag{6.66}$$

Combining Equations 5.46 and 6.62 with 6.66 we have the following expressions for cardinal spline refinement filters:

$$\hat{h}^{\alpha}_{\eta}(\omega) = \hat{h}^{\alpha}_{\beta}(\omega)\frac{\hat{m}^{\alpha}(\omega)}{\hat{m}^{\alpha}(2\omega)}, \tag{6.67}$$

where $\hat{m}^{\alpha}(\omega) = \sum_k \hat{\beta}^{\alpha}(\omega + 2\pi k)$ is the sampled B-spline in the Fourier domain.

Finally, in the case of orthogonal spline basis functions we have the refinement filter

$$\hat{h}^{\alpha}_{\gamma}(\omega) = \sqrt{2}\frac{\hat{\gamma}^{\alpha}(2\omega)}{\hat{\gamma}^{\alpha}(\omega)}. \tag{6.68}$$

By combining Equations 5.48, 6.62, and 6.68, we have the following formula for the orthogonal spline refinement filter:

$$\hat{h}^{\alpha}_{\gamma}(\omega) = \hat{h}^{\alpha}_{\beta}(\omega)\sqrt{\frac{\hat{a}^{2\alpha+1}(\omega)}{\hat{a}^{2\alpha+1}(2\omega)}}, \tag{6.69}$$

where $\hat{a}^{2\alpha+1}(\omega) = \sum_k |\hat{\beta}^\alpha(\omega + 2\pi k)|^2$ is the 2π periodic power spectrum in the Fourier domain.

The dual low-pass filters can be derived from the dual spline basis functions from Section 5.2.5. For B-splines and cardinal splines they are as follows:

$$\hat{\bar{h}}^\alpha_\beta(\omega) = \hat{h}^\alpha_\beta \frac{\hat{a}^{2\alpha+1}(\omega)}{\hat{a}^{2\alpha+1}(2\omega)}, \tag{6.70}$$

and

$$\hat{\bar{h}}^\alpha_\eta(\omega) = \hat{h}^\alpha_\beta \frac{\hat{m}^\alpha(2\omega)}{\hat{m}^\alpha(\omega)} \frac{\hat{a}^{2\alpha+1}(\omega)}{\hat{a}^{2\alpha+1}(2\omega)}. \tag{6.71}$$

Having both low-pass filters, we can complete the filter bank by deriving g and $\bar{g}$ from Equations 6.50 and 6.56.

The fractional spline wavelets derived above are called **semi-orthogonal wavelets**. They were derived from dual functions that span a Riesz basis. The multiresolution spaces for both scaling functions are the same. Thus $\mathcal{V}_j = \bar{\mathcal{V}}_j$ and $\mathcal{W}_j = \bar{\mathcal{W}}_j$. From this it follows that the wavelets are orthogonal to all wavelets at other scales, but not within the same scale (Strang and Nguyen, 1997, page 256).

In Figures 6-3 and 6-4 we show examples of causal and symmetric fractional B-spline wavelets, respectively. In Figure 6-5 we show examples of symmetric fractional cardinal spline wavelets. And, in Figure 6-6, we show symmetric fractional orthogonal spline wavelets. All wavelets are shown for α's ranging from 0 to 3 and in steps of $\alpha = 0.25$ except for the cardinal splines wavelets, which were plotted for $\alpha = 1$ to $\alpha = 3$.

Wavelets can be generalized to multiple dimensions by tensor outer products (Nowak and Baraniuk, 1998). We show a few examples of 2-D wavelets in Figures 6-7 through 6-10.

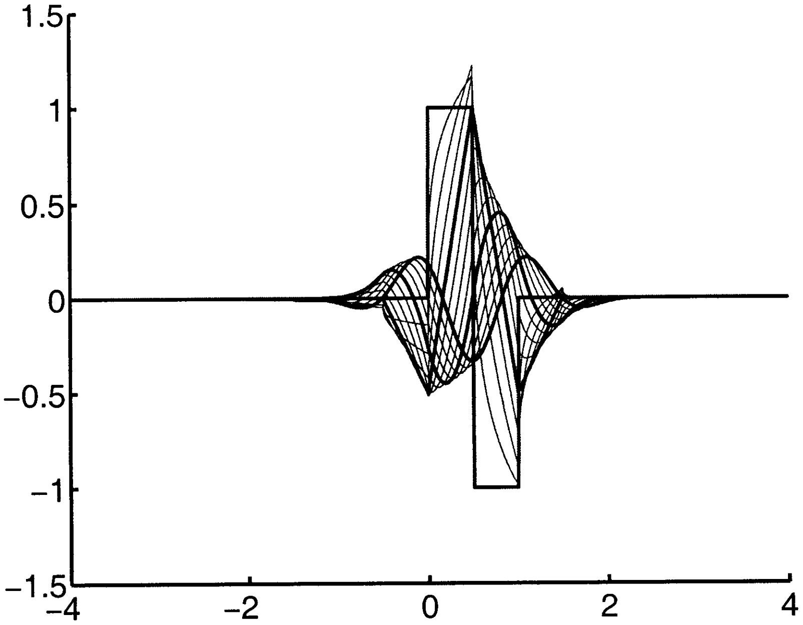

Figure 6-3: Causal B-spline wavelets for $\alpha = 0$ to $\alpha = 3$ in steps of 0.25.

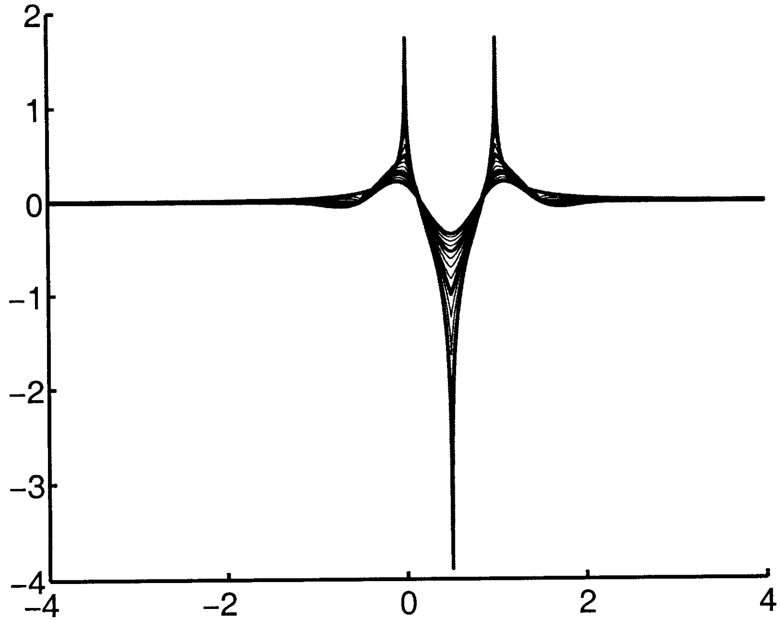

Figure 6-4: Symmetric B-spline wavelets for $\alpha = 0$ to $\alpha = 3$ in steps of 0.25.

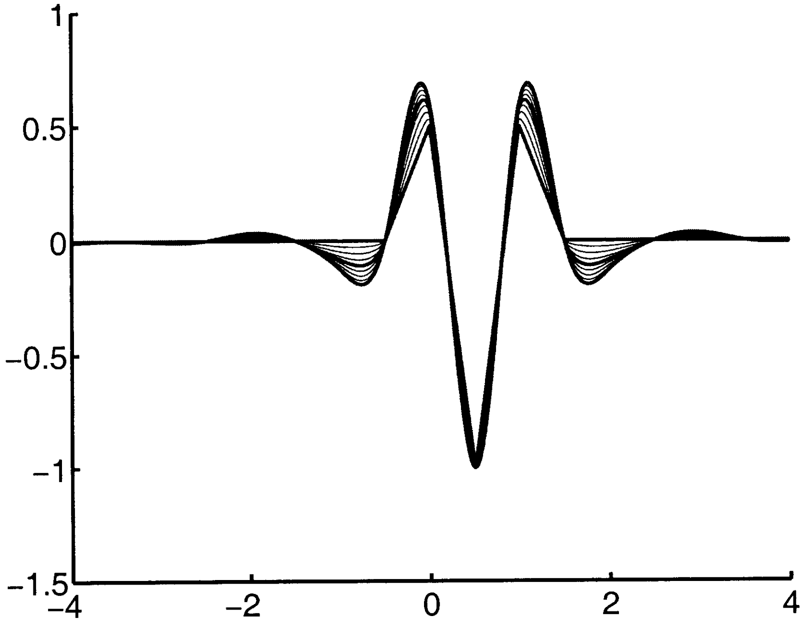

Figure 6-5: Symmetric cardinal spline wavelets for $\alpha = 1$ to $\alpha = 3$ in steps of 0.25.

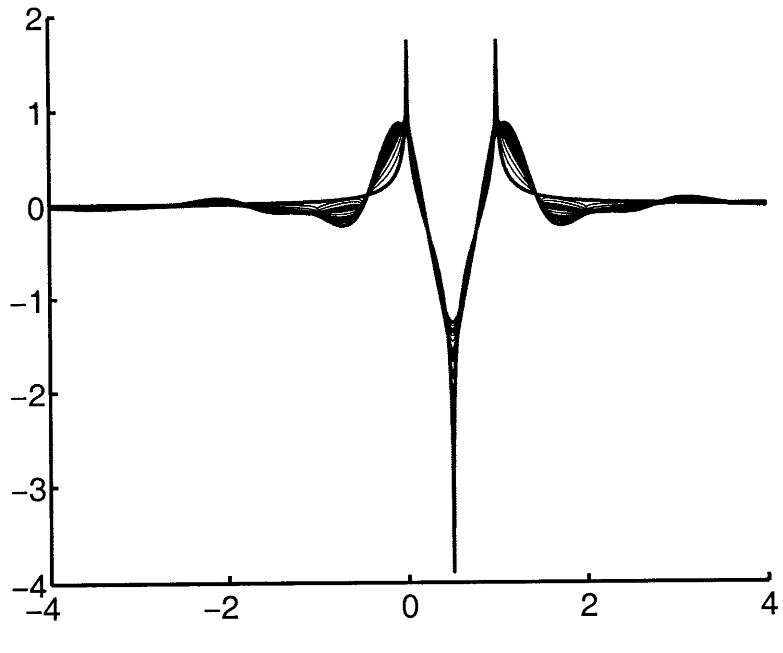

Figure 6-6: Symmetric orthogonal spline wavelets for $\alpha = 0$ to $\alpha = 3$ in steps of 0.25.

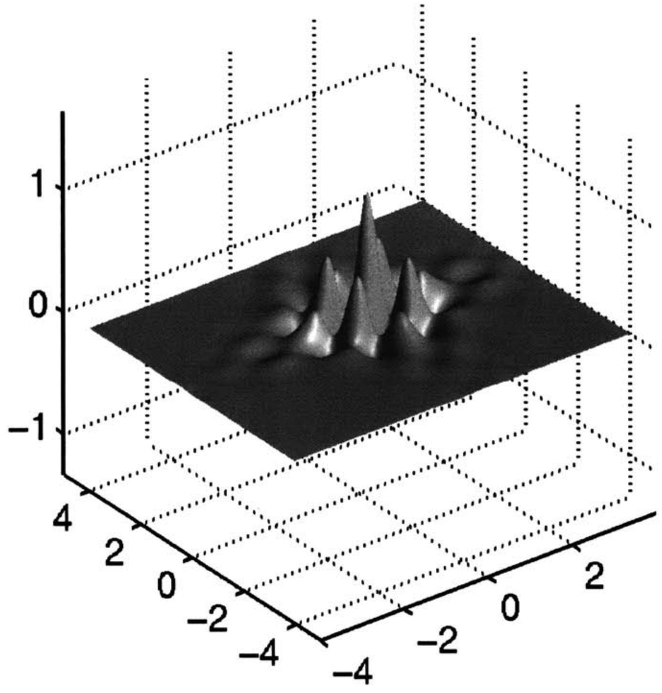

Figure 6-7: 2-D Symmetric cubic cardinal spline wavelet outer-product basis function.

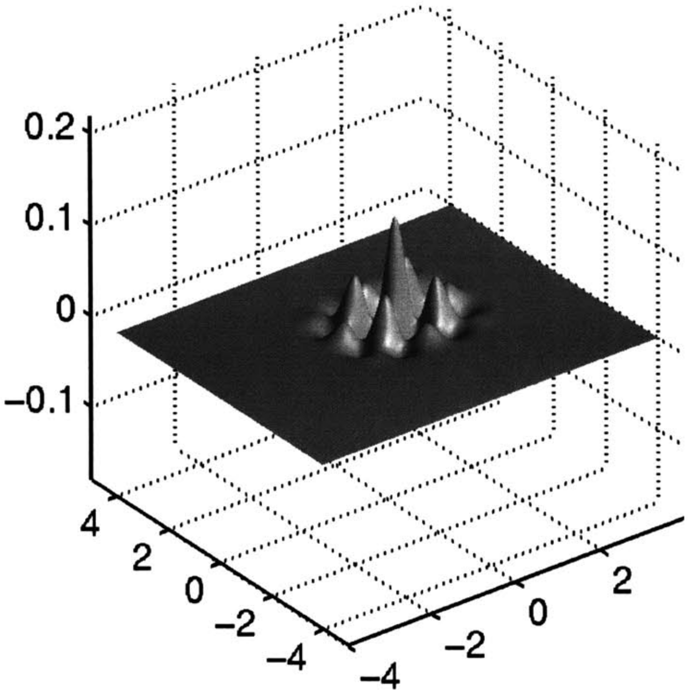

Figure 6-8: 2-D Symmetric cubic B-spline wavelet outer-product basis function.

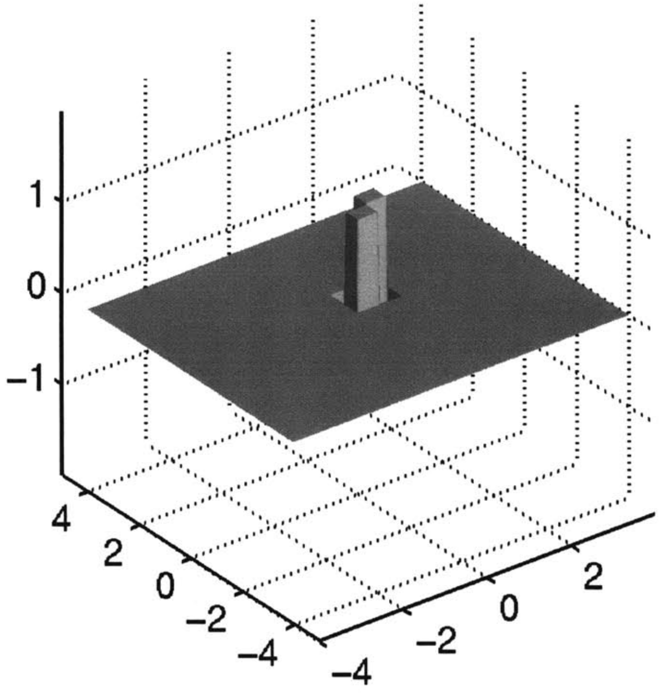

Figure 6-9: 2-D Causal 0^{th} degree orthogonal spline wavelet outer-product basis function.

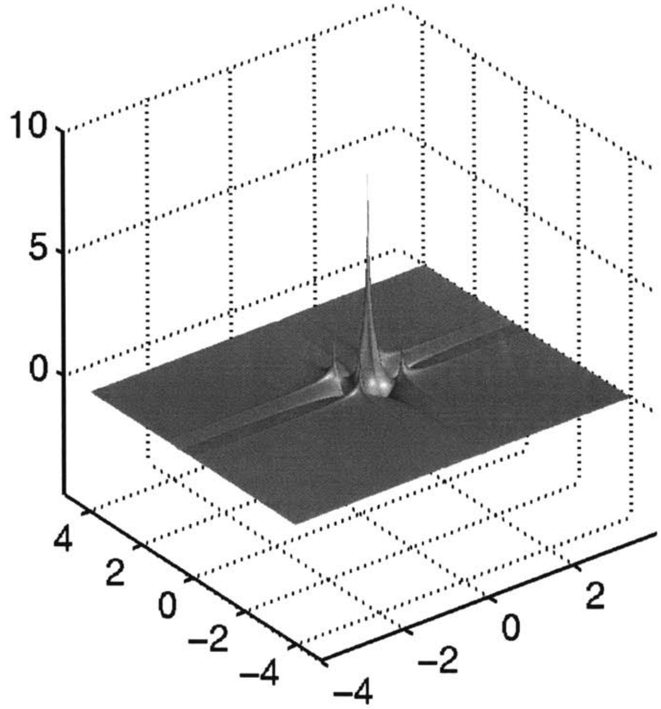

Figure 6-10: 2-D Symmetric 0^{th} degree orthogonal spline wavelet outer-product basis function.

Chapter 7

Wavelet Domain Geophysical Inversion[‡]

In this chapter we examine the usefulness of wavelets for solving inverse problems. Building upon the theory of Chapters 5 and 6, we construct fractional spline wavelets and project an inverse problem into the wavelet domain via a filter bank. A new method of wavelet coefficient thresholding is used to regularize the solution, rather than classical methods. We compare the wavelet based method to classical methods for both synthetic and real geophysical inverse problems.

Introduction

Wavelets have established themselves as highly useful mathematical functions for a variety of applications. In harmonic analysis they allow for the representation of certain functions for which classical basis functions are either inefficient, or completely incapable (Meyer, 1992; Donoho, 1995). In applied mathematics they have proven superb for data compression and denoising of audio and video signals (DeVore et al., 1992; Strang and Nguyen, 1997; Mallat, 1998).

Denoising an observed signal can be looked upon as an inverse problem, where

[‡] This chapter to be submitted to *Geophysics*.

the forward modeling operator is simply the identity operator on the model space. Given the success of wavelets in this application, it was reasonable to ask whether they were applicable to other inverse problems. In the 1990's a large amount of work in the applied mathematics and statistics community went into answering this question. Perhaps foremost among the workers was Donoho (1995), who formulated the wavelet-vaguelette decomposition (WVD) for solving certain linear inverse problems. His theory was then applied by various workers; Kolaczyk (1994, 1996) applied the method to 2-D tomographic problems; Abramovich and Silverman (1998), formulated a related vaguelette-wavelet method; Nowak and Thul (1998); Neelami et al. (1999, a) and Neelami et al. (b) applied it to deconvolution problems.

The WVD based inversion method is innovative in its use of a nonlinear thresholding operation in the wavelet domain. Thresholding is both fast and easy, and performs a non-stationary filtering of the estimated model. This effectively smooths the estimate more in areas of little signal and smooths less in more interesting areas, thereby preserving interesting features.

To our knowledge WVD methods have not appeared in the geophysical literature. Therefore the first goal of the work presented here is to apply the theory to a 1-D geophysical problem. The second goal is to combine the wavelet-vaguelette decomposition method with **fractional splines wavelets**, a versatile class of wavelets recently introduced by Unser and Blu (1999, 2000). To our knowledge this work also is the first use of the WVD framework with fractional spline wavelets.

The outline is as follows: we first introduce fractional splines, and from them, fractional spline wavelets. Using these functions we can discretize functions and operators. This allows us to pose a discrete linear inverse problem as an approximation to the analogous continuous problem, and then translate such a problem to the wavelet domain via a filter bank. We then present classical regularized least-squares inverse methods in the wavelet domain, as well as the alternate WVD framework and discuss pros and cons of the methods. After this we apply the WVD method to the 1-D problem of slowness estimation from picked traveltime data. Results are compared to both damped and regularized least-squares methods. We then apply all the methods to a real traveltime data set, discuss the results, and present conclusions.

7.1 Fractional splines and wavelets

Unser and Blu (1999, 2000) have presented α degree **fractional splines**: α-Hölder continuous functions (Mallat, 1998) used to approximate more complicated "real" functions. The adjustable α parameter controls the degree of differentiability of the approximation function. Fractional splines are a generalization of the classical integer degree splines to fractional degree of continuity. There are several convenient basis functions for fractional splines. Perhaps the most famous are the causal B-splines which have the Fourier domain representation

$$\hat{\beta}_\alpha(\omega) = \left(\frac{1 - e^{-i\omega}}{i\omega}\right)^\alpha \tag{7.1}$$

We show examples of these basis functions for varying degree α in Figure 7-1. Integer shifts of basis functions such as these are the building blocks of a spline. These basis functions can be made both symmetric about the origin and orthogonal to integer

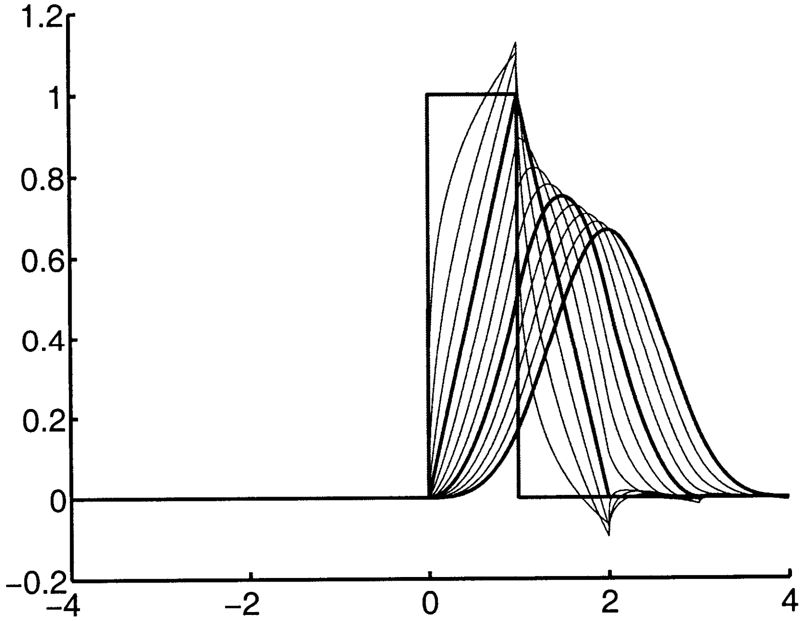

Figure 7-1: Causal fractional B-splines of degrees $\alpha = 0$ to $\alpha = 3$ in steps of 0.25.

shifts of themselves via the following equation (Mallat, 1998, p. 225):

$$\hat{w}_\alpha(\omega) = \frac{\left|\hat{\beta}_\alpha(\omega)\right|}{\left(\sum_k \left|\hat{\beta}_\alpha(\omega + 2\pi k)\right|^2\right)^{1/2}} \tag{7.2}$$

These **orthogonal spline basis functions** are the spline basis functions we will use in this work. As an example, we show in Figure 7-2 the $\alpha = 1.25$ symmetric orthogonal spline basis function derived from the $\alpha = 1.25$ B-spline via Equation 7.2.

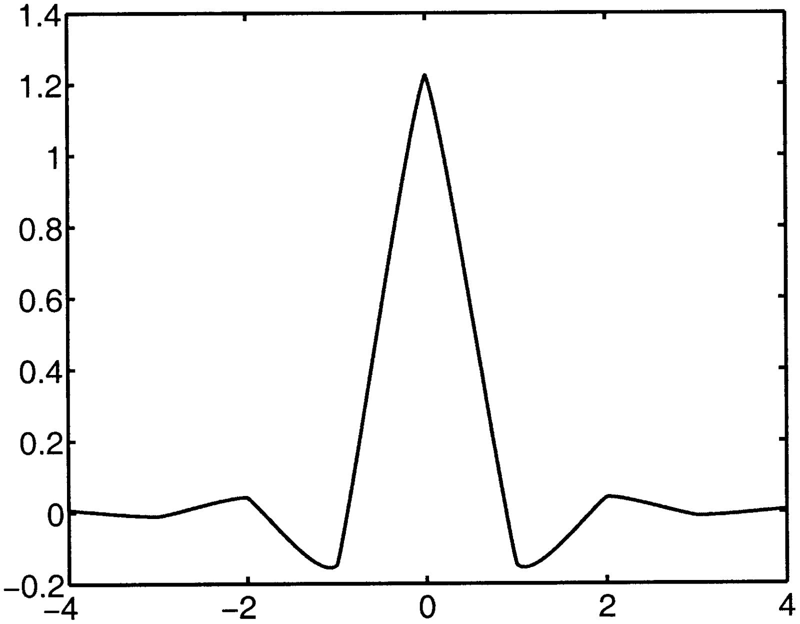

Figure 7-2: $\alpha = 1.25$ orthogonal spline basis function.

Specifying a basis of functions determines the values of coefficients when discretizing continuous functions into vectors and linear operators into matrices. Ignoring the details of such an operation will unnecessarily generate errors. Fractional spline basis functions are used for 3 reasons:

1. Given only prior information about the differentiability of a true function, one can use an equally differentiable fractional spline basis to optimally approximate that function.

2. Spline basis functions have better approximation properties than any other known function (Strang and Nguyen, 1997, p. 223). Fractional spline basis functions generalize this property to a fractional order of approximation (Unser and Blu, 1999).

3. Spline basis functions possess all the multiresolution properties necessary for creating wavelets (Blu and Unser, 2001b).

In particular, orthogonal spline basis functions can be used to create orthogonal spline wavelets (Blu and Unser, 2001b). Using the Fourier domain formula of Mallat (1998),

$$\hat{\psi}(2\omega) = \frac{1}{\sqrt{2}}\hat{g}(\omega)\hat{w}(\omega), \tag{7.3}$$

we can convert our basis functions into wavelets. $\hat{g}$ is a high pass filter determined uniquely from a lowpass filter $\hat{h}$, which, in turn, is uniquely determined by the basis function $\hat{w}$ (Mallat, 1998). In Figure 7-3 we show the $\alpha = 1.25$ orthogonal spline wavelet obtained from the basis function of Figure 7-2.

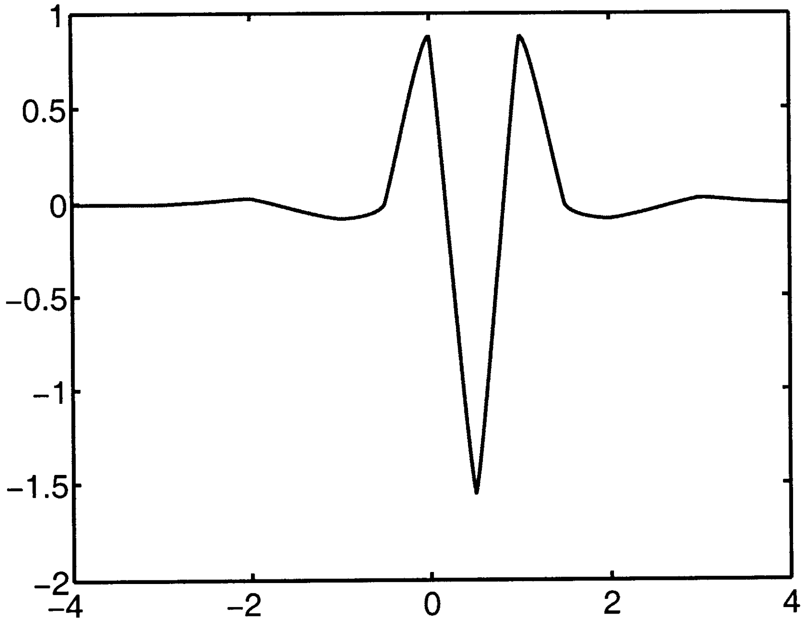

Figure 7-3: $\alpha = 1.25$ orthogonal spline wavelet.

The wavelet transform (Mallat, 1998) uses shifts and dilations of such a wavelet to project and represent a function in a basis of wavelets. It is shown in Chapter 6 that if the function has already been projected into a certain spline basis, and the wavelets are constructed from the dual functions of that same spline basis, the wavelet transform can be exactly implemented via a discrete filter bank operating on the spline coefficients. The filter bank (Strang and Nguyen, 1997) is how the transform will be computed in this chapter, and can be implemented in either $O(N)$ or $O(N \log N)$ operations, depending on the wavelet. We, however, will symbolically represent the transform as a discrete matrix, $\mathbf{W}$, which is an equivalent, although computationally inefficient method of applying it. Each row of $\mathbf{W}$ is coefficients of the projection of a shifted and/or dilated wavelet into the appropriate spline basis. Representing the wavelet transform as a matrix will allow us to analyze the use of wavelets in linear inverse problems more efficiently.

The superiority of wavelets over more traditional basis functions stems from the fact that they represent functions in both spatial and scale coordinates. This has led to their usefulness in signal compression and harmonic analysis, and it will also lead to their use in inversion in this chapter.

7.2 Classical inversion in the wavelet domain

Linear inverse problems encountered in science and mathematics often take the form

$$z = \mathrm{K}f + n, \tag{7.4}$$

where f is a function we desire to estimate, n is the uncertainty (noise) in the system (which is independent of f), K is a linear operator mapping models into data, and z is the noise contaminated data. In order solve such a problem on a computer we must discretize the functions and operator by projecting them onto a finite dimensional basis of functions. We project into the basis of orthogonal spline functions of degree $\alpha = 1.25$ shown in Figure 7-2. We explain why we use this particular value of α below. After projecting into this basis Equation 7.4 can be replaced by the related finite dimension system:

$$\mathbf{z} = \mathbf{Kf} + \mathbf{n}, \tag{7.5}$$

where $\mathbf{f}$, $\mathbf{n}$, and $\mathbf{z}$ are finite dimensional vectors and $\mathbf{K}$ is a matrix.

Upon observation of a vector $\mathbf{z}_{obs}$, and assuming that $\mathbf{n}$ is a multi-Gaussian random vector, finding the the maximum likelihood solution reduces to finding the $\mathbf{f}$ that solves

$$\min_{\mathbf{f}} \left\| \mathbf{C}_n^{-1/2} \left(\mathbf{z}_{obs} - \mathbf{Kf} \right) \right\|_{\ell_2}, \tag{7.6}$$

where $\mathbf{C}_n^{-1/2}$ is the square root of the inverse of the covariance matrix of $\mathbf{n}$.

The minimization of Equation 7.6 leads to the **least-squares** solution:

$$\hat{\mathbf{f}} = \mathbf{K}_{ls}^{-1}\mathbf{z} \tag{7.7}$$

where $\mathbf{K}_{ls}^{-1} = (\mathbf{K}^T\mathbf{C}_\mathbf{n}^{-1}\mathbf{K})^{-1}\mathbf{K}^T\mathbf{C}_\mathbf{n}^{-1}$. If $\mathbf{K}$ is square and invertible then $\mathbf{K}_{ls}^{-1} = \mathbf{K}^{-1}$.

If we substitute Equation 7.5 into Equation 7.7 we obtain

$$\hat{\mathbf{f}} = \mathbf{f} + \mathbf{K}_{ls}^{-1}\mathbf{n}. \tag{7.8}$$

From Equation 7.8 we can see that our solution will be contaminated with colored noise, $\mathbf{K}_{ls}^{-1}\mathbf{n}$. The effect of this noise can be seen through the inequality

$$||\hat{\mathbf{f}} - \mathbf{f}||_{\ell_2} \leq ||\mathbf{K}_{ls}^{-1}||_{\ell_2} \; ||\mathbf{n}||_{\ell_2} \tag{7.9}$$

If the noise, $\mathbf{n}$, is of too large a magnitude, or if $\mathbf{K}$ or $\mathbf{K}^T\mathbf{C}_\mathbf{n}^{-1}\mathbf{K}$ was not invertible in the first place, we can see that the right side of the equation will be large or infinite, and we have an **ill-posed** inverse problem (Tikhonov and Arsenin, 1977). The traditional way to solve such a problem is by putting some other constraint on the system. In Bayesian methods (Tarantola, 1987) the constraint involves simultaneously maximizing a prior probability density function (PDF) defined on the model to be estimated. A correctly chosen prior PDF will both damp the effects of the noise and provide a unique solution to the problem. The solution is then known as the maximum a-posteriori (MAP) solution. In a non-Bayesian setting (Tikhonov and Arsenin, 1977) the constraint usually comes in the form of minimizing a linear operator acting on the vector to be estimated, which is functionally equivalent to the Bayesian method with a multi-Gaussian prior. This framework is known as regularized maximum likelihood. Either way, the constraint is expressed as

$$\min_{\mathbf{f}} ||\mathbf{L}\mathbf{f}||_{\ell_2}, \tag{7.10}$$

where $\mathbf{L}$ is usually a differencing matrix obtained by projecting a differential operator into the chosen spline basis (Chapter 5). This leads to the regularized least squares solution

$$\hat{\mathbf{f}} = \mathbf{K}_{rls}^{-1}\mathbf{z}, \tag{7.11}$$

where

$$\mathbf{K}_{rls}^{-1} = \left(\mathbf{K}^T\mathbf{C}_{\mathbf{n}}^{-1}\mathbf{K} + \zeta\ \mathbf{C}_{\mathbf{f}}^{-1}\right)^{-1}\mathbf{K}^T\mathbf{C}_{\mathbf{n}}^{-1}. \tag{7.12}$$

In the Bayesian framework $\mathbf{C}_{\mathbf{f}} = (\mathbf{L}^T\mathbf{L})^{-1}$ is the covariance matrix of $\mathbf{f}$. ζ is an adjustable constant that changes the amount of damping in the inversion. In the Bayesian case it is the inverse of the prior variance. An optimal ζ is hard to estimate in advance and a poor choice of it can lead to poor results in the inversion. If it is too large, the solution will be too smooth. If it is too small the solution will be too rough. In the special case where $\mathbf{L} = \mathbf{I}$ we replace $\mathbf{C}_{\mathbf{f}}$ with $\mathbf{I}$ and have the **damped-least squares** solution, which essentially inputs less prior information into the inverse problem.

If one can decompose the matrix $\mathbf{K}$ into a sequence of other matrices that are more computationally efficient to apply, the least-squares inversion can also be made more efficient. For instance, if $\mathbf{K}$ is a square circular convolution matrix then it is diagonalized by the unitary Fourier transform, $\mathbf{F}$. Denoting the inverse of $\mathbf{F}$ as $\mathbf{F}^T$, we have

$$\mathbf{K} = \mathbf{F}^T \Lambda_{\mathrm{K}} \mathbf{F}. \tag{7.13}$$

The matrix Λ_{K} is diagonal with the spectrum of $\mathbf{K}$ on its diagonal. The matrices $\mathbf{F}$ and $\mathbf{F}^T$ can each be operated in $O(N \log N)$ operations with the fast Fourier transform, where N is the length of the vector being operated on. Plugging Equation 7.13 into Equation 7.7 we have

$$\hat{\mathbf{f}} = \mathbf{F}^T \Lambda_{\mathrm{K}}^{-1} \mathbf{F}\mathbf{z}. \tag{7.14}$$

We can see from this that inversion becomes an $O(N \log N)$ operation. If $\mathbf{C_n}$ and $\mathbf{C_f}$ are also square circular convolution matrices we have, respectively,

$$\begin{aligned} \mathbf{C_n} &= \mathbf{F}^T \Lambda_{\mathrm{n}} \mathbf{F} && (7.15) \\ &\ \text{and} && (7.16) \\ \mathbf{C_f} &= \mathbf{F}^T \Lambda_{\mathrm{f}} \mathbf{F}. && (7.17) \end{aligned}$$

These relationships lead us to the Fourier domain regularized least-squares inverse filter

$$\hat{\mathbf{f}} = \mathbf{F}^T \Lambda_{rls}^{-1} \mathbf{F}\mathbf{z}, \tag{7.18}$$

where

$$\Lambda_{rls}^{-1} = \left(\Lambda_{\mathrm{K}}^{*} \Lambda_{\mathrm{n}}^{-1} \Lambda_{\mathrm{K}} + \zeta \Lambda_{\mathrm{f}}^{-1}\right)^{-1} \Lambda_{\mathrm{K}}^{*} \Lambda_{\mathrm{n}}^{-1}. \tag{7.19}$$

Λ_{K}^{*} is the Hermitian transpose of Λ_{K}. When $\mathbf{C_f} = \mathbf{I}$ we have the $\Lambda_{\mathrm{f}} = \mathbf{I}$ and Equation 7.19 becomes the damped least-squares inverse filter. The implementation of the regularized filter in Equation 7.18 again requires $O(N \log N)$ operations.

As fast as the fast Fourier transform is, the wavelet transform is potentially faster. For wavelets of compact support $\mathbf{W}$ can be operated in $O(N)$ operations via Mallat's pyramid algorithm (Mallat, 1998). Therefore, to make inversion in the wavelet domain appealing to us, we would like to see a formulation similar to Equations 7.13 through 7.19, only with $\mathbf{W}$ in place of $\mathbf{F}$. To derive an equivalent formulation we also require that the wavelet transform be orthogonal. Only Daubechies wavelets (Daubechies, 1988) simultaneously satisfy these requirements. Orthogonal fractional spline wavelets do not because, in general, they are not compactly supported. Assuming for a moment that we choose to use Daubechies wavelets and that $\mathbf{K}$ is square and invertible, we decompose $\mathbf{K}$ in the wavelet domain to give

$$\mathbf{K} = \mathbf{W}^T \mathbf{\Omega}_{\mathrm{K}} \mathbf{W}. \tag{7.20}$$

Substituting 7.20 into Equations 7.7 and 7.11 we have

$$\hat{\mathbf{f}} \;=\; \mathbf{W}^T \mathbf{\Omega}_{\mathrm{K}}^{-1} \mathbf{W}\mathbf{z}. \tag{7.21}$$

and

$$\hat{\mathbf{f}} \;=\; \mathbf{W}^T \mathbf{\Omega}_{rls}^{-1} \mathbf{W}\mathbf{z}, \tag{7.22}$$

where

$$\mathbf{\Omega}_{rls}^{-1} \;=\; \left(\mathbf{\Omega}_{\mathrm{K}}^T \mathbf{\Omega}_{\mathrm{n}}^{-1} \mathbf{\Omega}_{\mathrm{K}} + \zeta \mathbf{\Omega}_{\mathrm{f}}^{-1}\right)^{-1} \mathbf{\Omega}_{\mathrm{K}}^T \mathbf{\Omega}_{\mathrm{n}}^{-1}, \tag{7.23}$$

$$\mathbf{\Omega}_{\mathrm{f}} \;=\; \mathbf{W}\mathbf{C}_{\mathrm{f}}\mathbf{W}^T, \tag{7.24}$$

and

$$\mathbf{\Omega}_{\mathrm{n}} \;=\; \mathbf{W}\mathbf{C}_{\mathrm{n}}\mathbf{W}^T. \tag{7.25}$$

We show an example of a circular convolution matrix $\mathbf{K}$ and its corresponding $\mathbf{\Omega}_{\mathrm{K}}$ matrix in Figure 7-4. We use the Haar wavelet for this example, which is of both the Daubechies and fractional spline families. In order to show the dominant structure in $\mathbf{\Omega}_K$ the coefficients were thresholded at a certain value. The location of coefficients

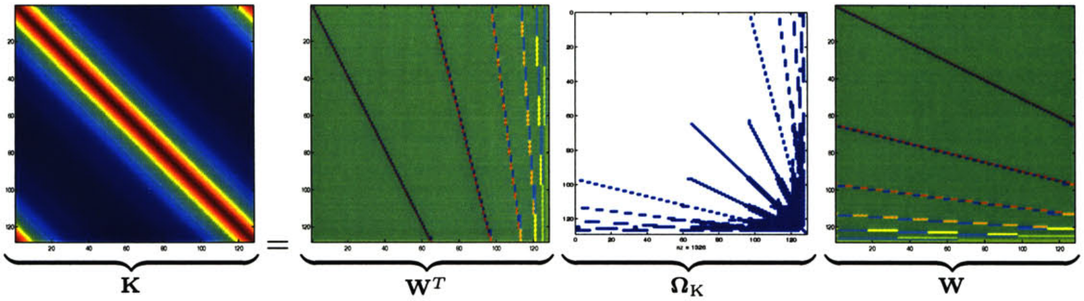

Figure 7-4: Matrix decomposition via wavelet transform matrices.

that survived the thresholding are shown in blue.

There are several problems with the formulation of Equation 7.22. First, $\boldsymbol{\Omega}_\mathrm{K}$ is not diagonal for *any* commonly encountered class of operators. There exists no known computationally efficient algorithm for applying this matrix, or inverting it.[1] Second, in general, it will not even be sparse. Rather, it will be a full matrix and its coefficients may have to be individually calculated. Thus any computational gains made with the $\mathbf{W}$ matrix are lost with $\boldsymbol{\Omega}_K$.

Li et al. (1996) solve Equation 7.22 for the same 1-D geophysical problem addressed in this work. Their synthetic problem is small enough to allow direct creation and inversion of the matrices in Equation 7.22. They, however, approximate $\boldsymbol{\Omega}_\mathbf{f}$ with a diagonal matrix that correctly models the variance of wavelet coefficients of a fractal process at each scale. Their solution is actually obtainable without the use of wavelets. Using Equation 7.11 would even allow the correct prior covariance matrix of a fractal be used, rather than an approximation.

Other workers, such as Amaratunga and Castrillon-Candas (2000), use the thresholded approximation of $\boldsymbol{\Omega}_K$ along with iterative methods to invert $\boldsymbol{\Omega}_K$ in Equation 7.21. Such a method requires only the repeated operation of $\boldsymbol{\Omega}_\mathrm{K}$ to find the solution, and is not limited to small problems. While this is potentially a computationally efficient method once the coefficients of $\boldsymbol{\Omega}_\mathrm{K}$ have been calculated and stored, the calculation of these coefficients may be a much more intensive task than the inverse

[1] Beylkin (1992); Saito and Beylkin (1993); Alpert et al. (1993); Beylkin and Torrésani (1996) have created methods for efficiently applying certain operators via filter banks. Such methods may be amenable to applying the matrix but not to inverting it.

problem itself. It is also difficult to measure the error caused by thresholding the $\mathbf{\Omega}_{\mathrm{K}}$ matrix. And, as with the method of Li et al. (1996), the solution obtained will still be the classical solution. We will not investigate such methods, but rather turn to a simpler formulation that gives superior results.

7.3 The wavelet-vaguelette decomposition

Donoho (1995) presented an alternative methodology for solving a certain set of linear inverse problems in which K is a **homogeneous operator**. These operators are defined by having the following dilation invariant property:

$$\mathrm{K}\mathrm{G}_a = a^{\alpha}\mathrm{G}_{\alpha}\mathrm{K}, \tag{7.26}$$

where G_a is the *dilation* operator, i.e. $\mathrm{G}_a f(\mathbf{x}) = f(\frac{\mathbf{x}}{a})$. Operators that obey this property include fractional integration, fractional differentiation, and the radon transform.

With some minimal constraints on a wavelet, the operation of a homogeneous operator on the wavelet will produce a vaguelette:

$$\mathrm{K}\psi_{j,k} = c(j)v_{j,k}, \tag{7.27}$$

where $\psi_{j,k} = \frac{1}{\sqrt{2^j}}w(\frac{x}{2^j} - k)$ is a valid wavelet, $v_{j,k} = \frac{1}{\sqrt{2^j}}v(\frac{x}{2^j} - k)$ is a vaguelette, and $c(j)$ is a constant dependent on scale only. Vaguelettes have a precise definition according to Meyer (1992). Among their properties are localization in space, zero mean, and some degree of Hölder continuity. Donoho's main reason for using vaguelettes is that for homogeneous operators, there exist dual vaguelette functions, $u_{j,k} = \frac{1}{\sqrt{2^j}}u(\frac{x}{2^j} - k)$, that are *approximately* orthogonal to the vaguelettes. We will show why this is a useful property below. For homogeneous operators the constants $c(j)$ are also easy to (approximately) calculate.

Since we are dealing with discrete data sets, the homogeneous operator K needs to be projected into the same spline basis as the functions were. This converts the linear operator K into a matrix $\mathbf{K}$. In a matrix formulation, the WVD method proceeds as follows: an inverse orthogonal wavelet transform matrix $\mathbf{W}^T$ is constructed, where each column are the coefficients of the projection of a wavelet into the spline basis. $\mathbf{K}$ then operates on each of these individual wavelets to produce the coefficients of the continuous vaguelette in the same basis. These discrete vaguelettes occupy each column of a matrix $\mathbf{V}^T$:

$$\mathbf{K}\mathbf{W}^T = \mathbf{V}^T\mathbf{\Gamma}. \tag{7.28}$$

In Donoho (1995) the matrix $\mathbf{\Gamma}$ is a diagonal matrix with the diagonal entries equal to the $c(j)$ constants above. For the homogeneous operators listed above Donoho provided simple formulae for calculating these constants such that the columns of $\mathbf{V}^T$ would almost have unit energy (and also be almost orthogonal to each other).

Moving $\mathbf{W}^T$ to the other side of Equation 7.28, we get the **wavelet-vaguelette decomposition** (WVD):

$$\mathbf{K} = \mathbf{V}^T\mathbf{\Gamma}\mathbf{W}. \tag{7.29}$$

Equation 7.29 looks similar to the **singular value decomposition** (Strang, 1986; Trefethen and Bau, 1997; Golub and Van Loan, 1996), with the exception that $\mathbf{V}^T$ is not an orthogonal matrix (although for homogeneous operators, it is *near* orthogonal). Donoho (1995) therefore called the entries of $\mathbf{\Gamma}$ *quasi*-singular values. It is important to note here that $\mathbf{W}$ and $\mathbf{\Gamma}$ are always invertible, but $\mathbf{V}^T$ is invertible only if $\mathbf{K}$ is. If $\mathbf{V}^T$ is indeed invertible we call its inverse $\mathbf{U}$, which has as its rows the discrete biorthogonal duals of the columns of $\mathbf{V}^T$.

We are now ready to apply the WVD to the solution of a linear inverse problem. Assuming temporarily that $\mathbf{K}$ is invertible and square, we can represent its inverse via the WVD as

$$\mathbf{K}^{-1} = \mathbf{W}^{-1}\mathbf{\Gamma}^{-1}\mathbf{U}. \tag{7.30}$$

Plugging Equation 7.30 into Equation 7.7 we get

$$\hat{\mathbf{f}} = \mathbf{W}^{-1}\mathbf{\Gamma}^{-1}\mathbf{U}\mathbf{z}. \tag{7.31}$$

This is where the crux of the WVD inversion method appears. A nonlinear **thresholding operator**, Θ_T, will be inserted between the $\mathbf{\Gamma}^{-1}$ and the $\mathbf{U}$ matrices, and will remove those vaguelette coefficients $\mathbf{v} = \mathbf{U}\mathbf{z}$ that are deemed to possess much greater noise than signal. Inserting such a nonlinear thresholding operator into Equation 7.31 we obtain

$$\hat{\mathbf{f}} = \mathbf{W}^{-1}\mathbf{\Gamma}^{-1}\Theta_T[\mathbf{U}\mathbf{z}]. \tag{7.32}$$

To gain insight as to how to create such a thresholding operator we plug Equation 7.5 into Equation 7.31 obtaining

$$\hat{\mathbf{f}} = \mathbf{W}^{-1}\mathbf{\Gamma}^{-1}\mathbf{U}(\mathbf{K}\mathbf{f} + \mathbf{n}). \tag{7.33}$$

We then examine the covariance, $\mathbf{C}_{\check{\mathbf{n}}}$, of the noise among the vaguelette coefficients, $\check{\mathbf{n}} = \mathbf{U}\mathbf{n}$. Assuming for a moment that the noise in the data is white, we have

$$\begin{aligned} \mathbf{C}_{\check{\mathbf{n}}} &= E\left[\mathbf{U}\mathbf{n}\mathbf{n}^T\mathbf{U}^T\right] \qquad (7.34)\\ &= \mathbf{U}E\left[\mathbf{n}\mathbf{n}^T\right]\mathbf{U}^T \qquad (7.35)\\ &= \sigma_{\mathbf{n}}^2 \underbrace{\mathbf{U}\mathbf{U}^T}_{\approx\mathbf{I}}. \qquad (7.36) \end{aligned}$$

Due to the approximate orthogonality of $\mathbf{U}$ we may believe that the noise in the data will be uniformly distributed among the vaguelette coefficients. Following this assumption through we may make the intuitive argument that any vaguelette coefficient of a magnitude significantly larger than the standard deviation of the noise in the vaguelette coefficients will *probably* contain more signal that noise. Conversely, vaguelette coefficients with magnitude much smaller than the standard deviation of the noise are *probably* just noise. According to Donoho (1995), these assumptions are valid when the function being estimated belongs to certain function spaces that contain smooth signals with a finite number of discontinuities. Real world signals are well approximated by functions in such spaces. In such a case the vaguelette transform will tend to isolate interesting features of a signal into a few large coefficients, while spreading the noise evenly at small magnitude over all coefficients. We can now define the thresholding estimator.

We define Θ_T by

$$\Theta_T[\cdot] = \begin{cases} \mathbf{v}_k & : \quad |\mathbf{v}_k| > T \\ 0 & : \quad |\mathbf{v}_k| \leq T \end{cases}, \tag{7.37}$$

where $\mathbf{v}_k$ are vaguelette coefficients and a "universal" threshold criterion has been defined by Donoho (1995) as

$$T = \sigma_{\mathbf{n}}\sqrt{2\ln(N)}. \tag{7.38}$$

N is the number of coefficients in $\mathbf{z}$.

This universal threshold level comes from the fact that for N independent, unit variance, Gaussian random variables $x_1, x_2, \ldots, x_N$,

$$\lim_{N \mapsto \infty} P\left[\max|x_j| > \sqrt{2\ln N}\right] = 0. \tag{7.39}$$

Applying this to our inverse method, this basically says that if the noise $\mathbf{n}$ is white with unit variance, the probability of getting a coefficient of vaguelette noise above $\sqrt{2\ln N}$ tends to zero as N gets larger.

A difficulty in Equation 7.32 is how to implement $\mathbf{U}$. Creating a vaguelette from each individual wavelet would be tedious and inefficient. Noting the similarities between wavelets and vaguelettes, Kolaczyk (1994, page 4) has said,

> [Scale invariance of vaguelettes] should enable the development of fast algorithms for the computation of the vaguelette coefficients, at which point methods similar to the wavelet shrinkage of Donoho et al. become

feasible.

This seems to imply that there might exist a *fast vaguelette transform* akin to the fast wavelet transform. For convolution operators, the method of Beylkin and Torrésani (1996) may be useful for this purpose, but we do not investigate it here. Noting that $\mathbf{U} = \mathbf{\Gamma}\mathbf{W}\mathbf{K}^{-1}$, we see an alternate way of generating the vaguelette coefficients. We substitute this expression into Equation 7.32 and obtain

$$\hat{\mathbf{f}} = \mathbf{W}^{-1}\mathbf{\Gamma}^{-1}\Theta_T[\mathbf{\Gamma}\mathbf{W}\mathbf{K}^{-1}\mathbf{z}]. \tag{7.40}$$

To implement this Equation we simply need to be able to apply $\mathbf{K}^{-1}$, in addition to the wavelet transform.

The intuitive idea behind Equation 7.40 is this: solving Equation 7.7 leads to a noise contaminated solution. Transforming this solution to the wavelet domain according to Equation 7.40 and weighting with $\mathbf{\Gamma}$ tends to isolate "good" signal into a few large valued, isolated coefficients, while the noise tends to be spread around equally with smaller energy. Thus thresholding the small vaguelette coefficients will tend to remove noise and leave the more interesting and coherent features untouched. Inverse weighting with $\mathbf{\Gamma}^{-1}$ and inverse wavelet transforming will result in an adaptively smoothed solution. It effectively *regularizes* the solution without any prior information other than the distribution of the noise in the data, and the assumed differentiability of the underlying function (needed for the wavelet transform).

In examining the computational cost of the WVD inversion method in Equation 7.40 we see that the most expensive part may be applying $\mathbf{K}^{-1}$. This cost depends greatly on the operator $\mathbf{K}$. If it is a convolution matrix, Fourier methods could still be applied. As mentioned above, a forward and inverse wavelet transform each cost $O(N)$ if the wavelet is of compact support, otherwise it can be done in the Fourier domain in $O(N \log N)$ operations (Blu and Unser, 2001a). Thresholding is an extremely fast $O(N)$ operation on a vector. Calculating the quasi-singular values via Donoho's formula for homogeneous operators is a simple $O(N)$ operation. Hence the efficiency of the WVD inverse method is limited by the computational cost of applying $\mathbf{K}^{-1}$.

7.3.1 Deriving the γ's

We will deviate from Donoho's framework here and derive the γ's as a function of the energy of the noise $\mathbf{n}$. The reason to do this is that from experience it has been observed that the vaguelettes deviate too much from the orthogonality assumption to believe the noise has been evenly distributed among the vaguelette coefficients.

This will not be a problem if we can successfully map the covariance of $\mathbf{n}$ to the covariance of the vaguelette coefficients. In fact, it may allow us to generalize the WVD method. To see this we temporarily assume that $\mathbf{K}$ and $\mathbf{C_n}$ are square circular convolution matrices and $\mathbf{K}$ is invertible. We also slightly redefine Equation 7.28 into

$$\mathbf{K}\mathbf{W}^T = \tilde{\mathbf{V}}^T. \tag{7.41}$$

This differs in that no normalizing factors are removed from the vaguelettes to make $\mathbf{\Gamma}$. We then have

$$\mathbf{K} = \tilde{\mathbf{V}}^T\mathbf{W}. \tag{7.42}$$

From this we have

$$\mathbf{K}^{-1} = \mathbf{W}^T\tilde{\mathbf{U}}, \tag{7.43}$$

where $\tilde{\mathbf{U}} = \left(\tilde{\mathbf{V}}^T\right)^{-1}$. We clearly have that $\tilde{\mathbf{U}} = \mathbf{W}\mathbf{K}^{-1}$. Recalculating the covariance of the new vaguelette coefficients we have

$$\begin{aligned}
\mathbf{C}_{\tilde{v}} &= E\left[\tilde{\mathbf{U}}\mathbf{n}\mathbf{n}^T\tilde{\mathbf{U}}^T\right] && (7.44)\\
&= E\left[\mathbf{W}\mathbf{K}^{-1}\mathbf{n}\mathbf{n}^T(\mathbf{W}\mathbf{K}^{-1})^T\right] && (7.45)\\
&= \mathbf{W}\mathbf{K}^{-1}E\left[\mathbf{n}\mathbf{n}^T\right](\mathbf{W}\mathbf{K}^{-1})^T && (7.46)\\
&= \mathbf{W}\mathbf{K}^{-1}\mathbf{C}_{\mathbf{n}}\mathbf{K}^{-T}\mathbf{W}^T && (7.47)\\
&= \mathbf{W}\underbrace{\mathbf{K}^{-1}\mathbf{C}_{\mathbf{n}}^{1/2}}_{\mathbf{Q}}\mathbf{C}_{\mathbf{n}}^{1/2}\mathbf{K}^{-T}\mathbf{W}^T && (7.48)\\
& && (7.49)
\end{aligned}$$

The matrix $\mathbf{Q}$ is also a square circular convolution matrix. It is diagonalized by the Fourier transform and has a spectrum $\Lambda_{\mathbf{Q}}$. Remembering that a wavelet transform is achieved by a convolution and downsampling process, we can calculate the noise energy at any scale of wavelet coefficients by multiplying $\Lambda_{\mathbf{Q}}$ by the Fourier transform of a wavelet at that scale, and calculating the energy of the resulting amplitude spectrum. This energy would be the standard deviation of the vaguelette noise at that scale.

Such an analysis of the noise in the vaguelette coefficients is not limited to convolutional operators but will become more complicated with other operators. Even if the noise is still stationary, if the operator $\mathbf{K}$ is not convolutional, there is not a straightforward way of calculating the noise distribution on the vaguelette coefficients. A brute force Monte Carlo method could be used where multiple realizations of $\mathbf{n}$ are generated and converted into realizations of $\tilde{\mathbf{n}} = \mathbf{W}\mathbf{K}^{-1}\mathbf{n}$. Then the standard deviations of each element of $\tilde{\mathbf{n}}$ could be empirically estimated.

Once these standard deviations are calculated, their inverses become the elements of the diagonal of a matrix $\tilde{\mathbf{\Gamma}}$. Then the result of $\tilde{\mathbf{n}} = \tilde{\mathbf{\Gamma}}\mathbf{W}^T\mathbf{K}^{-1}\mathbf{n}$ would be a random vector that has unit variance (but not a diagonal covariance matrix). This is exactly what we want because we can now apply the $T = \sqrt{2\ln N}$ threshold uniformly to each coefficient. This is the method used to calculate the $\tilde{\mathbf{\Gamma}}_{k,k}$'s in the applications that follow. Our WVD based estimator now takes the form

$$\hat{\mathbf{f}} = \mathbf{W}^T\tilde{\mathbf{\Gamma}}^{-1}\Theta_T[\tilde{\mathbf{\Gamma}}\mathbf{W}\mathbf{K}^{-1}\mathbf{z}]. \tag{7.50}$$

There is one glaring problem that we have not addressed in the linear algebraic framework presented here. Since we are not using any regularizing term in the inverse, how can we apply the method if **K** is not invertible? There are two options. The first is to add a very small amount of damping to make the operator invertible. We do not recommend this because, although it gives good results, it essentially inputs extra information into the problem. We want to extract all our information from the data itself. Therefore we recommend applying a **pseudo-inverse** (Nashed and Wahba, 1974) when **K** is not invertible. This method ignores singular values that equal zero when applying an inverse and leaves any signal present in the data untouched (although potentially drowned in noise). This is essentially what Donoho (1995) does when addressing *weakly invertible* homogeneous operators. If an important feature of the model is lost because of the pseudo-inverse, it can only be obtained in the form of prior information. This is what is done in the slowness estimation problem in the next section. The pseudo-inverse can be calculated directly by the SVD for small matrices, or indirectly via iterative methods for larger problems.

7.3.2 Slowness estimation

In this section we compare classical least-squares inversion methods with the thresholded WVD method. We apply all methods to the problem of slowness estimation from traveltime data. This kind of problem arises in VSP experiments, where an impulsive source at the surface is received at sensors within a well. The sensors are usually uniformly distributed at depth in the well and measure the time it takes for the signal to travel from the surface to each depth. Mathematically, the traveltime can be modeled as a cumulative integration of the slowness of the rock formation:

$$\mathrm{K} = \int_0^x [-]du, \tag{7.51}$$

where x is depth. In the Fourier domain this operator has a singularity at zero frequency. Projecting this operator into our spline basis results in a lower triangular matrix. This makes it hard to compute the $\tilde{\mathbf{\Gamma}}_{k,k}$'s by our method without resorting to expensive Monte Carlo methods. As we mentioned above, wavelet based methods are more tractable when the K is a circular convolutional operator. We will therefore remove the singularity of the operator and simplify the problem by assuming a known mean. Edge effects aside, this allows us to model the integration operator and its inverse as convolution operators. Assuming a known mean changes Equations 7.7 and 7.11, respectively, to

$$\hat{\mathbf{f}} = \boldsymbol{\mu}_{\mathrm{f}} + \mathbf{K}_{ls}^{-1}\left(\mathbf{z} - \mathbf{K}\boldsymbol{\mu}_{\mathrm{f}}\right) \tag{7.52}$$

and

$$\hat{\mathbf{f}} = \boldsymbol{\mu}_{\mathrm{f}} + \mathbf{K}_{rls}^{-1}\left(\mathbf{z} - \mathbf{K}\boldsymbol{\mu}_{\mathrm{f}}\right), \tag{7.53}$$

where $\boldsymbol{\mu}_{\mathrm{f}}$ is the known mean vector of $\mathbf{f}$. We will perform inversion first on a synthetic data set and then on a real one. The synthetic data is generated from a real slowness log shown in Figure 7-5. To make the solution easier to visually compare with the true model we will block the log by thresholding coefficients of its wavelet transform. We use orthogonal fractional spline wavelets with $\alpha = 1.25$ for the blocking. Our blocked model now has Hölder continuity of degree $\alpha = 1.25$. We assume we know this prior information on the differentiability of the model and hence use the same order wavelets in the WVD inversion. The blocked log is shown in Figure 7-6. This blocked model is cumulatively integrated and stationary white noise with standard deviation $\sigma_{\mathbf{n}} = 0.005 s/m$ is added to the data. The traveltime data are shown in Figure 7-7. For the synthetic inversion we calculate the mean of the true slowness model beforehand and assume this to be known in all estimations of the model.

Our job now is to invert this data set and try to get back original slowness log in Figure 7-6. In doing this we will compare three different methods:

1. WVD based inversion via Equation 7.50.

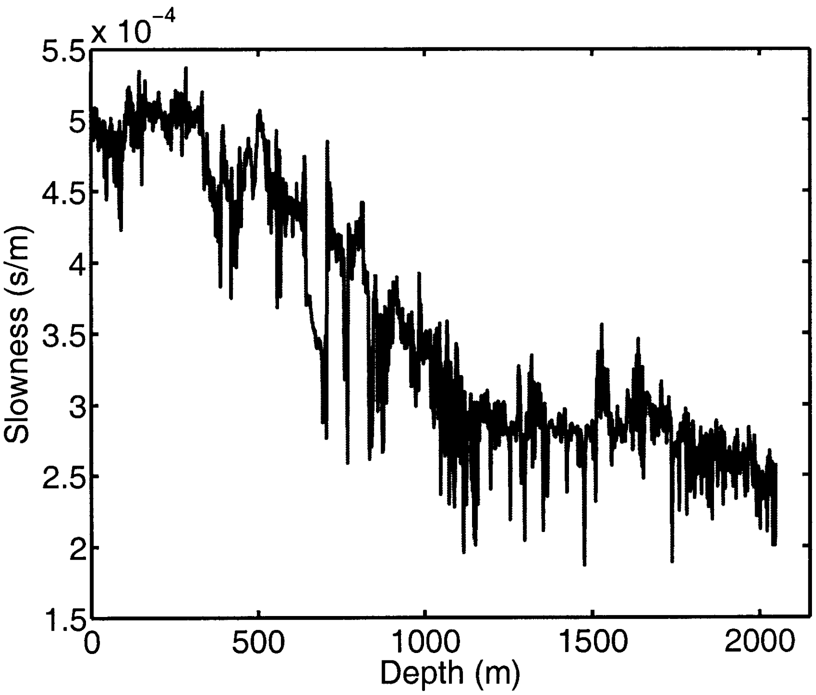

Figure 7-5: Slowness well log.

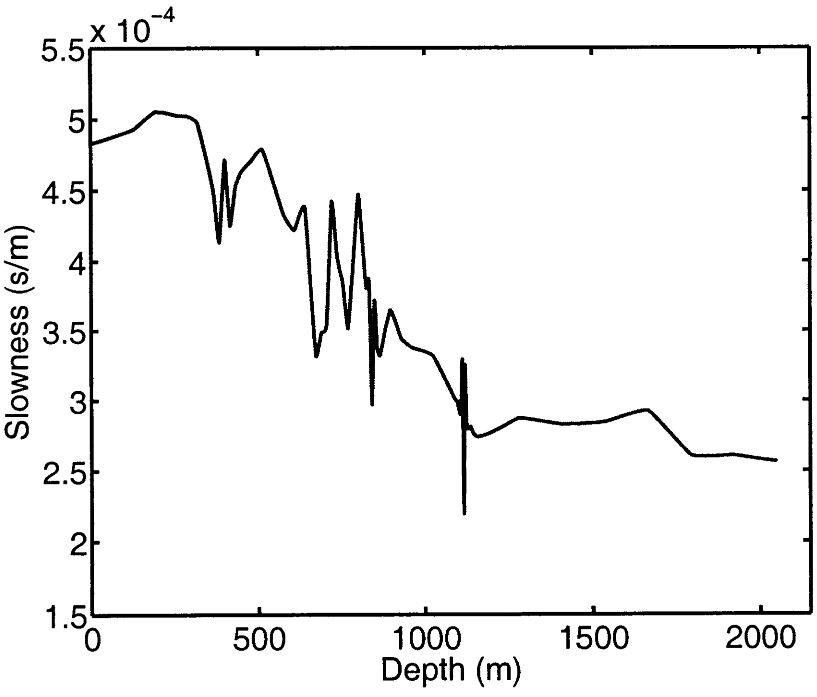

Figure 7-6: Blocked slowness well log.

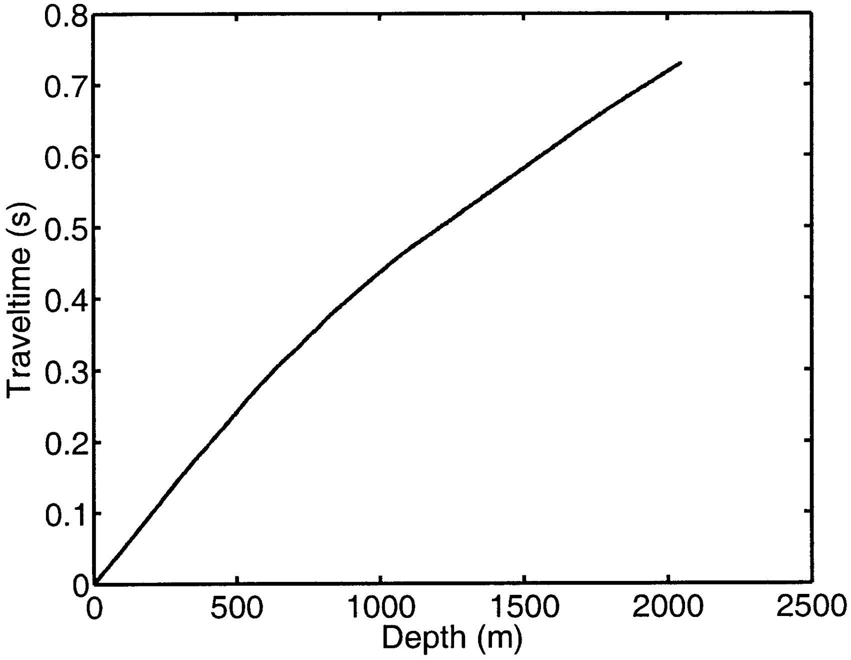

Figure 7-7: Noisy synthetic traveltime data.

2. Damped least squares according to Equation 7.7.

3. Regularized least squares according to Equation 7.11.

We first graphically go through the steps of WVD thresholded inversion. Before any inversion is performed, edge effects are minimized by symmetric extension of the data (Strang and Nguyen, 1997). Then, following Equation 7.50, we first apply $\mathbf{K}^{-1}$. Since there is an ignored eigenvalue in this convolutional inverse, we are essentially performing a pseudo-inverse. This gives rise to the noise contaminated solution shown in Figure 7-8. We next apply the orthogonal fractional spline wavelet transform to the noisy estimate. This is shown in Figure 7-9. It is followed by the weighting with the $\tilde{\mathbf{\Gamma}}$ as shown in Figure 7-10. The universal threshold is also shown here by the red lines. We pause here to show in Figure 7-11 the difference in γ's between our method and the method of Donoho (1995). We can see that Donoho's method underweights small scale coefficients and overweights large scale ones. We can demonstrate that our $\tilde{\mathbf{\Gamma}}$ does what it should by removing the signal $\mathbf{Kf}$ from our data and passing only noise, $\mathbf{n}$ through the WVD algorithm. We show such a result in Figure 7-12. This plot shows us two things: 1) the variance of the noise appears to be uniformly

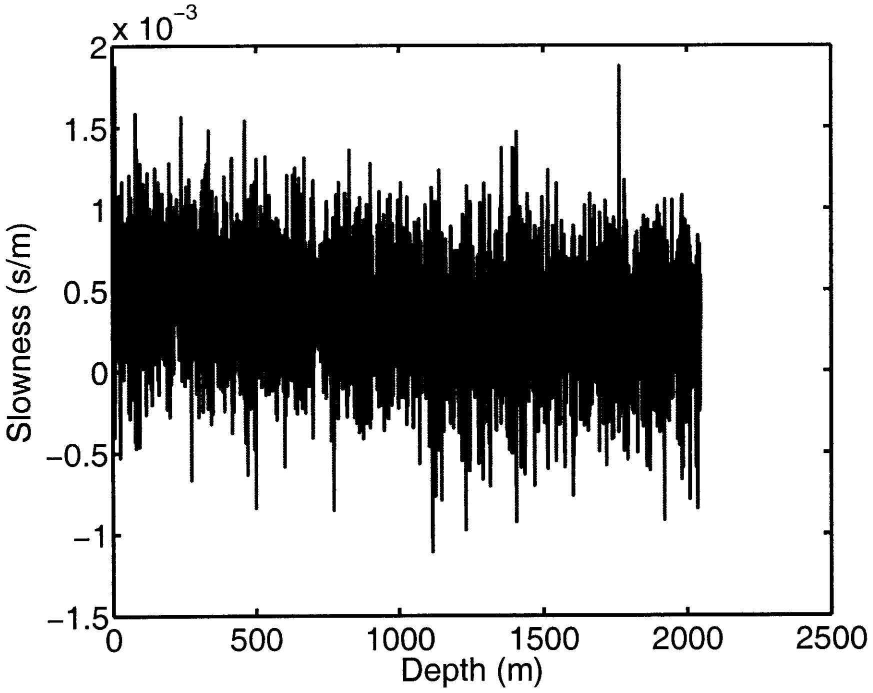

Figure 7-8: $\mathbf{K}_{ls}^{-1}\mathbf{z}$

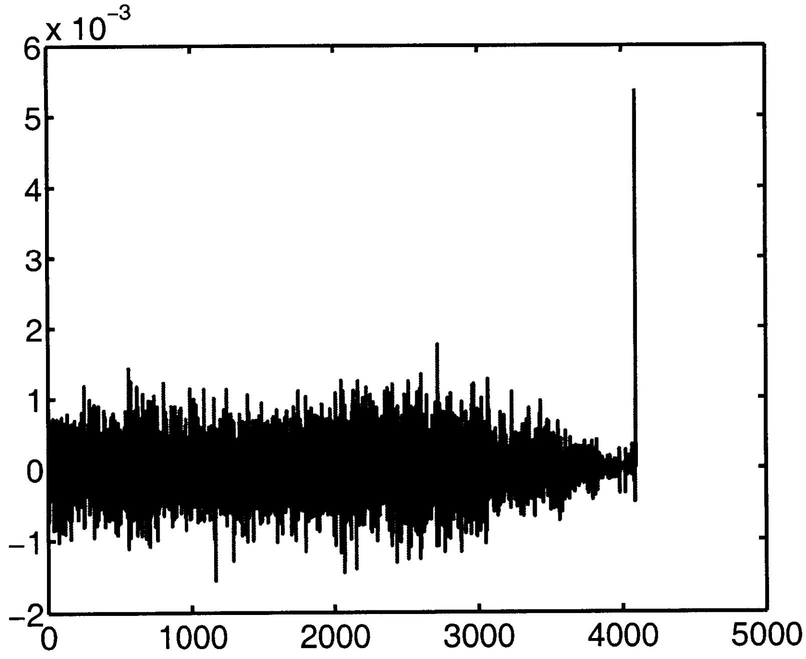

Figure 7-9: $\mathbf{W}\mathbf{K}_{ls}^{-1}\mathbf{z}$

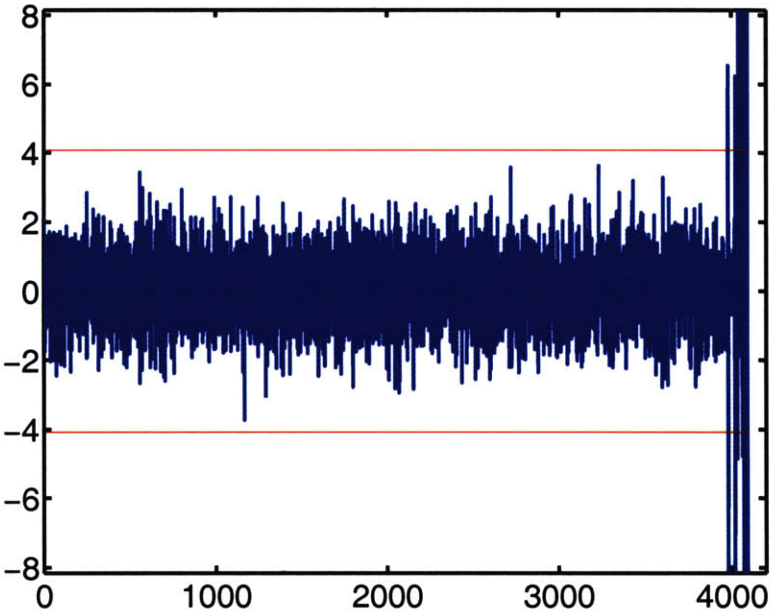

Figure 7-10: $\tilde{\mathbf{\Gamma}}\mathbf{W}\mathbf{K}_{ls}^{-1}\mathbf{z}$

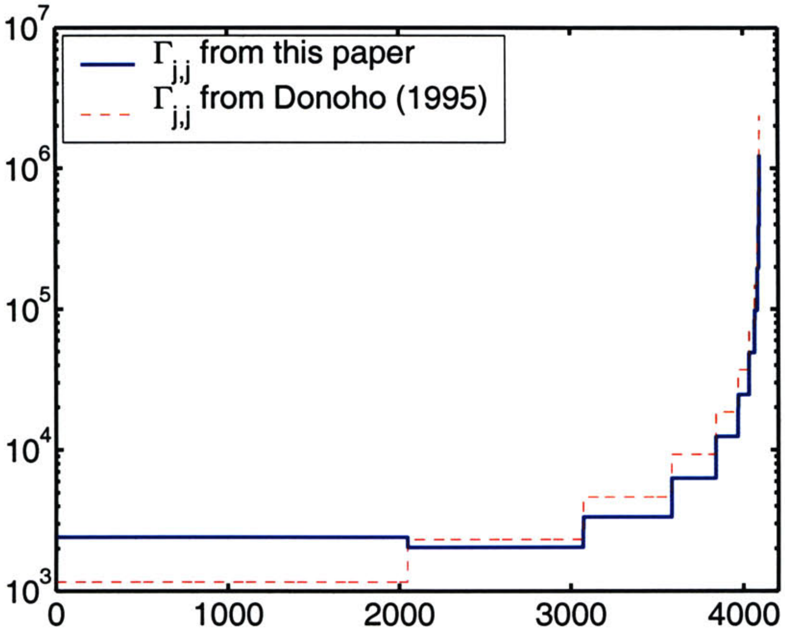

Figure 7-11: γ's calculated via our method and that of Donoho (1995).

distributed, and, 2) It does not exceed the universal threshold level, as it shouldn't. We also show in Figure 7-13 the result of removing all *noise* from our data and passing only $\mathbf{Kf}$ through the WVD method. From this plot we can see the distribution in the vaguelette domain of the signal we wish to recover.

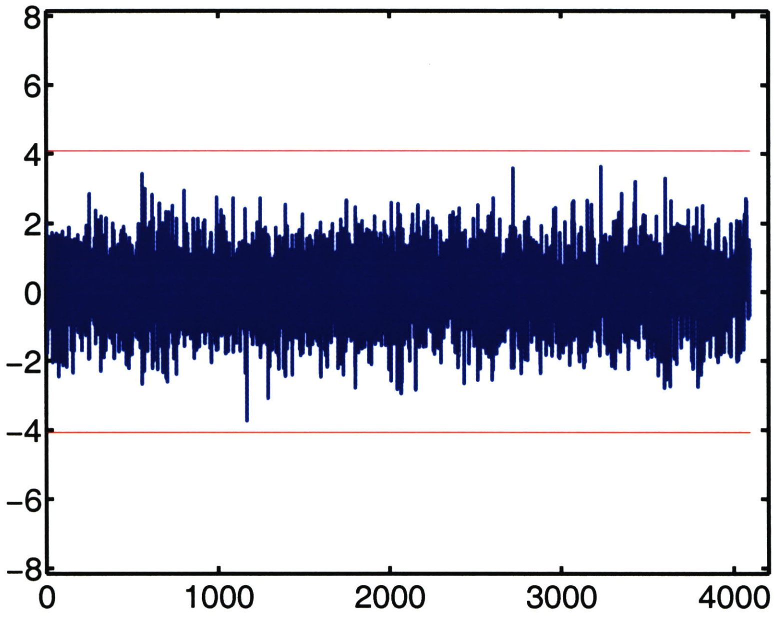

Figure 7-12: $\tilde{\mathbf{\Gamma}}\mathbf{W}\mathbf{K}_{ls}^{-1}\mathbf{n}$

Moving on, we both threshold the vaguelette coefficients from the noisy data and unweight with $\tilde{\mathbf{\Gamma}}^{-1}$ in Figure 7-14. From this plot we can see how our solution model is greatly compressed into a few coefficients in the wavelet domain. This shows how the compressional properties of the wavelet transform have lent themselves to the inversion. Finally we inverse wavelet transform and arrive at the solution in Figure 7-15.

It is interesting to see what the effects on the WVD inversion method would be if the degree of the fractional spline wavelet was incorrectly chosen. In Figure 7-16 we show the mean-square error (MSE) of the WVD method for varying values of α. The curve is convex and the error remains at the same order of magnitude for all α values. This tells us that the WVD inversion method is insensitive to incorrectly chosen α.

We now move on to the more classical regularized least-squares methods. In Figures 7-17 and 7-18 we show, respectively, the regularized least-squares and damped

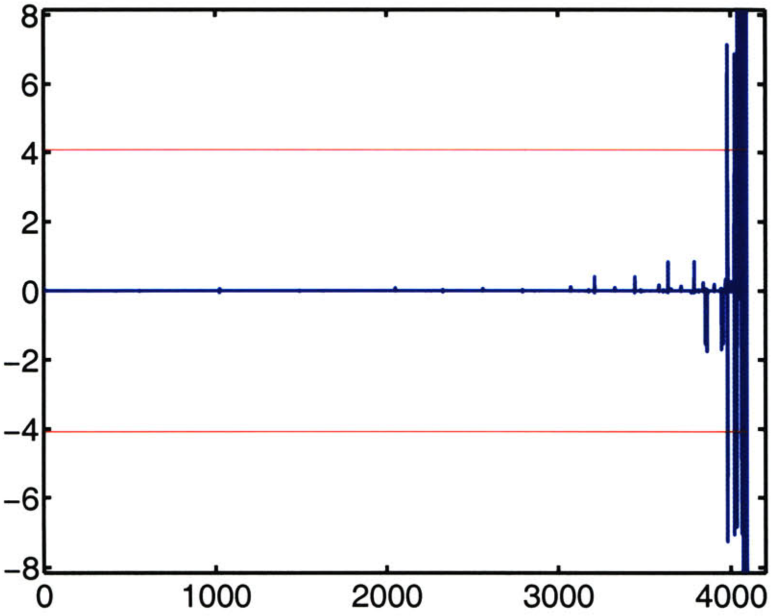

Figure 7-13: $\tilde{\mathbf{\Gamma}}\mathbf{W}\mathbf{K}_{ls}^{-1}\mathbf{K}\mathbf{f}$

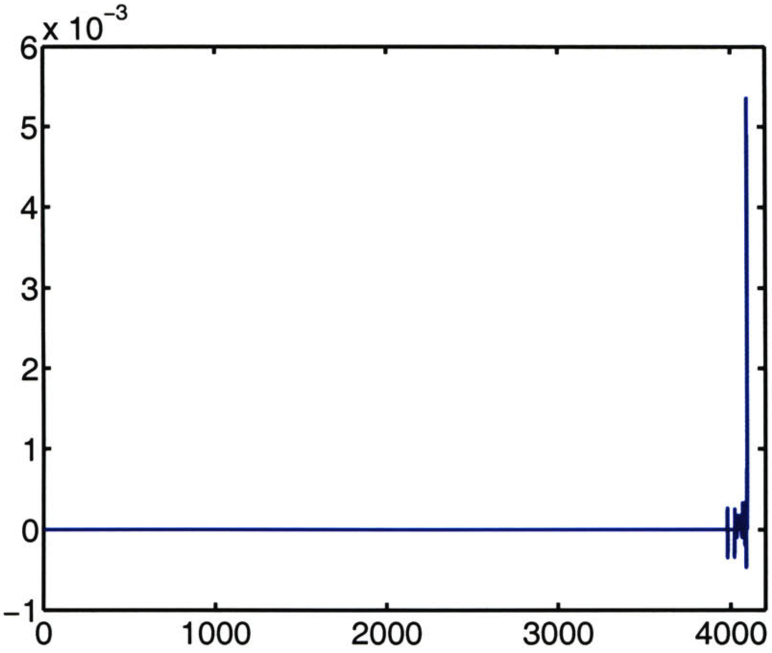

Figure 7-14: $\tilde{\mathbf{\Gamma}}^{-1}\Theta_T\left[\tilde{\mathbf{\Gamma}}\mathbf{W}\mathbf{K}_{ls}^{-1}\mathbf{z}\right]$

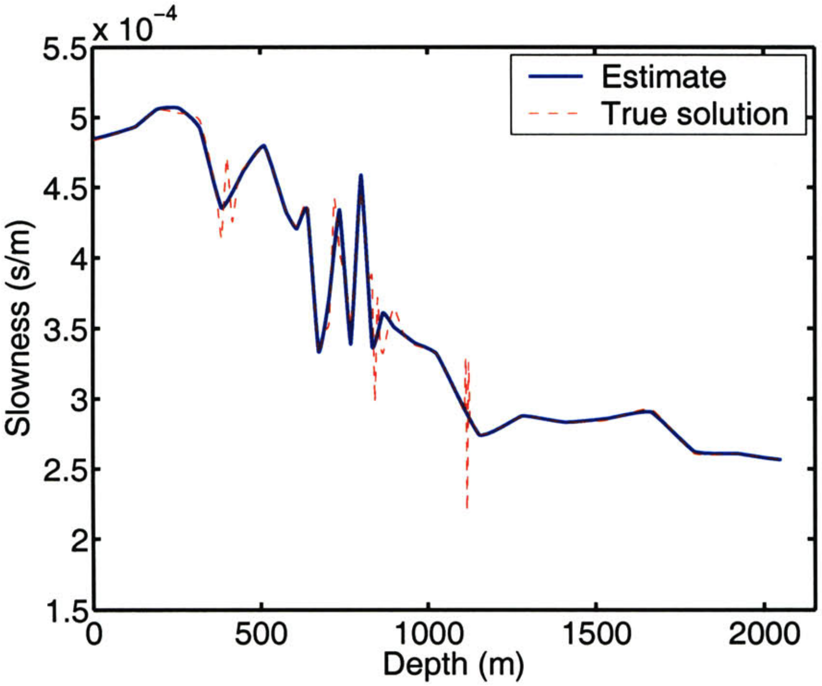

Figure 7-15: $\mathbf{W}^T\tilde{\mathbf{\Gamma}}^{-1}\Theta_T\left[\tilde{\mathbf{\Gamma}}\mathbf{W}\mathbf{K}_{ls}^{-1}\mathbf{z}\right]$

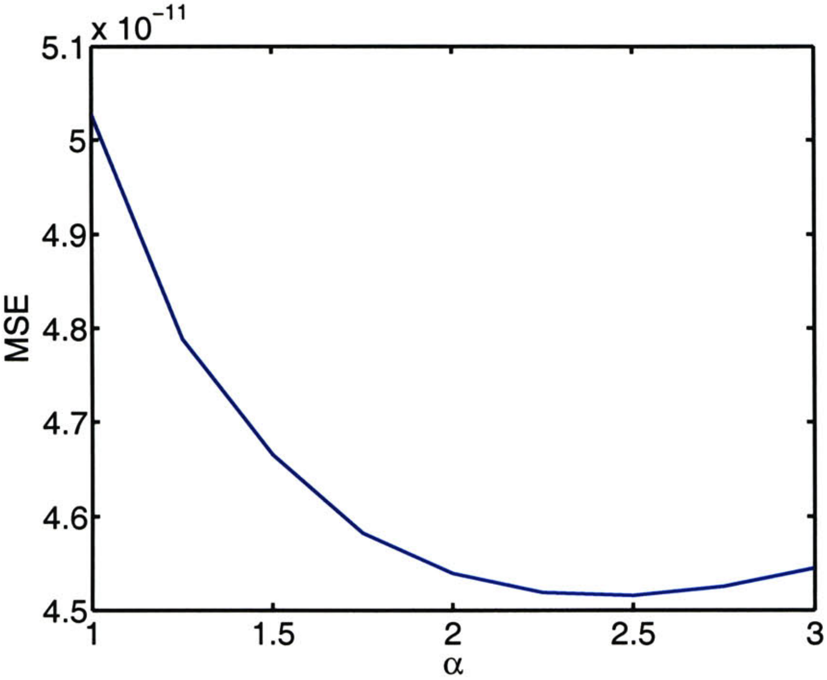

Figure 7-16: MSE of WVD inversion for varying degree α.

least-squares solutions from Equations 7.52 and 7.53. In order to implement these

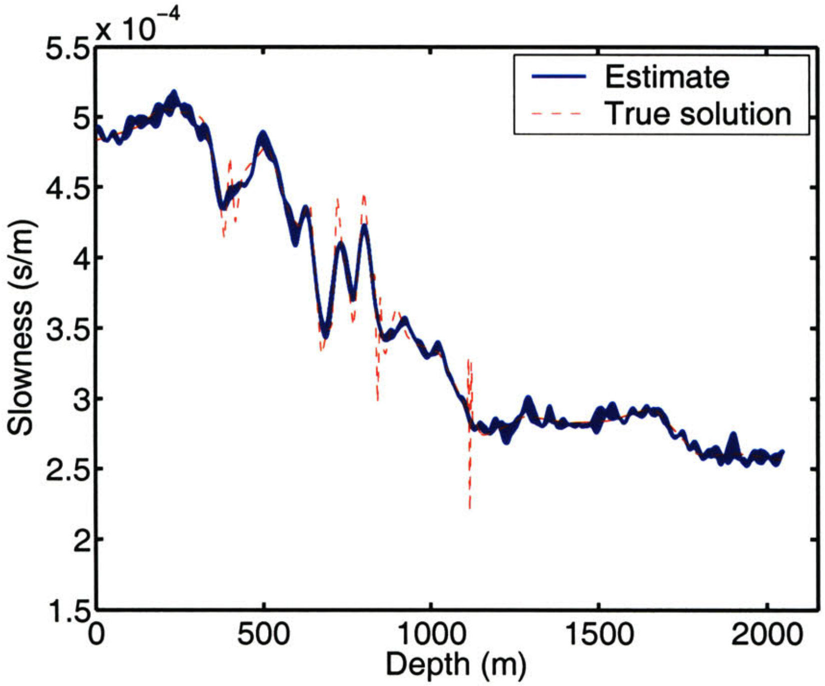

Figure 7-17: Solution via regularized least-squares.

solutions the parameter ζ must be set. Since this is a synthetic problem we can run the regularized least-squares inversion multiple times and measure the MSE. This is shown in Figure 7-19 for damped least-squares over for range of ζ values. For all comparisons between regularized least squares and WVD based inversion we run the regularized inversion with the optimal value of ζ obtained from such a plot. Thus for all comparisons between the methods we are using the best possible regularized inversion result, which is a better solution than would probably be obtained in reality.

To correctly compare the WVD based inverse method with regularized and damped least squares methods, we vary the standard deviation of the noise over a large range and compare the MSE of each method for each value of $\sigma_{\mathbf{n}}$. We plot the results of this comparison in Figure 7-20. This is the most important plot in this work. It shows that the WVD based inverse method consistently outperforms the classical methods over an order of magnitude range of noise on the data. For values of $\sigma_{\mathbf{n}}$ greater than those plotted in Figure 7-20 the noise is so large that essentially only the mean is recovered from the data. For values of $\sigma_{\mathbf{n}}$ less than those plotted, there is so little

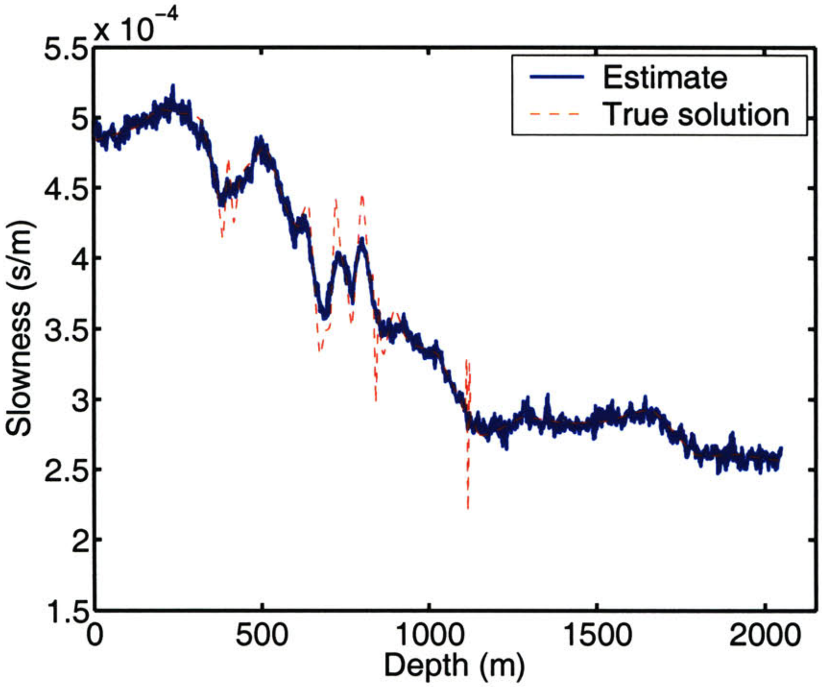

Figure 7-18: Solution via damped least-squares.

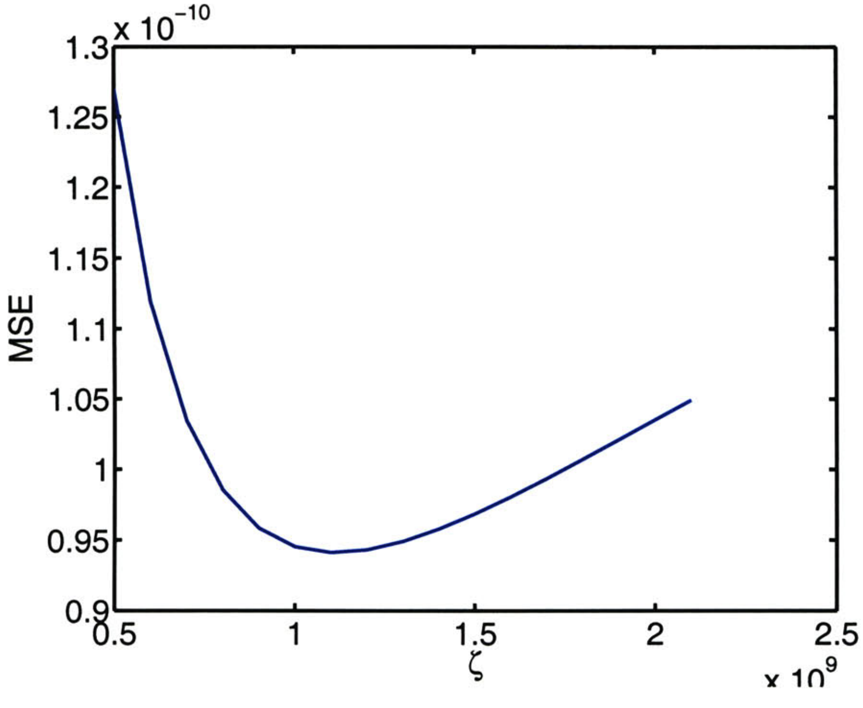

Figure 7-19: MSE of damped least-squares inversion for varying ζ.

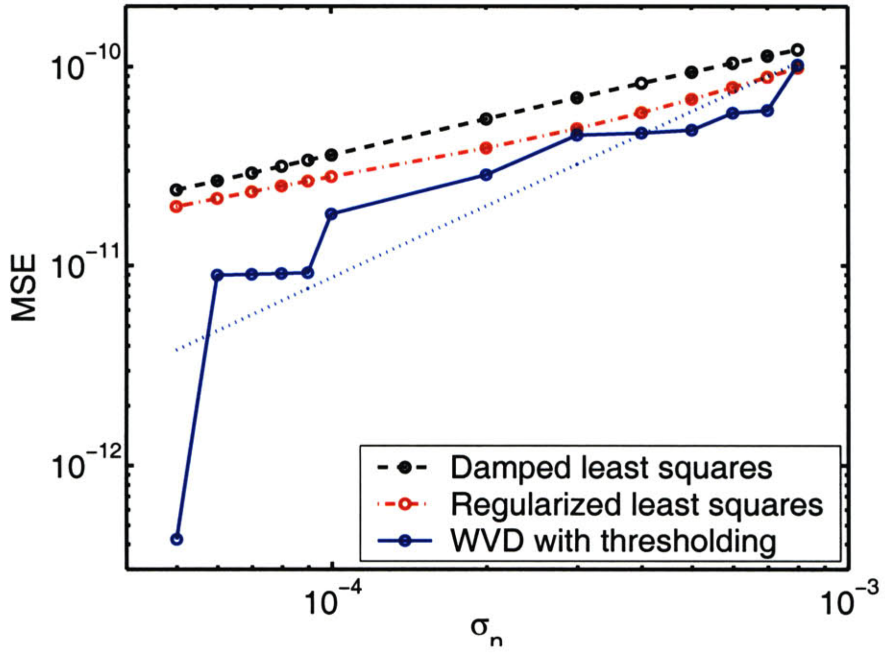

Figure 7-20: MSE of of all inversion method for varying $\sigma_{\mathbf{n}}$.

noise on the data that the MSE will not significantly decrease for the WVD method.

7.3.3 Inversion of real data

We now apply each of the methods compared above to the inversion of real traveltime data shown in Figure 7-21. We did not have information as to the noise level on this data so an intuitively plausible level of $\sigma_{\mathbf{n}} = 0.0003$ was chosen. We again use orthogonal spline wavelets of degree $\alpha = 1.25$. We plot the WVD based inversion result in Figure 7-22. It has clearly done an adaptive estimation. We see interesting features appearing at 500 meters, 1300 meters, and 3500 meters. The rest of the estimate is smooth by comparison. Such a signal adaptive estimate as this is the whole reason for developing the WVD based method.

For comparison we plot the regularized least-squares solution in Figure 7-23. ζ was chosen by trial and error, looking for a solution that neither looks too rough or too smooth. The solution for damped least-squares inversion looked almost the same as regularized least-squares solution so we do not show the latter result here. It is difficult to say whether the small scale detail in the damped least-squares estimate is

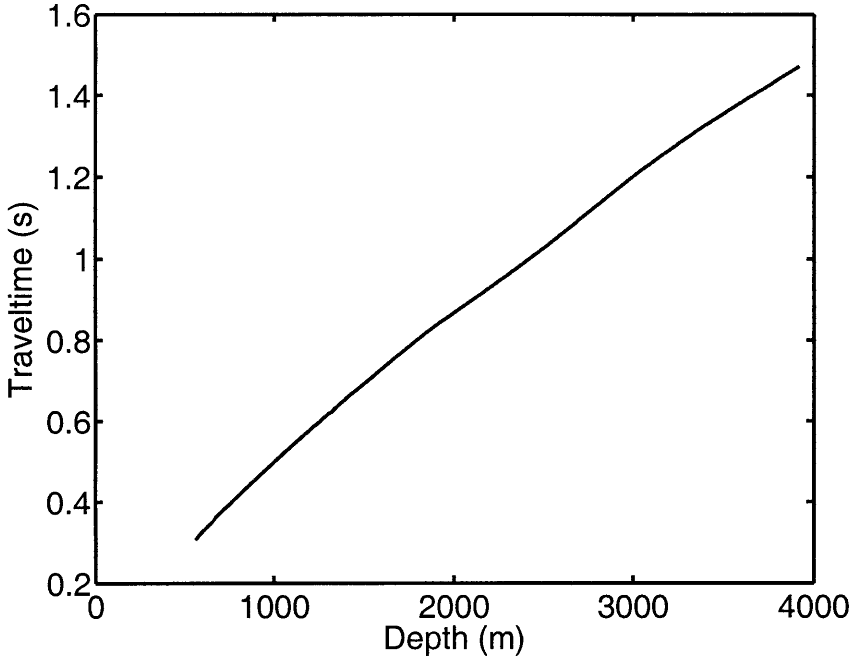

Figure 7-21: Real traveltime data.

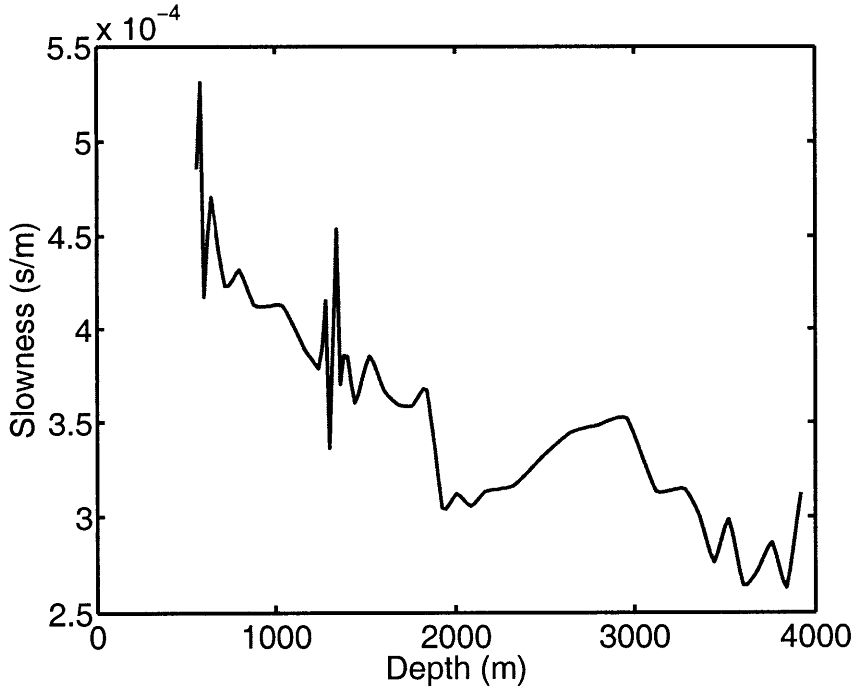

Figure 7-22: WVD based inversion of real data.

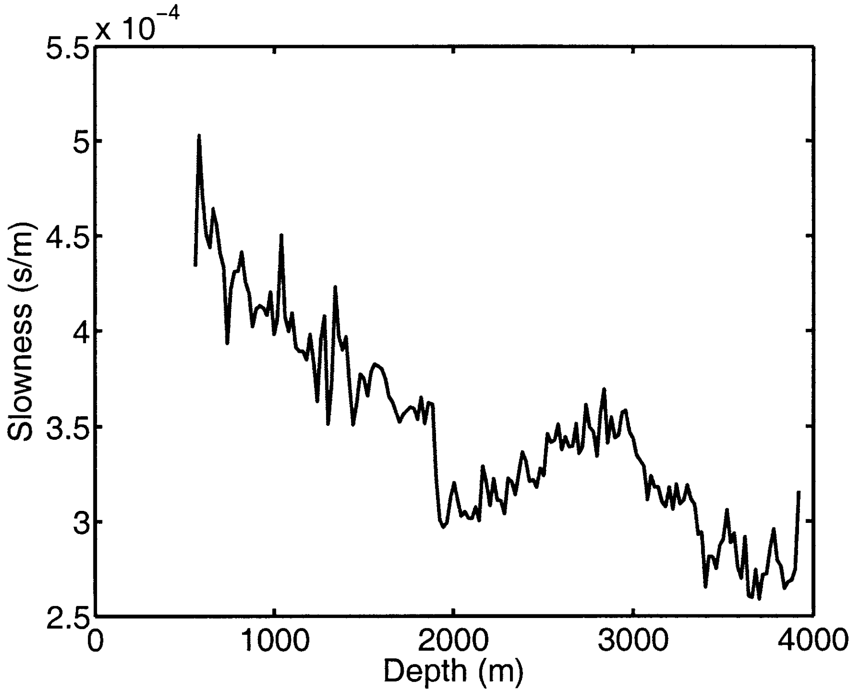

Figure 7-23: Regularized least-squares inversion of real data.

noise or signal.

7.4 Conclusions

The work presented here introduces the wavelet-vaguelette decomposition to the geophysical community and presents its use in solving linear inverse problems via a thresholding operation on the vaguelette coefficients. WVD method was compared to other wavelet based methods used to solve inverse problems and then applied to a 1-D synthetic inverse problem, that of slowness estimation from traveltime data. The WVD method was shown to consistently outperform classical least-squares method in terms of MSE over a large range of noise magnitudes on the traveltime data. The method was also applied to a real geophysical data set and the results examined.

This work also presents fractional splines as a good basis for representing geophysical data and creating orthogonal wavelets. To our knowledge, this is the first application combining the WVD methodology with fractional spline wavelets.

7.5 Acknowledgments

We thank the ERL Founding Members Consortium for its support of this project. We also thank Dr. Bill Rodi for many useful conversations about inverse theory. We thank Drs. Unser and Blu for use of their fractional spline software.

Chapter 8

Conclusions

In this chapter we summarize the major contributions and limitations of the thesis. Possible future directions for the research are also postulated.

8.1 Joint deconvolution/interpolation of geophysical data

The first contribution of this thesis was a statistically optimal way to deconvolve a blurred image for an improved model, while simultaneously interpolating sparse samples of the same model. Efficient solution of this problem required a non-standard derivation of interpolation. Classical minimum variance based kriging methods were not applicable. A careful analysis of the structure of the problem was also needed for efficient computation of the solution via the conjugate gradients algorithm.

Chapter 3 applied the method to an application in reservoir characterization: deconvolving (deblurring) a 3-D poststack seismic data cube constrained by existing well data. The joint inversion led to a solution with super resolution within a correlation length around the well locations, and laterally correlated geologic features throughout the deconvolved cube. This solution was superior to that obtained by using only one of the data sets.

Chapter 4 applied the same joint inversion method to a different application:

integrating synthetic DEM and GPS data for an optimal topographic elevation model. The best results were again obtained when both data sets were used.

8.1.1 Limitations and future directions

1. Missing from the joint inversion method is a way to calculate posterior covariance information. The matrices are too big to calculate the posterior covariance matrix explicitly. Recent work (Schneider, 2000) has shown the possibility of obtaining the posterior variances as a by-product of the conjugate gradients algorithm. Instability problems, however, preclude this working on anything but a small problem, or problems requiring only a few iterations of the algorithm.

2. It was shown in Chapter 3 that for large dimensional joint inverse problems it is easier to perform regularized inversion with the inverse of the covariance matrix rather than with the covariance matrix itself. It would be helpful to have a flexible, parameterized class of covariance matrices which have simple and sparse inverses.

3. Our particular application of joint inversion was for two simple linear forward modeling operators: convolution and picking. Generalizing the method for arbitrary linear operators is conceptually simple, but will require exploitation of the structure of the operators in order to be computationally efficient. Such structure will change from problem to problem.

   More interesting is what to do in the case of nonlinear operators. Many interesting geophysical problems are inherently nonlinear. One loses information if they are linearized. The joint inversion framework again handles such operators without a problem, but different algorithms are needed to solve the problem. The solution is no longer a matrix inversion. Perhaps Monte Carlo methods such as Gibbs sampling, the Metropolis method, sample rejection, or genetic algorithms would be adequate to obtain a solution without too many iterations. The first three of these methods can also provide statistics of the posterior distribution.

4. More care needs to be taken in obtaining the optimal regularization parameter. In Chapters 3 and 4 it was done empirically by trial and error. Methods such as cross-validation or the L-curve (Vogel, 2002) could be applied.

8.2 Wavelet domain geophysical inversion

The second major contribution of this thesis was to bring a new wavelet based inversion method to geophysical problems. Our method was based on a previous method of Donoho (1995). As opposed to regularized inverse methods, the wavelet based method inputs less prior information into the problem, and extracts more information from the data. It results in a solution that has been smoothed more in more smoothly varying areas, and smoothed less in regions with sharp transitions, thereby preserving geologically interesting features. Inversion of a synthetic 1-D problem showed it to consistently outperform the classical regularized solutions over a broad range noise magnitude on the data.

8.2.1 Limitations and future directions

1. A limitation of the WVD based inverse method is that it is more computationally expensive. In our method the threshold level, γ, had to be calculated in advance of the inversion. It was computationally efficient in the case of a convolutional forward operator. In the case of an arbitrary forward modeling operator it would be much more difficult. Monte Carlo realizations of the noise would have to be made, each passed through the generalized inverse and wavelet transform in order to calculate statistics on the wavelet coefficients.

2. Beylkin and Torrésani (1996) implement convolution operators via filter banks. Implementing this would greatly speed up WVD based inversion in the case of convolution forward operators.

3. Since the WVD based inversion is not a Bayesian method, there exists no posterior PDF. It is still theoretically possible to estimate statistics of the estimate.

How to do this is not clear, though.

4. The WVD based inverse method gave impressive results but was limited to 1-D. To make the method useful for real geophysical applications one needs to generalize wavelets to higher dimensions, a not so trivial task. The most common way is to use tensor outer products of 1-D wavelets to define the 2-D wavelets, but this is not optimal. In a 2-D image, singularities tend to align themselves on curves. The optimal wavelet basis would therefore be one where the wavelets are perpendicular to the curves, thus triggering on the singularities. Recent work by Candès and Donoho (2000) and Starck et al. (2000) has introduced a new class of wavelet like functions, called **curvelets**, that align themselves on curved singularities. It would be interesting to apply such functions in the WVD framework of Chapter 7.

8.3 Fractional splines and discretization

Chapter 5 illustrates the non-triviality of manipulating discrete data. The simple example of the derivative operator in Section 5.3 shows the possible error arising from incorrect manipulation of coefficients.

8.3.1 Limitations and future directions

1. What about multi-fractals? Splines are great when approximating a function that is uniformly Hölder α. If the Hölder exponent changes spatially, it is not clear which exponent to use for the spline.

2. A geophysical example needs to be found, using real data, that more clearly illustrates the danger of not treating discrete data correctly.

Bibliography

Abramovich, F. and Silverman, B. W. (1998). Wavelet decomposition approaches to statistical inverse problems. *Biometrika*, 85(1):115–129.

Al-Moqbel, A. M., Kane, J. A., and Töksoz, M. N. (2002). Reservoir characterization from seismic attributes: Case study. *Geophysics*. Submitted for publication.

Aldroubi, A., Unser, M., and Eden, M. (1992). Cardinal spline filters: stability and convergence to the ideal sinc interpolator. *Signal Processing*, 28(2):127–138.

Alpert, B., Beylkin, G., Coifman, R., and Rokhlin, V. (1993). Wavelet-like bases for the fast solution of second-kind integral equations. *SIAM Journal on Scientific Computation*, 14(1):159–184.

Amaratunga, K. and Castrillon-Candas, J. E. (Accepted December 2000). Surface wavelets: A multiresolution signal processing tool for 3D computational modeling. *International Journal for Numerical Methods in Engineering.*

Battle, G. (1987). A block spin construction of ondelettes. Part 1: Lemarié functions. *Comm. Math. Phys.*, 110:601–615.

Bertero, M. and Boccacci, P. (1998). *Introduction to Inverse Problems in Imaging.* Institute of Physics Publishing.

Beylkin, G. (1992). On the representation of operators in bases of compactly supported wavelets. *SIAM Journal of Numerical Analysis*, 29(6):1716–1740.

Beylkin, G. and Torrésani, B. (1996). Implementation of operators via filter banks: Hardy wavelets and autocorrelation shell. *Applied and Computational Harmonic Analysis*, (3):164–185. Article No. 0014.

Blu, T. and Unser, M. (2001a). The fractional spline wavelet transform: Definition and implementation. Technical report, Swiss Federal Institute of Technology EPFL.

Blu, T. and Unser, M. (2001b). Wavelets, fractals and radial basis functions. *IEEE Transactions on Signal Processing.* To appear.

Candès, E. J. and Donoho, D. L. (2000). Recovering edges in ill-posed inverse problems: Optimality of curvelet frames. Technical report, Department of Statistics.

Choi, H. and Baraniuk, R. (1999). Interpolation and denoising of nonuniformly sampled data using wavelet-domain processing. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP'99.*

Daniel, M. M. (1997). *Multiresolution Statistical Modeling With Application To Modeling Groundwater Flow.* PhD thesis, Massachusetts Institute of Technology. Laboratory for Information and Decision Systems.

Daubechies, I. (1988). Orthonormal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, (41):909–996.

Deutsch, C. and Journel, A. (1998). *GSLIB: Geostatistical Software Library and User's Guide, 2nd ed.* Oxford University Press, Inc.

DeVore, R. A., Jawerth, B., and Popov, V. (1992). Compression of wavelet decompositions. *American Journal of Mathematics*, 114:737–785.

Donoho, D. L. (1995). Nonlinear solution of linear inverse problems by wavelet-vaguelette decomposition. *Applied and Computational Harmonic Analysis*, 2(2):101–126.

Doyen, P. M. (1988). Porosity from seismic data: A geostatistical approach. *Geophysics*, 53:1263–1275.

Eide, A. L., Omre, H., and Ursin, B. (2002). Prediction of reservoir variables based on seismic data and well observations. *Journal of the American Statistical Association*, 97(457):18–28.

Fomel, S. (2001). *Three-dimensional Seismic Data Regularization.* PhD thesis, Stanford University.

Franklin, J. N. (1970). Well-posed stochastic extensions of ill-posed linear problems. *Journal of Mathematical Analysis and Applications*, (31):682–716.

Gel'fand, I. M. and Shilov, G. E. (1964). *Generalized functions*, volume 1. Academic press.

Golub, G. H. and Van Loan, C. F. (1996). *Matrix Computations.* The Johns Hopkins University Press, third edition.

Gorell, S. B. (1995). Using geostatistics to aid in reservoir characterization. *The Leading Edge*, 14(9):967–974.

Haas, A. and Dubrule, O. (1994). Geostatistical inversion - a sequential method of stochastic reservoir modelling constrained by seismic data. *First Break*, 12(11):561–569.

Herrmann, F. J. (1997). *A scaling medium representation, a discussion on well-logs, fractals and waves.* PhD thesis, Technische Universiteit Delft.

Idier, J. and Goussard, Y. (1993). Multichannel seismic deconvolution. *IEEE Transactions on Geoscience and Remote Sensing*, 31(5):961–979.

Isaaks, E. H. and Srivastava, R. M. (1989). *An Introduction to Applied Geostatistics.* Oxford University Press.

Kaaresen, K. F. and Taxt, T. (1998). Multichannel blind deconvolution of seismic signals. *Geophysics*, 63(6):2093–2107.

Kane, J. A. (2003). *Wavelet Domain Inversion and Joint Deconvolution/Interpolation of Geophysical Data.* PhD thesis, Massachusetts Institute of Technology.

Kolaczyk, E. D. (1994). *Wavelet methods for the inversion of certain homogeneous linear operators in the presence of noisy data.* PhD thesis, Dept. of Statistics, Stanford University.

Kolaczyk, E. D. (1996). A wavelet shrinkage approach to tomographic image reconstruction. *Journal of the American Statistical Association*, 91(435):1079–1090.

Lavielle, M. (1991). 2-D Bayesian deconvolution. *Geophysics*, 56(12):2008–2018.

Lemarié, P. G. (1988). Ondelettes a localisation exponentielle. *J. Math Pures et Appl.*

Leung, W. V., Bones, P. J., and Lane, R. G. (2001). Statistical interpolation of sampled images. *Optical Engineering*, 40(4):547–553.

Li, X., Sacchi, M. D., and Ulrych, T. J. (1996). Wavelet transform inversion with prior scale information. *Geophysics*, 61(05):1379–1385.

Mallat, S. G. (1998). *A Wavelet Tour of Signal Processing.* Academic Press.

Meyer, Y. (1992). *Wavelets and Operators.* Cambridge University Press.

Morozov, V. A. (1993). *Regularization Methods for Ill-Posed Problems.* CRC Press.

Nashed, M. Z. and Wahba, G. (1974). Generalized inverses in reproducing kernel spaces: An approach to regularization of linear operator equations. *SIAM Journal of Mathematical Analysis*, 5(6):974–987.

Naylor, A. W. and Sell, G. R. (1982). *Linear Operator Theory in Engineering and Science.* Springer-Verlag.

Neelami, R., Choi, H., and Baraniuk, R. Wavelet-based deconvolution for ill-conditioned systems. Submitted to IEEE Transactions on Image Processing, February 2000.

Neelami, R., Choi, H., and Baraniuk, R. Wavelet-based deconvolution using optimally regularized inversion for ill-conditioned systems. Technical report, Deparment of Electrical and Computer Engineering, Rice University.

Neelami, R., Choi, H., and Baraniuk, R. (1999). Wavelet-domain regularized deconvolution for ill-conditioned systems. Technical report, Deparment of Electrical and Computer Engineering, Rice University.

Nowak, R. D. and Baraniuk, R. G. (1998). Wavelet-based transformations for nonlinear signal processing. Submitted to IEEE Transactions on Signal Processing, February 1997, Revised June 1998.

Nowak, R. D. and Thul, M. J. (1998?). Wavelet-vaguelette restoration in photon-limited imaging. Technical report, Michigan State University.

Papoulis, A. (1991). *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, Inc., third edition.

Parker, R. L. (1994). *Geophysical Inverse Theory.* Princeton University Press, Princeton, New Jersey.

Prenter, P. M. and Vogel, C. R. (1985). Stochastic inversion of linear first kind integral equations 1. Continuous theory and the stochastic generalized inverse. *Journal of Mathematical Analysis and Applications.*

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1995). *Numerical Recipes in C.* Cambridge University Press.

Robinson, E. A. (1954). *Predictive decomposition of time series with application to seismic exploration.* PhD thesis, Massachusetts Institute of Technology.

Robinson, E. A. and Trietel, S. (1980). *Geophysical Signal Analysis.* Prentice-Hall, Inc.

Rodi, W. (1989). *Regularization and Backus-Gilbert Estimation in Non-linear Inverse Problems: Application to Magnetotellurics and Surface Waves.* PhD thesis, The Pennsylvania State University.

Saggaf, M. M. and Robinson, E. A. (2000). A unified framework for the deconvolution of traces of non-white reflectivity. *Geophysics*, 65:1660.

Saito, N. and Beylkin, G. (1993). Multiresolution representations using the autocorrelation functions of compactly supported wavelets. *IEEE Transactions on Signal Processing*, 41(12):3584–3590.

Schneider, M. K. (2000). Krylov subspace estimation. Technical report, Massachusetts Institute of Technology. Laboratory for Information and Decision Systems.

Schoenberg, I. J. (1946). Contribution to the problem of approximation of equidistant data by analytic functions. *Quart. Appl. Math.*, 4:45–99.

Sengbush, R. L., Lawrence, P. L., and McDonal, F. J. (1960). Interpretation of synthetic seismograms. *Geophysics*, 26:138–157.

Smith, M. J. T. and Barnwell, T. P. (1986). Exact reconstruction techniques for tree-structured subband coders. *IEEE Trans. ASSP.*

Starck, J., Candès, E. J., and Donoho, D. L. (2000). The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, (11):670–684.

Strang, G. (1986). *Introduction to Applied Mathematics.* Wellesley-Cambridge Press.

Strang, G. and Nguyen, T. (1997). *Wavelets and Filter Banks.* Wellesley-Cambridge Press.

Tarantola, A. (1987). *Inverse Problem Theory: Methods for Data Fitting and Model Parameter Estimation.* Elsevier.

Thévanaz, P., Blu, T., and Unser, M. (2000). Interpolation revisited. *IEEE Transactions on Medical Imaging.*

Tikhonov, A. N. and Arsenin, V. Y. (1977). *Solutions of Ill-Posed problems.* V. H. Winston and Sons, Washington, D.C.

Trefethen, L. N. and Bau, D. (1997). *Numerical Linear Algebra.* Society of Industrial and Applied Mathematics.

Unser, M. (1999). Splines: A perfect fit for signal and image processing. *IEEE Signal Processing Magazine.*

Unser, M. and Blu, T. (1999). Construction of fractional spline wavelet bases. Technical report, Biomedical Imaging Group, Swiss Federal Institute of Technology, Lausanne.

Unser, M. and Blu, T. (2000). Fractional splines and wavelets. *SIAM Review*, 42(1):43–67.

Vogel, C. R. (2002). *Computational Methods for Inverse Problems.* SIAM.

Wiener, N. (1949). *Extrapolation, Interpolation, and Smoothing of Stationary Time Series.* Wiley, New York.

Willsky, A. S., Wornell, G. W., and Shapiro, J. H. (1999). *Stochastic Processes, Detection and Estimation: 6.432 Course notes.* MIT.

Xu, W., Tran, T. T., Srivastava, R. M., and Journel, A. G. (1992). Integrating seismic data in reservoir modeling: the collocated cokriging alternative. In *67th Annual Technical Conference and Exhibition.* Society of Petroleum Engineers.

Zemanian, A. H. (1987). *Distribution Theory and Transform Analysis.* Dover Publications, Inc.