

Inversion of the Linear and Parabolic Radon Transform

Martin Sarajærví

MASTER'S THESIS IN
PETROLEUM GEOSCIENCE



DEPARTMENT OF EARTH SCIENCE
UNIVERSITY OF BERGEN
NORWAY

MAY 31, 2010

Introduction

In this thesis, different mathematical techniques are applied to the linear and parabolic Radon transforms. It includes a review of classical work, recent developments, and testing of two proposed methods. Studies span from sampling and numerical stability to kinematic considerations, and computational efficiency.

The thesis is organized into six chapters. In Chapter 1, the Radon transform is introduced by a brief example. It is followed by a definition of the geophysical adaptation, known as the linear and parabolic Radon transform. Analytic results, in form of sampling criterions and frequency domain derivations, provide the mathematical background. An introduction to the discrete transform is given, this defines the forward and inverse transforms. For comparison, a direct implementation of the time-variant (hyperbolic) transform, including a discussion on interpolation, is given in the last section.

In Chapter 2, the linear and parabolic Radon transforms are studied from a geometrical point of view. Curves and their envelopes are studied using fundamental principles from differential geometry. This gives some insight in what to expect from a ‘perfect’ solution (Radon panel).

Chapter 3 introduce theory on inverse and ill-conditioned problems. The least squares method and different regularization techniques are studied from the viewpoint of mathematical optimization (by differential operators) and singular value decomposition. Minimizing the least squares objective function of a complex variable, is used as a motivation for a theoretical discussion on the complex Newton method, complex (\mathbb{CR}) differential operators and eigenvalue decomposition.

Chapter 4 focuses on the structure of Toeplitz operators and inversion methods. This includes techniques such as Cholesky factorization, the Levinson algorithm and the preconditioned (bi)conjugate gradient method, efficiently calculated by the fast Fourier transform. Fundamental theory concerning Toeplitz matrices and its relation to Fourier series is briefly introduced, with emphasis on eigenvalues and generating functions. The chapter concludes with numerical experiments, using the different techniques.

Chapter 5 studies a low-pass filter with an impulse response in form of the characteristic Dirichlet kernel. This impulse response generates the columns of the Toeplitz matrix, subject to inversion in the forward Radon transform. A circular approximation of the filter, and fixed point iteration, is considered as an alternative approach with computational advantages by the fast Fourier transform.

In Chapter 6, sparse approximations, also known as high-resolution Radon transforms, are investigated. The theory is motivated by the desire to discriminate fast, short-period multiples under limited offset situations. An industry standard high-resolution method is presented, reviewing the original paper, using a Bayesian framework. A similar approach is then tested. That is, solving the basis pursuit denoise problem, where the troublesome one-norm is handled by a modern

optimization algorithm. The latter method is tested on both synthetic and real datasets, including a multiple removal scheme.

Appendix A includes a complementary description of the geophysical parameters and frequency domain derivations for the linear and parabolic Radon transform.

To facilitate the use of algorithms, discussed in Chapter 4 and 5, Appendix B includes MATLAB programs for the numerical results.

ACKNOWLEDGMENTS

I first of all want to thank my advisor, Professor Einar Mæland, for introducing me to the field of seismic data processing. His suggestions on topics and learning strategy have been most valuable.

Further, I would like to thank the Signal Analysis and Imaging Group, University of Alberta, for making their seismic data processing (SeismicLab) package available under the GNU General Public License. A useful toolkit for seismic plotting and basic processing routines.

I also want to thank everybody else at the Department of Earth Science who have in some way contributed to this thesis.

Martin Sarajærví

Contents

Introduction	i
Chapter 1. The Radon Transform	1
1. Time-invariant Radon Transform	2
2. Discrete Frequency Formulation	4
3. Sampling and Aliasing	5
4. Time-variant Radon Transform and Interpolation	9
Chapter 2. Envelopes and Focusing Aspects	15
1. Envelopes and the Linear Radon Transform	16
2. Envelopes and the Parabolic Radon Transform	18
3. Curvature and Focusing	20
Chapter 3. Least Squares and Regularization Techniques	23
1. Least Squares and ℓ_p -norm Regularization	24
2. Complex Roots and Newton's Method	27
3. Complex Differential Operators Using CR-Calculus	30
4. Diagonalization of the Complex Newton Update	33
Chapter 4. Direct and Iterative Inversion Methods	37
1. Toeplitz Structure	38
2. Quadratic Forms and the Conjugate Gradient Method	41
3. Preconditioned Conjugate Gradient	45
4. Circulant Preconditioning of Toeplitz Matrices	46
5. Matrix-vector Multiplication by the Fast Fourier Transform	47
6. Numerical Results	49
Chapter 5. Analysis of the Low-pass Filter	59
1. Zero-phase Low-pass Filter	60
2. Sampling and Aliasing	62
3. Optimal Circulant Approximation	64
4. Circulant Approximation and Fixed Point Iteration	67
5. Numerical results	68
Chapter 6. Sparse Approximations - High-Resolution Methods	73
1. The Bayesian Approach	73
2. The Principle of Maximum Entropy	74
3. Maximum a Posteriori Solution	75
4. One-norm Regularization and Basis Pursuit Solutions	78
5. Basis Pursuit Denoise and the Radon Transform	79

6. Recent Developments and Greedy Methods	87
Chapter 7. Summary and Conclusions	89
Appendix A. Alternative Frequency Domain Formulations	91
1. Linear Radon in the FK-domain	92
2. Parabolic Radon and the Fresnel transform	93
Appendix B. Algorithms, Code Examples and Parameters	95
1. Algorithms	95
2. MATLAB Code	96
Bibliography	103

CHAPTER 1

The Radon Transform

The Radon transform is an integral transform, introduced by the mathematician Johan Radon in 1917. It is defined by assuming a function $f(x, y)$, illuminated by parallel rays. An image or observation $r(s, \theta)$ is constructed in the following manner

$$r(s, \theta) = \iint_{-\infty}^{\infty} f(x, y) \delta(s - x \cos \theta - y \sin \theta) dx dy,$$

where s is the distance from the origin with θ as a reference direction for the rays. Figure 1 illustrates the geometry. Explicit and efficient inversion formulas for the Radon transform are available in literature on tomography (Deans, 1983; Jain, 1989; Ramm and Katsevich, 1996; Helgason, 1999). In Figure 2, a square box is illuminated and reconstructed, analogous to the above definition.

Originally discovered without consideration to any practical problems, the Radon transform finds applications with the development of modern technology. The most common application is the process of reconstructing images in the field of medical tomography; the inverse Radon transform is applied to data obtained from computed tomography (CT) scans. This led to a Nobel-prize in 1979 for the groundbreaking concept of viewing organs from outside the body.

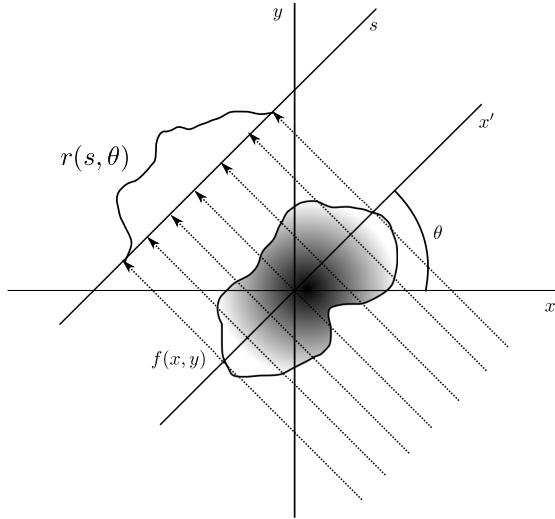


Figure 1: Illustrative example, geometry of the Radon transform.

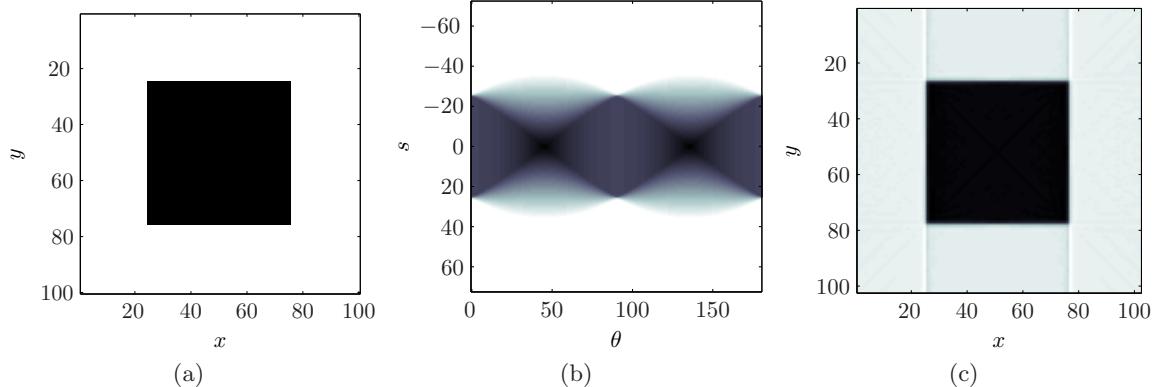


Figure 2: The Radon transform using built-in MATLAB functions. a) The original image $f(x, y)$. b) The Radon transform of the original image. Angles from $0 \leq \theta < 180^\circ$ with $\Delta\theta = 1^\circ$. c) Reconstruction of the original image $f(x, y)$, without filtering.

The present work focuses on modifications of this classical formulation, versions suitable for geophysical problems. These modifications will be referred to as the linear Radon ($\tau-p$) transform and the parabolic Radon ($\tau-q$) transform.

In this chapter, the transforms are defined in both the frequency and time domain, including a review of sampling criterions. The $\tau-p$ transform is discretized, representing a unified theory for both transforms. This makes up a framework for analysis following in subsequent chapters. In these chapters, inverse theory is presented in a general manner.

1. Time-invariant Radon Transform

The time-invariant $\tau-p$ transform is defined in time-offset as

$$(1) \quad r(p, \tau) = \int_{-\infty}^{\infty} f(x, t = \tau + px) dx,$$

where the data $f(x, t)$, a function of offset and time, is summed along the straight line $t = \tau + px$, parametrized by the ray parameter p and intercept time τ . After filtering of $r(p, \tau)$, the seismic data $f(x, t)$ is reconstructed. The natural inverse operator can be written

$$(2) \quad f(x, t) = \int_{-\infty}^{\infty} r(p, \tau = t - px) dp.$$

It should be noted that equations (1) and (2) are correct from a kinematical point of view only (Maeland, 1998). From an amplitude point of view, a correction term is required.

The Radon transform can be generalized to curves other than straight lines

$$t = \tau + \varphi(x).$$

Suitable choices for $\varphi(x)$ could focus certain seismic events in the transform domain. This domain may have parameters representing geophysical properties like curvature (residual moveout) q or seismic velocity v . The parabolic formulation, summing along trajectories with $\varphi(x) = qx^2$, also yield a time-invariant transform. Figure 3 illustrates the geometrical principle with linear and parabolic trajectories. These transforms serve a purpose in seismic data processing (Zhou and Greenhalgh, 1994), where they may be used in a range of methods: Velocity analysis, inversion (migration and

modeling), interpretation, plane-wave decomposition, wave separation, noise attenuation and data interpolation. Details on the geophysical properties and geometry of the parameters are included in Appendix A.

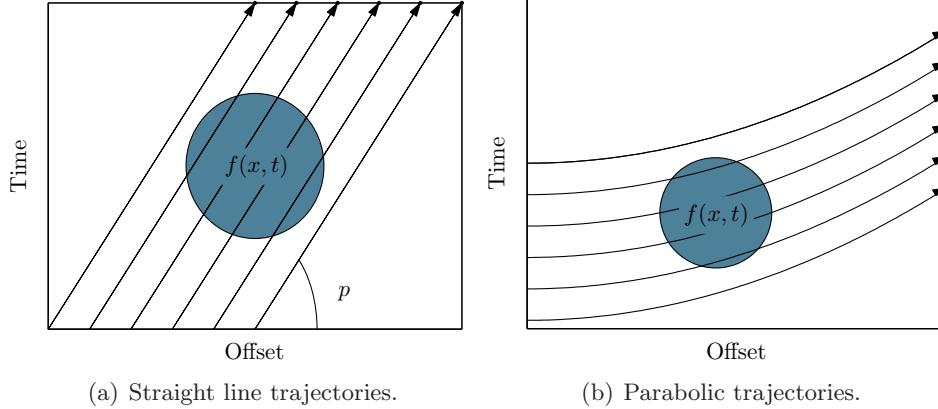


Figure 3: Geometrical principle of the time-invariant transforms. a) The tau- p transform. b) The tau- q transform.

1.1. Transforms in the frequency domain. An invariant is a property of a system which remains unchanged under some transformation. The advantage of the tau- p and tau- q transforms over e.g. the time-variant hyperbolic transform, lies in the fact that it is possible to take the Fourier transform of both sides of the equations and solve them independently for each frequency component (Hampson, 1986; Kostov, 1990). This reduces the computational cost considerably. For convenience, only the tau- p transform is evaluated¹ in the following derivations. Let $f(x, \omega)$ and $r(p, \omega)$ denote the temporal Fourier transform of respectively $f(x, t)$ and $r(p, \tau)$, such that

$$(3) \quad r(p, \omega) = \iint_{-\infty}^{\infty} f(x, \tau + px) e^{-i\omega\tau} dx d\tau.$$

The substitution $\tau' = \tau + px$, yields

$$(4) \quad r(p, \omega) = \iint_{-\infty}^{\infty} f(x, \tau') e^{-i\omega(\tau' - px)} dx d\tau' = \int_{-\infty}^{\infty} f(x, \omega) e^{i\omega px} dx.$$

The time-shift is expressed as a multiplication by a complex number. Different frequency components of the data can be transformed to the (p, ω) -domain independently from each other. An inverse frequency formulation is then

$$(5) \quad f(x, \omega) = \int_{-\infty}^{\infty} r(p, \omega) e^{-i\omega px} dp.$$

Alternative frequency domain formulations using the two-dimensional Fourier and Fresnel transforms, for respectively tau- p and tau- q , are included in Appendix A.

¹Results are analogous with the parabolic Radon transform.

2. Discrete Frequency Formulation

A frequency domain formulation of the time-invariant discrete Radon transform (Thorson, 1984; Beylkin, 1987) is presented in this section. It is the basis of what will be discussed in the subsequent chapters.

2.1. Forward transform, $(x, t) \rightarrow (p, \tau)$. Evaluating the linear Radon formulation, let $p_m = m\Delta p$ and $x_n = n\Delta x$, such that a discrete forward transform is given as

$$(6) \quad r(p_m, \omega) \approx \sum_{n=1}^N f(x_n, \omega) \exp(i\omega x_n p_m),$$

for $m = 1, 2, \dots, M$. The approximation is used to emphasize the fact that this is not an exact solution, and that in general, $M \neq N$.

2.2. Inverse transform, $(p, \tau) \rightarrow (x, t)$. An inverse transform is given by

$$(7) \quad f(x_n, \omega) \approx \sum_{m=1}^M r(p_m, \omega) \exp(-i\omega x_n p_m),$$

for $1, 2, \dots, N$. Let $A_{nm} = \exp(-i\omega x_n p_m)$, $f_n = f(x_n, \omega)$ and $r_m = r(p_m, \omega)$, then the above equation can be written

$$\begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{pmatrix} \approx \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1M} \\ A_{21} & A_{22} & \cdots & A_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N1} & A_{N2} & \cdots & A_{NM} \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_M \end{pmatrix}.$$

When $M < N$, the set of equations will be overdetermined.

2.3. Least squares forward transform. It is common practice to replace the forward transform with a least squares forward transform. This gives better resolution in the Radon domain, which allows better signal and noise separation (Schonewille and Duijndam, 2001).

Multiplying f_n by $A_{nj}^* = \exp(i\omega x_n p_j)$ where $p_j = j\Delta p$ with $j = 1, 2, \dots, M$, and summing over n , yields an approximate solution \tilde{r}_j

$$\tilde{r}_j = \sum_{n=1}^N A_{nj}^* f_n \approx \sum_{n=1}^N \exp(i\omega x_n p_j) \sum_{m=1}^M r_m \exp(-i\omega x_n p_m).$$

Further, let \hat{r}_m be the least squares (generalized inverse) solution

$$\begin{aligned} \tilde{r}_j &= \sum_{n=1}^N \sum_{m=1}^M \exp[-i\omega(p_m - p_j)x_n] \hat{r}_m \\ &= \sum_{m=1}^M \hat{r}_m \sum_{n=1}^N \exp[-i\omega(m-j)\Delta p x_n] \\ &= \sum_{m=1}^M \hat{r}_m \sum_{n=1}^N A_{n(m-j)}. \end{aligned}$$

The approximate solution \tilde{r}_j , can be written as a convolution²

$$(8) \quad \tilde{r}_j = \sum_{n=1}^N A_{nj}^* f_n = \sum_{n=1}^N f_n \exp(i\omega x_n p_j) = \sum_{m=1}^M \hat{r}_m h_{j-m},$$

where h_m is a low-pass filter, subject to investigation in Chapter 5. Formulating the discrete Radon transform in this manner has both advantages and disadvantages. It obviously could give a detailed insight to the problem, but by using a more general vector-matrix notation, techniques from linear algebra can be denoted and utilized more easily.

2.4. Notation using linear algebra. The discrete analysis gives a linear transform for each frequency. For the remainder of the present work, a linear system (without consideration to the variable names introduced in the discretization) is denoted

$$\mathbf{A}\mathbf{z} = \mathbf{b},$$

where vector \mathbf{z} represents the Radon panel, vector \mathbf{b} represents the data, and the operator \mathbf{A} represents a linear mapping. An operator \mathbf{A}^H denotes the Hermitian transpose (conjugate transpose) and it is important to notice that \mathbf{A} is generally not an unitary operator, i.e. it does not define the inverse. For the Radon transform, \mathbf{A}^H represents a low-resolution forward Radon operator.

Let $\mathcal{R}(\mathbf{A})$ denote the range (column space) of a $N \times M$ matrix \mathbf{A} . For a solution of the linear system to exist, $\mathbf{b} \in \mathcal{R}(\mathbf{A})$ has to be satisfied with \mathbf{b} as

$$\mathbf{b} = \mathbf{A}\mathbf{z} + \mathbf{u},$$

where $\mathbf{A}\mathbf{z}$ is the projection of \mathbf{b} onto $\mathcal{R}(\mathbf{A})$ and \mathbf{u} is the residual vector, orthogonal to $\mathcal{R}(\mathbf{A})$ (Maeland, 1998). A least squares solution is calculated by the normal equations

$$\tilde{\mathbf{z}} = \mathbf{A}^H \mathbf{b} = \mathbf{A}^H \mathbf{A} \hat{\mathbf{z}}.$$

Finding the least squares solution $\hat{\mathbf{z}}$ requires the matrix (Hessian) $\mathbf{A}^H \mathbf{A}$ to be nonsingular. This is generally a difficult problem and is subject of further investigation in Chapter 3.

3. Sampling and Aliasing

Calculating the Radon transform is a computational expensive process, especially with large datasets. Therefore, mathematical operations should be minimized by limiting the number of samples. This section gives a summary of sampling criterions for both the linear and parabolic transforms. It is important to note that aliasing in the Radon panel is diffused. Sampling criterions are defined for the highest frequency in the signal, hence, all the lower frequencies will be oversampled since a uniform sampling scheme is required. Therefore, datasets with a single monochromatic signal often used for illustrations.

3.1. Sampling and aliasing in tau-p. For a basic understanding of sampling and reconstruction of the linear Radon transform, a harmonic monochromatic signal is evaluated. It can be used to demonstrate how aliasing manifests itself, and how critical sampling intervals are defined. Suppose a function $f(x, t)$, described by a delta-function

$$f(x, t) = \delta(x - x_0) e^{i\omega_0(t - t_0)}$$

²It is important to notice that this convolution cannot be classified as neither linear nor circular.

and its two-dimensional Fourier transform

$$\begin{aligned}
\mathcal{F}\{f(x, t)\} &= \int_{-\infty}^{\infty} \delta(x - x_0) e^{ixk_x} dx \int_{-\infty}^{\infty} e^{i\omega_0(t-t_0)} e^{-i\omega t} dt \\
&= e^{ix_0 k_x} \int_{-\infty}^{\infty} e^{i\omega_0(t-t_0)} e^{-i\omega t} dt \\
&= e^{ix_0 k_x} e^{-i\omega_0 t_0} \int_{-\infty}^{\infty} e^{i\omega_0 t} e^{-i\omega t} dt \\
&= 2\pi \delta(\omega - \omega_0) e^{i(x_0 k_x - \omega_0 t_0)}.
\end{aligned}$$

Substituting $k_x = p\omega$ gives

$$q(p\omega, \omega) = 2\pi \delta(\omega - \omega_0) e^{i(x_0 p\omega - \omega_0 t_0)},$$

and applying the inverse temporal Fourier transform

$$r(p, \tau) = \mathcal{F}^{-1}\{q(p\omega, \omega)\} = \frac{1}{2\pi} \int_{-\infty}^{\infty} 2\pi \delta(\omega - \omega_0) e^{i(x_0 p\omega - \omega_0 t_0)} e^{i\omega\tau} d\omega.$$

The function $\delta(\omega - \omega_0)$ is zero for all ω except when $\omega = \omega_0$, such that

$$r(p, \tau) = e^{i\omega_0(x_0 p - t_0 + \tau)} = r(p + m \frac{2\pi}{\omega_0 x_0}, \tau + n \frac{2\pi}{\omega_0}),$$

where $m, n \in \mathbb{Z}$. This gives a periodicity in x

$$\tilde{x}_0 = x_0 + \frac{2\pi m}{\omega_0 p} \implies X = \frac{2\pi}{\omega_0 \Delta p} = \frac{1}{\nu_0 \Delta p},$$

where ν_0 is the linear frequency and X is the period. Finally, a criterion for sampling and reconstruction in tau-p,

$$\Delta p \leq \frac{\pi}{\omega_0 x_0} \quad \text{and} \quad \Delta\tau \leq \frac{\pi}{\omega_0}.$$

In Turner (1990), a more generalized version of tau-p sampling is given. The paper is based on a geometrical principle of constructive summing along slants. The criterion is given as

$$(9) \quad \Delta p < \frac{T}{x_r} = \frac{1}{x_r \nu_{\max}},$$

where x_r is the range of x values, T is the period of the signal and ν_{\max} is the maximum linear frequency. A similar criterion is given for spatial sampling

$$(10) \quad \Delta x \leq \frac{1}{p_r \nu_{\max}} = \frac{1}{2p_{\max} \nu_{\max}} = \frac{\pi}{k_{\max}},$$

where p_r is the range of p values. Avoiding undersampling by the criterions described above will in theory give perfect reconstruction. However, the fact that finite regions are used, smearing effects according to some resolution matrix will be introduced (Maeland, 2004).

Following the first theoretical derivation in this section, a simple example is given in Figure 4. A cosine is sampled at $\nu_0 = 20$ Hz at $x_0 = 0.5$ km. Using $\Delta x = 25$ m with $\Delta p = 1/50$ s/km on Figure 4(b) and $\Delta p = 1/8$ s/km on Figure 4(c). The latter demonstrates aliased events at $x = 0.1$ km and $x = 0.9$ km, when the period is $X = 2/5$ km whereas the period for critical sampling is $X = 2x_0 = 1$ km.

In a recent paper by (Dev and McMechan, 2009), a spatial antialias filter is presented. Their filter is applied to spatially coarsely sampled seismic data, removing all energy above the first

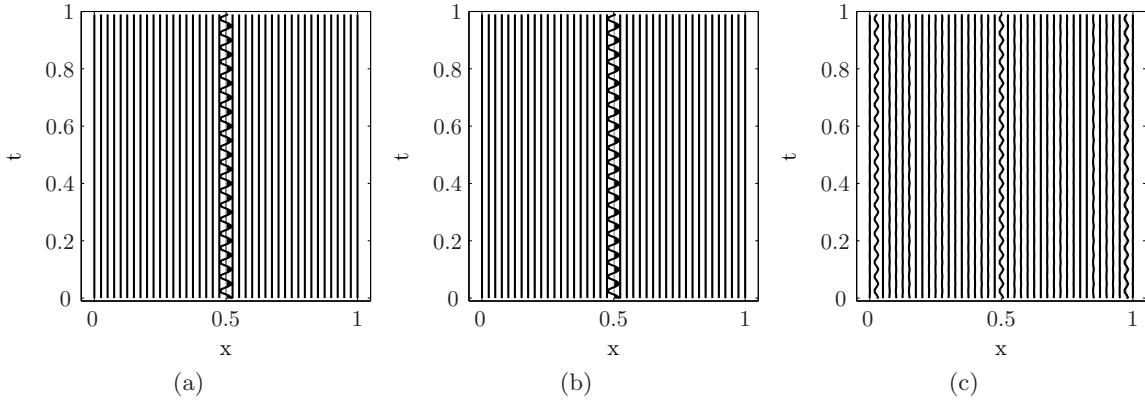


Figure 4: Demonstration of sampling and reconstruction in p-tau. a) Original signal. b) Properly sampled reconstruction. c) Undersampled reconstruction.

Nyquist wavenumber, and aliased energy that is folded back across the Nyquist (in the slowness-frequency domain). A novel approach to the tau- p transform is given in (Averbuch et al., 2001). Their method relies on the discrete projection-slice theorem, relating this to what they call the pseudopolar Fourier transform. The theory is presented as algebraically exact, geometrically faithful and efficiently invertible.

3.2. Sampling and aliasing in tau- q . Sampling theorems for the parabolic Radon transform are not as straightforward as with the linear transform. Let $r(q, \omega)$ be the temporal Fourier transform of the Radon panel $r(q, \tau)$ and let $f(x, \omega)$ be the temporal Fourier transform of $f(x, t)$, such that

$$(11) \quad r(q, \omega) = \int_{-\infty}^{\infty} f(x, \omega) e^{i\omega q x^2} dx,$$

and

$$(12) \quad f(x, \omega) = \int_{-\infty}^{\infty} r(q, \omega) e^{-i\omega q x^2} dq.$$

A transform pair analogous to the previous analysis with tau- p .

3.2.1. Continuous convolution operator. Substituting Equation (12) into Equation (11), and defining a new variable q' , gives

$$\begin{aligned} r'(q, \omega) &= \iint_{-\infty}^{\infty} r(q', \omega) e^{i\omega(q-q')x^2} dq' dx \\ &= \int_{-\infty}^{\infty} e^{i\omega x^2(q-q')} dx \int_{-\infty}^{\infty} r(q', \omega) dq' \\ &= \int_{-\infty}^{\infty} r(q', \omega) \sigma(q - q', \omega) dq' \\ &= r(q, \omega) * \sigma(q, \omega). \end{aligned}$$

Studying the parabolic Radon transform by using the convolution/deconvolution formalism and the Fourier transform, can be used determine critical sampling intervals for q . The convolution operator $\sigma(q, \omega)$ with respect to q does not have an antiderivative expressed by an elementary function, hence

integration in the complex plane is needed. For q , the operator is given as (Zhou and Greenhalgh, 1994)

$$\sigma(q, \omega) = 2 \int_0^\infty e^{i\omega q x^2} dx = [1 + i \operatorname{sgn}(q)] \sqrt{\frac{\pi}{2\omega|q|}}, \quad q, \omega \neq 0$$

and in Fourier space, k_q ,

$$(13) \quad \sigma(k_q, \omega) = [1 + \operatorname{sgn}(k_q)] \frac{\pi}{\sqrt{\omega|k_q|}}, \quad k_q, \omega \neq 0.$$

When the range of x is finite, there is no analytical solution for the integral

$$\sigma(q, \omega) = \int_{x_{\min}}^{x_{\max}} e^{i\omega q x^2} dx.$$

Therefore, numerical implementations of the convolution operator introduce problems. This problem makes the aliasing phenomenon diffused for the parabolic Radon transform. Using regular sampling, the operator is given as

$$(14) \quad \sigma(q, \omega) = \Delta x \sum_n e^{i\omega q(n\Delta x)^2}.$$

3.2.2. Sampling criterions. In general, formulations where integrals are simply approximated by a sum introduce vast amounts of arbitrariness and are too imprecise to constitute a definition of a Radon transform for digital data; they should be avoided (Averbuch et al., 2001). However, following (Hugonnet and Canadas, 1995; Apeland, 2001), discretization in x , the convolution operator $\sigma(q, \omega)$ becomes periodic in q with a period

$$Q = \frac{2\pi}{\omega(\Delta x)^2}.$$

This is not a problem with implementations as very high q -values are required.

Hugonnet and Canadas (1995) introduced the terminology: Local summations in phase. That is, secondary lobes at

$$(15) \quad \tilde{q} = \pm \frac{1}{2\nu_{\max}|x_{\max}|\Delta x},$$

are identified in implementations of $\sigma(q, \omega)$. It is clear that for smaller Δx , sidelobes will appear at higher q -values. High frequencies will make the sidelobes appear at lower q -values. These sidelobes represent an unwanted build-up of energy. Applying the sampling criterions from the linear transform given in Equation (10), with the substitution $X = x^2$, such that $\Delta X = 2x\Delta x$. A new criterion is derived as

$$(16) \quad \Delta x \leq \frac{1}{2\nu_{\max}|x|_{\max}(q_{\max} - q_{\min})}.$$

From Equation (13), Hugonnet and Canadas (1995) derives a critical sampling interval in q ,

$$(17) \quad \Delta q_c = \frac{1}{\nu_{\max}\hat{x}},$$

where $\hat{x} = (x^2)_{\max} - (x^2)_{\min}$. In a former master's thesis, Apeland (2001) study the above criterions, including how sampling affects stability of the least squares operator.

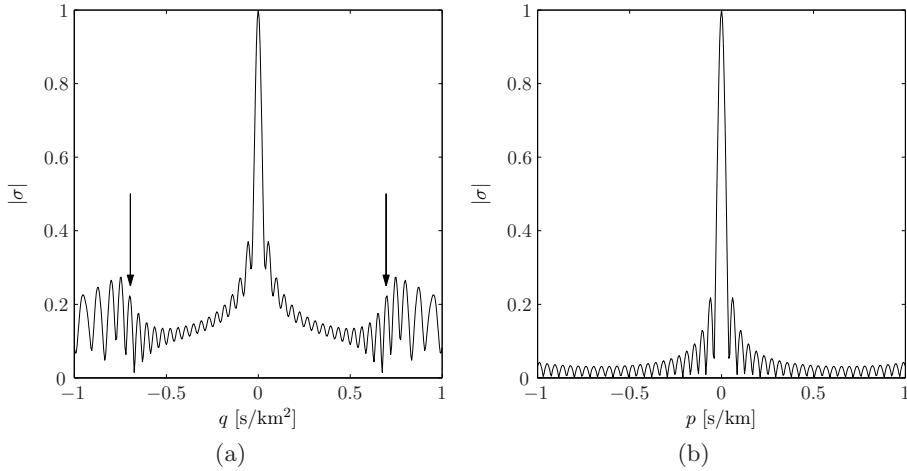


Figure 5: The convolution operator at frequency $\nu = 23$ Hz. a) The parabolic version, demonstrating sidelobes from undersampling x . b) The linear version, which does not suffer from the sidelobes.

Figure 5 shows Equation (14), evaluated with parameters chosen as $\nu = 23$ Hz, $x_{\max} = 1$, $x_{\min} = 0$, $\Delta x = 1/32$ km and $\Delta q = 1/200$ s/km². Arrows indicate where sidelobes start at $\tilde{q} = \pm 1/2\nu|x|_{\max}\Delta x$. A version for the linear transform is included using the same sampling intervals, demonstrating how this version does not suffer from the same energy build-up.

4. Time-variant Radon Transform and Interpolation

There are advantages and disadvantages of time-variant Radon transforms like the hyperbolic or elliptic versus the linear or parabolic transforms. Several different considerations need to be taken into account; computational speed, focusing ability, stability and robustness. In seismic data processing, summing along the paths which most closely approximate the actual traveltime curve will in theory yield the best focusing of seismic events. For a shot gather with a source at offset $x = 0$, the travelttime at zero offset from the base of the n th layer is

$$t_n(0) = 2 \sum_{j=1}^n h_j/v_j,$$

where h_j is the horizontal thickness and v_j is the interval velocity of each layer (Maeland, 1998). For small offsets, the travelttime curve can be approximated by the hyperbola

$$t_n^2(x) = t_n^2(0) + (x/\bar{v}_n)^2 + \dots \approx \tau_n^2 + (x/\bar{v}_n)^2,$$

where the rms velocity \bar{v}_n satisfies

$$\bar{v}_n^2 t_n(0) = 2 \sum_{j=1}^n h_j v_j.$$

Based on these assumptions, defining the parameters v and τ as respectively velocity and intercept time, the hyperbolic Radon transform (Thorson and Claerbout, 1985; Yilmaz, 1988), is given as

$$r(v, \tau) = \int_{-\infty}^{\infty} f(x, t = \sqrt{\tau^2 + (x/v)^2}) dx.$$

Discretization and implementation of this integral is straightforward in the time-offset domain, but at the expense of heavy computational burden. It involves an interpolation scheme as the hyperbolas does not match the existing grid where the seismic data are sampled. Using a time-offset implementation, any curve could be used, e.g. non-linear and more complicated functions which could include anisotropy effects. Other advantages of this formulation is the possibility to have non-uniform sampling for velocity, using dense sampling where events are expected, and less dense otherwise. Summing along hyperbolas and limiting the offset-range is another possibility, a parameter choice which could be used to e.g. accommodate for amplitude versus offset effects.

For seismic data with large moveout, the hyperbolic Radon transform generally yields better focus of different events. However, on NMO corrected gathers, the parabolic Radon transform with the parameter q as residual moveout, manages to focus the NMO corrected events closely to zero. This makes it easier to separate them from the multiples.

4.1. Interpolation. In a time-offset implementation of the Radon transform, the curves on which the summation is carried out, defines new sampling points. These points regularly do not match the existing grid in the vertical direction (time). Interpolation or extrapolation is used to deal with this problem.

For the purpose of illustration, Figure 6(a) shows a matrix generated by translating and scaling Gaussian distributions. On top of this grid, three straight lines are plotted with crosses indicating their sampling points. For one column of the matrix, let the function $f_j = f(t_j)$, with $j \in \mathbb{N}$, describe the signal. Using a constant sampling interval, $\Delta t = t_{j+1} - t_j$, it is possible to estimate the value of $f(t)$ in a point where $t_{j-1} < t < t_j$. This could be done by linear interpolation

$$f(t) = \frac{(t_{j+1} - t)f_j + (t - t_j)f_{j+1}}{\Delta t}.$$

This is a simple and efficient formulation, often sufficient for obtaining good results, however more sophisticated methods are easily available. Using the sampling theorem (Folland, 2009), and assuming a bandlimited function $f(t)$, interpolation can be written as a convolution

$$f(t) = \sum_{-\infty}^{\infty} f_j \operatorname{sinc}[\pi(t - t_j)/\Delta t].$$

In digital signal processing, the exact sinc interpolation kernel is replaced by the Dirichlet kernel from a finite number of samples (Yaroslavsky, 1997). This method is most commonly implemented by zero padding the frequency domain before applying the inverse discrete Fourier transform.

Using the sample system from Figure 6(a), the 9th column of the matrix is shown in Figure 6(b). Existing values f_j are plotted as black circles while new points, not matching the existing grid, are plotted as red and blue crosses at their respective sinc-interpolated values. For this specific example, the sinc interpolation method seems to adapt well to the data, however, it does suffer from oscillating effects which sometimes may distort data severely.

In Figures 7 and 8, time-offset implementations of the hyperbolic and parabolic Radon transforms are compared to the standard least squares parabolic Radon transform. A CMP gather (courtesy of the signal analysis and imaging group, University of Alberta) with a long period multiple of $v = 1500$ m/s, is plotted in Figure 7(a). The data is sampled with $\Delta t = 1/250$ s, $\Delta x = 50$ m, $\nu_{\min} = 1$ Hz, $\nu_{\max} = 90$ Hz and $M = 40$ sampling points for the residual moveout q and velocity v . For comparison, the same transforms are applied to the NMO corrected gather in Figure 8(a). In

the time-offset implementations, linear interpolation is used when needed, although, no significant difference is observed when using e.g. nearest-neighbor or sinc interpolation. This exercise shows how the different transforms and implementations focus events, both from a NMO corrected gather and a regular shot. For the parabolic transform, the focusing power is investigated in Chapter 2.

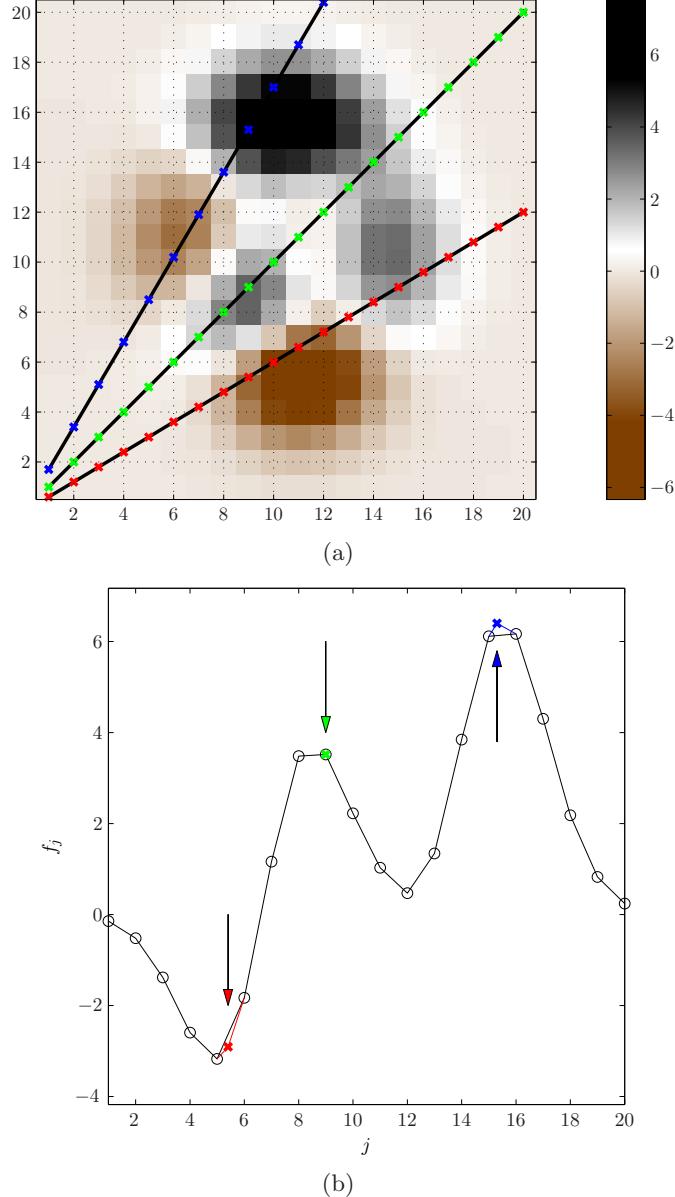


Figure 6: The need for interpolation with time-offset implementations of the Radon transform. a) An image generated by translating and scaling Gaussian distributions, also showing three straight lines, defining new sampling points. b) A selection of the 9th column with sinc (Dirichlet) interpolated values indicated with red and blue points.

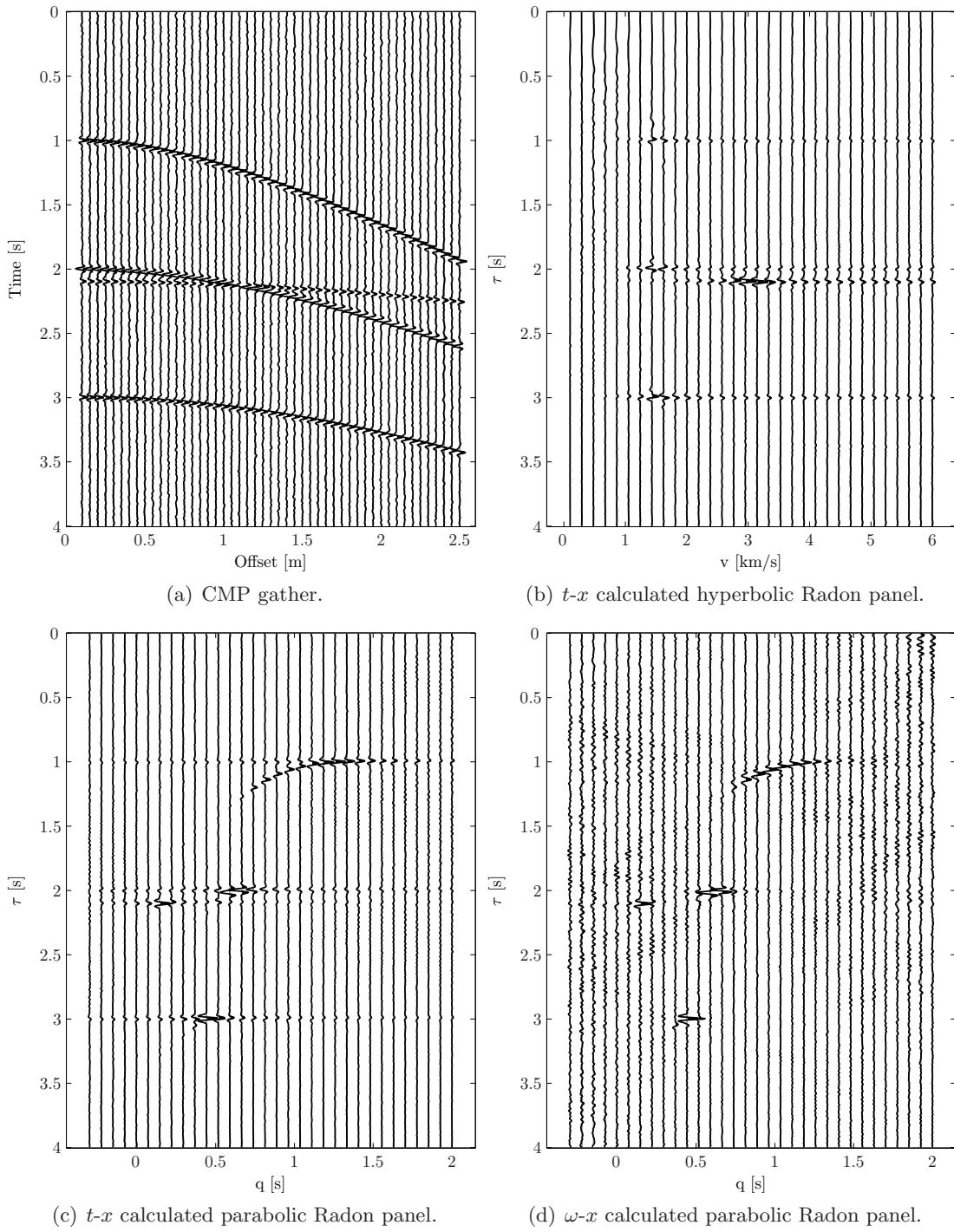


Figure 7: Radon panels calculated by different methods, plotted at the same amplitude scale for comparison.

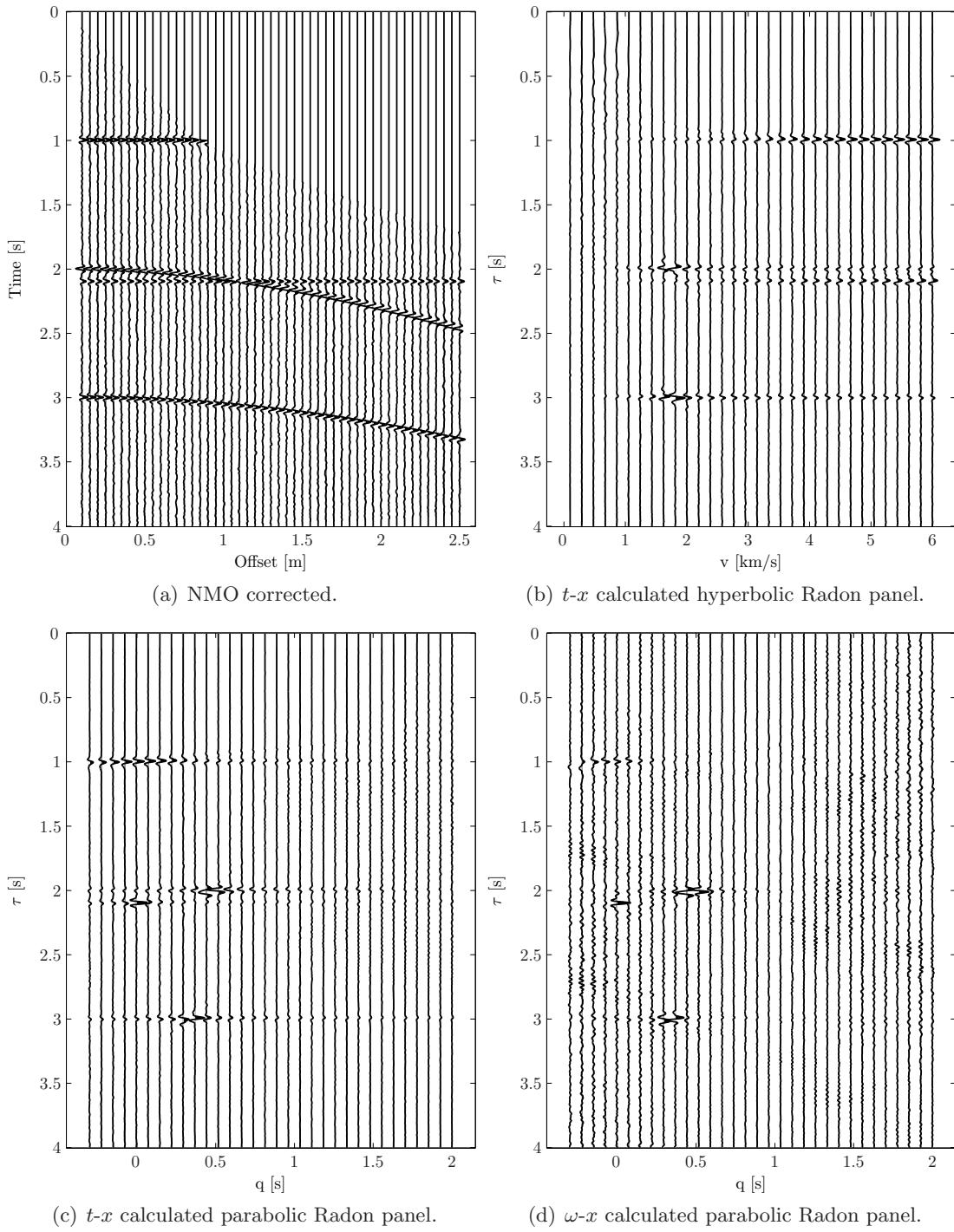


Figure 8: Radon panels calculated by different methods, plotted at the same amplitude scale for comparison.

CHAPTER 2

Envelopes and Focusing Aspects

An envelope is a curve \mathcal{E} , made up from a family of curves \mathcal{L} . A typical example could be the formation of an astroid curve by a family of ellipses, illustrated in Figure 1. Envelopes are commonly encountered in seismic imaging; seismic migration, modeling, wave refraction and diffraction could be described by Huygens principle and envelopes. Huygens principle asserts that each point on an advancing wavefront is an elementary wave, originating from a fixed source. This set of elementary waves then form a propagating wavefront.

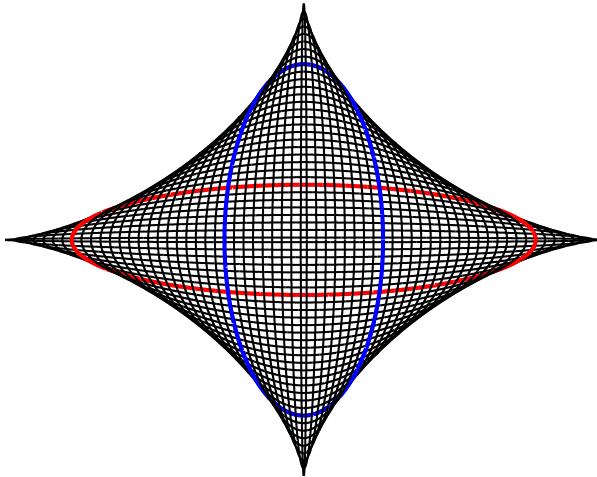


Figure 1: An astroid curve formed by a family of ellipses. Two curves are selected in red and blue.

Kinematics are the study of movement. In this setting, the terminology is used to describe how geometrical shapes change between transform domains. It is useful to do the basic analysis without the dynamics included, i.e. investigating strictly geometrical effects.

Suppose an impulse in (x_0, t_0) , given as

$$f(x, t) = \delta(x - x_0) \delta(t - t_0).$$

The linear Radon transform of $f(x, t)$ is then, formally

$$r(p, \tau) = \int_{-\infty}^{\infty} \delta(x - x_0) \delta(\tau + px - t_0) dx = \delta(\tau + px_0 - t_0),$$

a straight line on the form $\tau = t_0 - px_0$. A more thorough investigation is required to understand how the linear Radon transform responds to curves, and not only points. Before moving on with this analysis, a few fixed terms have to be established. A curve \mathcal{L} is defined by:

$$\mathcal{L} : \tau = t - \varphi(x).$$

An envelope \mathcal{E} is found by solving the set of equations

$$(18) \quad \begin{aligned} \Phi(p, \tau; x) &= t(x) - \varphi(x) - \tau = 0 \\ \partial\Phi(p, \tau; x)/\partial x &= dt/dx - d\varphi(x)/dx = 0. \end{aligned}$$

In the following subsections, a typical hyperbolic (seismic) event is used as a basis for the envelope derivations. Representations are derived for both tau- p and tau- q , where the ‘seismic hyperbola’ takes the shape of a new curve. To keep track of which curves correspond to each other, a calligraphic font is used to denote them.

1. Envelopes and the Linear Radon Transform

Analogous to the previous chapter, the linear transform is evaluated first. Let $\varphi(x) = px$ and find the envelope of a hyperbolic travelttime curve,

$$\mathcal{C}_1 : t^2(x) = a + bx^2 \implies dt/dx = p = bx/t,$$

where a and b are constants. Using the Equations (18), this gives

$$\begin{aligned} \Phi(p, \tau; x) &= \sqrt{a + bx^2} - px - \tau = 0 \\ \partial\Phi(p, \tau; x)/\partial x &= bx/t - p = 0, \end{aligned}$$

where the objective is to eliminate t and x . For this exercise, all algebraic manipulations are included, starting with squaring of τ

$$\tau^2 = t^2 - 2tpx + p^2x^2,$$

and elimination of t by \mathcal{C}_1 . This gives

$$(19) \quad \tau^2 = a + bx^2 - 2bx^2 + p^2x^2 = x^2(p^2 - b) + a.$$

To eliminate x , another expression is derived from \mathcal{C}_1 ,

$$p = dt/dx = bx/\sqrt{a + bx^2}.$$

Squaring and algebraic manipulation of this expression, gives

$$\begin{aligned} (bx)^2 &= p^2(a + bx) \\ -p^2a/b &= x^2(p^2 - b). \end{aligned}$$

Substituting into Equation (19), eliminates x and gives the envelope \mathcal{E}_1 , an ellipse in normal form

$$\mathcal{E}_1 : \frac{\tau^2}{a} + \frac{p^2}{b} = 1.$$

Manipulating this hyperbola with intention to let it represent a seismic event. Let $a = (2h/v)^2$ and $b = 1/v^2$, where h is the depth of the reflective boundary and v the velocity of the medium. This yields a hyperbola in the t - x domain and an ellipse in the tau- p domain.

$$(vt)^2 = x^2 + (2h)^2 \iff \frac{\tau^2}{(2h)^2} + p^2 = \frac{1}{v^2}.$$

A synthetic example is given in Figure 2. A hyperbola, representing a reflective boundary at depth $h = 0.5$ km with velocity $v = 1$ km/s. The linear Radon transform results in the expected ellipse with major and minor semi-axes, $a = 1/v^2 = 1$ and $b = (2h/v)^2 = 1$.

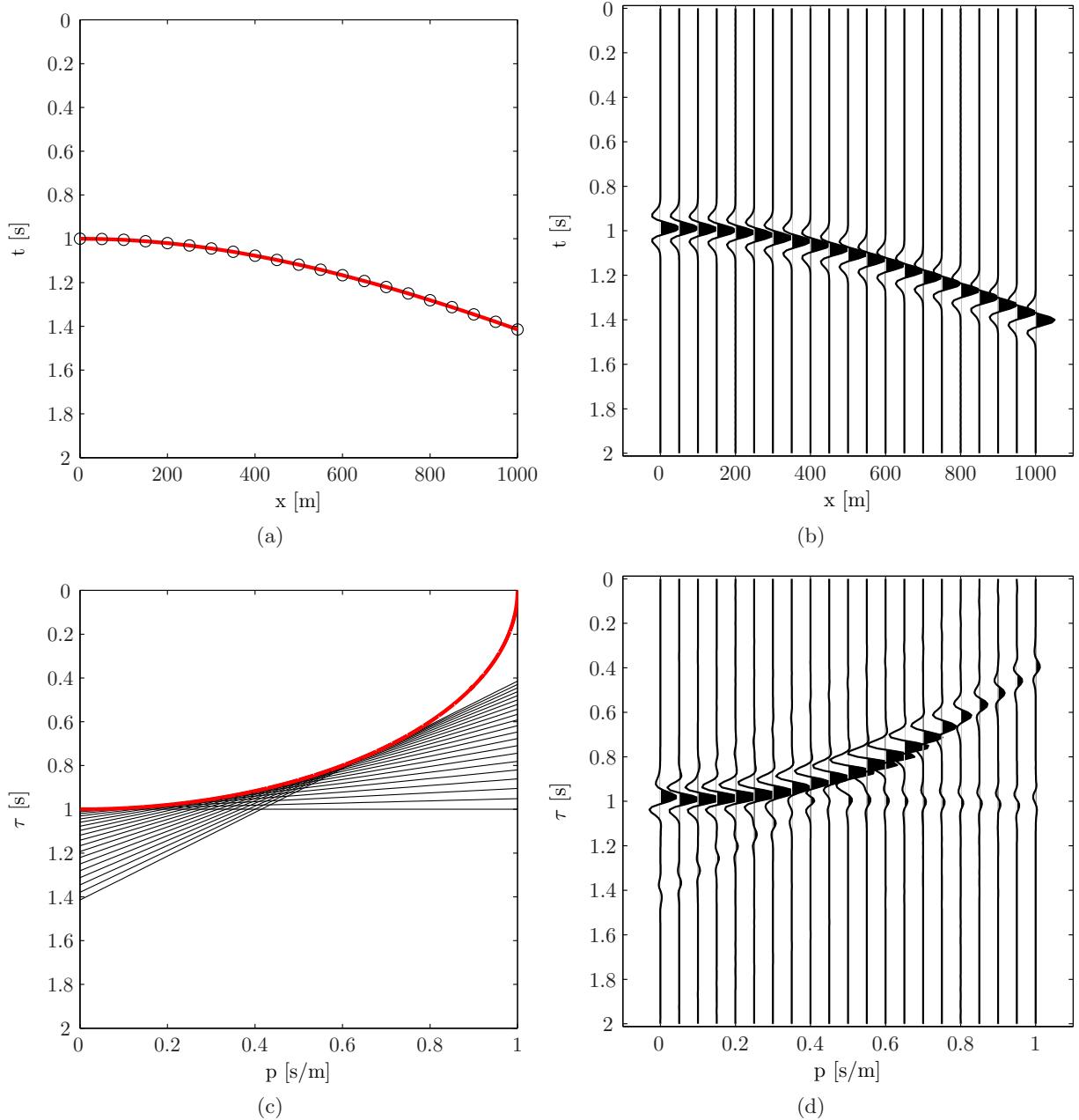


Figure 2: Kinematics and focusing of the linear Radon transform. The envelope of straight lines corresponding to points on the hyperbola. a) Kinematic plot of a hyperbolic seismic event. b) The seismic event, including amplitudes. c) Corresponding ellipse in transform domain. d) A synthetic Radon panel using a least squares implementation.

2. Envelopes and the Parabolic Radon Transform

Focusing aspects of the parabolic Radon transform (Maeland, 1998) are important to gain insight into what to expect from a perfectly sampled Radon panel of infinite aperture. With the framework already established, let $\varphi(x) = qx^2$ and assume a linear event \mathcal{C}_2 with a slope $dt/dx = \beta$, on the form

$$\mathcal{C}_2 : t(x) = \beta x.$$

A single point on \mathcal{C}_2 is represented as a line \mathcal{L} in the transform domain. This gives a trivial set of equations; eliminating t yields

$$q = \beta/2x \quad \text{and} \quad \tau = \beta x/2.$$

Eliminating x , the envelope \mathcal{E}_2 is

$$\mathcal{E}_2 : q\tau = \beta^2/4,$$

a hyperbola with asymptotes along the coordinate axes. Figure 3(a) shows a family of parabolas tangential to the curve \mathcal{C}_2 , which has a slope $\beta = 2/3$. In the corresponding transform domain, in Figure 3(b), each parabola is represented as a line, forming the envelope \mathcal{E}_2 in red.

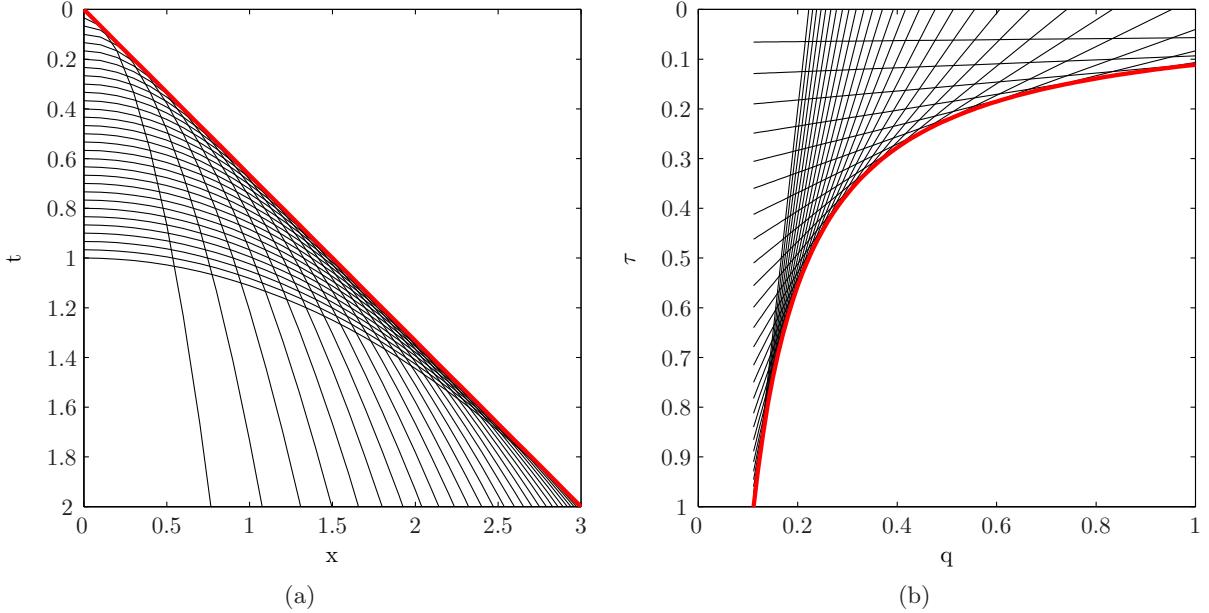


Figure 3: Kinematics of the parabolic Radon transform. A linear event is transformed to a hyperbola: $\mathcal{C}_2 \iff \mathcal{E}_2$. Every parabola, tangential to the red linear event, is represented as a line in the transform domain, forming the envelope \mathcal{E}_2 (red).

To study how well the parabolic Radon transform focus typical seismic events, further analysis are based on the hyperbolic curve \mathcal{C}_1 , which was given earlier as

$$\mathcal{C}_1 : t^2(x) = a + bx^2.$$

Using Equations (18) with $\Phi(q, \tau)$, the transform domain, is given parametrically as

$$q(x) = b/2t(x) \quad \text{and} \quad \tau(x) = t(x)/2 + a/2t(x)$$

From this set of equations it is possible to eliminate $t(x)$, resulting in the envelope

$$\mathcal{E}_3 : \tau = \frac{b}{4q} + \frac{aq}{b}.$$

In context with a hyperbolic event, representing a seismic signal, $a = (2h/c)^2$ and $b = 1/c^2$ gives

$$(ct)^2 = x^2 + (2h)^2 \iff \tau = \frac{1}{(2c)^2 q} + (2h)^2 q.$$

The hyperbola \mathcal{C}_1 , with depth $h = 0.2$ km and velocity $v = 1.5$ km/s, including a family of parabolas tangential to the curve is shown in Figure 4(a). In the corresponding transform domain, shown in Figure 4(b), each parabola is represented as a line, forming the envelope \mathcal{E}_3 in red.

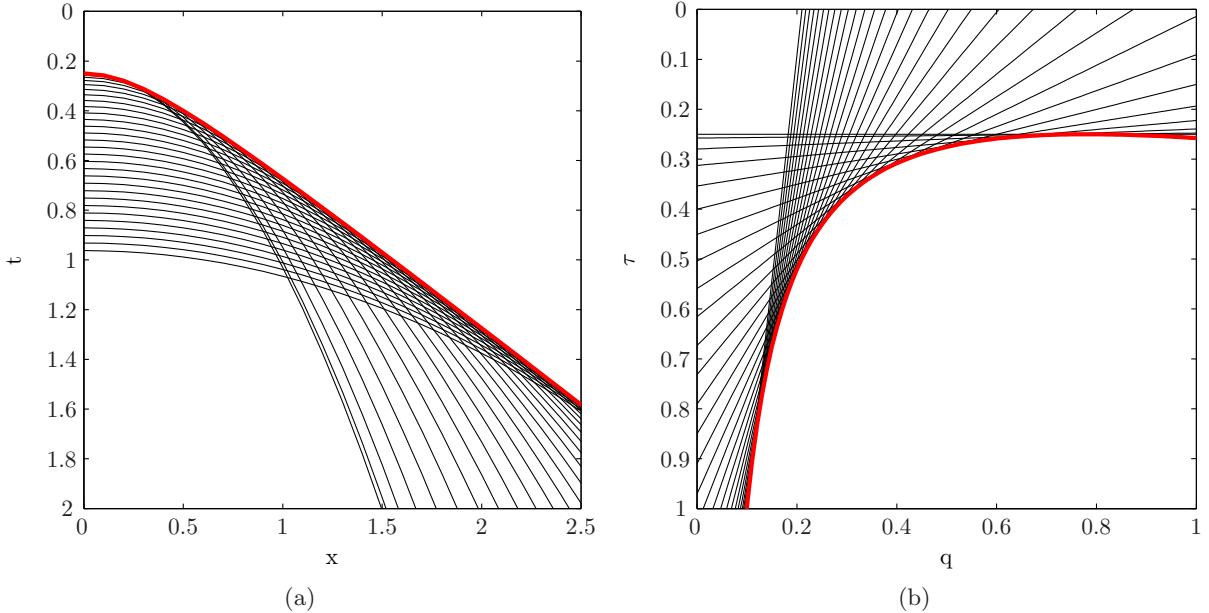


Figure 4: Kinematics of the parabolic Radon transform. A hyperbolic event is transformed to another hyperbola: $\mathcal{C}_1 \iff \mathcal{E}_3$. Every parabola, tangential to the red hyperbolic event, is represented as a line in the transform domain, forming the envelope \mathcal{E}_3 (red).

3. Curvature and Focusing

Curvature of the hyperbolic event \mathcal{C}_1 is used for describing focusing power of the parabolic Radon transform. Curvature is defined as

$$\kappa(x) = \frac{\partial^2 t / \partial x^2}{(1 + (\partial t / \partial x)^2)^{3/2}} = \frac{dp(x) / dx}{(1 + p^2(x))^{3/2}}.$$

From the previous analysis, the relation between p and q is

$$dt/dx = p(x) = 2xq.$$

Let $x = 0$, then the following is true when symmetry of $t(x)$ is assumed,

$$p(0) = 0 \implies \kappa(0) = dp(0) / dx \implies q(0) = \kappa(0)/2,$$

where $q(0)$ is not defined unless $p(0) = 0$. The apex of \mathcal{E}_3 at (q_a, τ_a) is given by $d\tau/dq = 0$,

$$q_a = \frac{b}{2\sqrt{a}} \quad \text{and} \quad \tau_a = \sqrt{a}.$$

That is, curvature in the apex of \mathcal{E}_3 is

$$\mathcal{K}_a = \left. \frac{d^2\tau}{dq^2} \right|_a = \frac{b}{2q_a^3} = \frac{\tau_a}{q_a^2} = \frac{4t(0)}{\kappa^2(0)},$$

where $\kappa(0) = b/\sqrt{a}$ is the curvature at the apex of the hyperbolic curve \mathcal{C}_1 (Maeland, 1998). As a side note, it can be shown that the parabolic Radon transform of hyperbolic traveltimes curves with nonzero apex disrupts seismic images by changing geometry (Maeland, 2003). The hyperbola in tau- q is split into two disjoint branches with a characteristic cusp.

It has been established a relation between the curvature in the apex of \mathcal{C}_1 and the hyperbolic envelope \mathcal{E}_3 . The curvature in apex of the envelope \mathcal{K}_a is inverse proportional to the square of the curvature κ of the hyperbolic traveltime curve at $x = 0$.

The largest curvature of κ will be from shallow reflections at zero-offset, while the curvature of \mathcal{K}_a will have highest value for deep reflectors. Hence, deep reflectors will yield better focus. In practice, the parabolic Radon transform of a hyperbolic event is only represented as a small part of the curve \mathcal{E}_3 . This is a consequence of the limited aperture in x .

The parabolic Radon transform of a hyperbolic curve \mathcal{C}_1 is shown in Figure 5. It represents a seismic event where $h = 0.8$ km is the depth of the reflective boundary and $v = 1.8$ km/s is the velocity. A synthetic seismogram, using a ricker-wavelet with center-frequency $\nu_0 = 7$ Hz is transformed by the standard damped least squares technique (discussed in the next chapter). In figures 5(a) and 5(c) the apex point (q_a, τ_a) is indicated with blue, while the dashed line represents the complete theoretical hyperbola \mathcal{E}_3 .

Figure 6 shows hyperbolas from depths $h = \{600, 1100, 1700\}$ m with constant velocity $v = 1.5$ km/s (for simplicity). This illustrates how deep reflectors yield better focus.

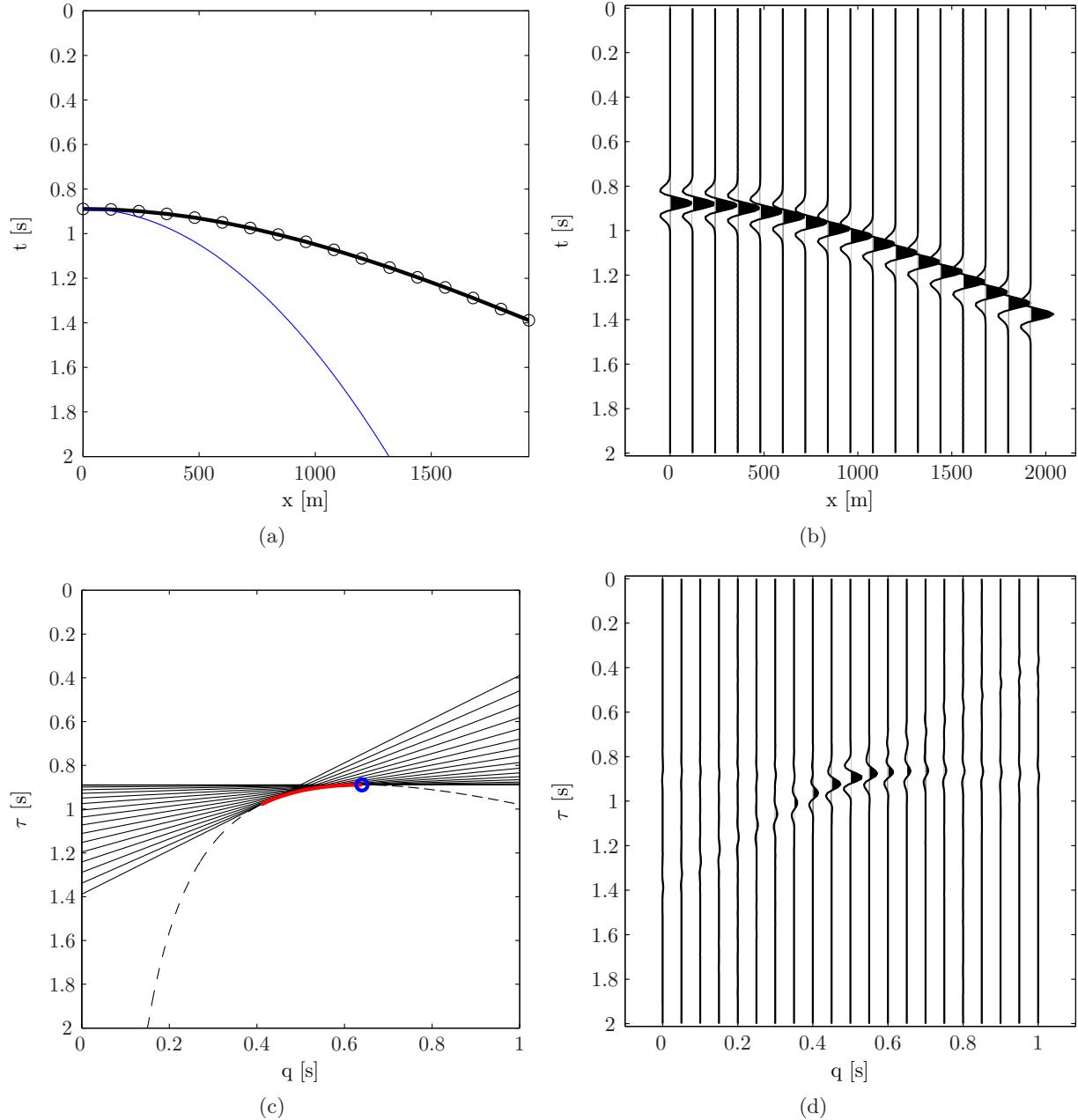


Figure 5: The parabolic Radon transform showing how the focus is limited to a certain part (red) of the theoretical hyperbola (dashed). The apex shown as a blue circle in the Radon panel, with its corresponding parabola as a blue line in the time-offset domain.

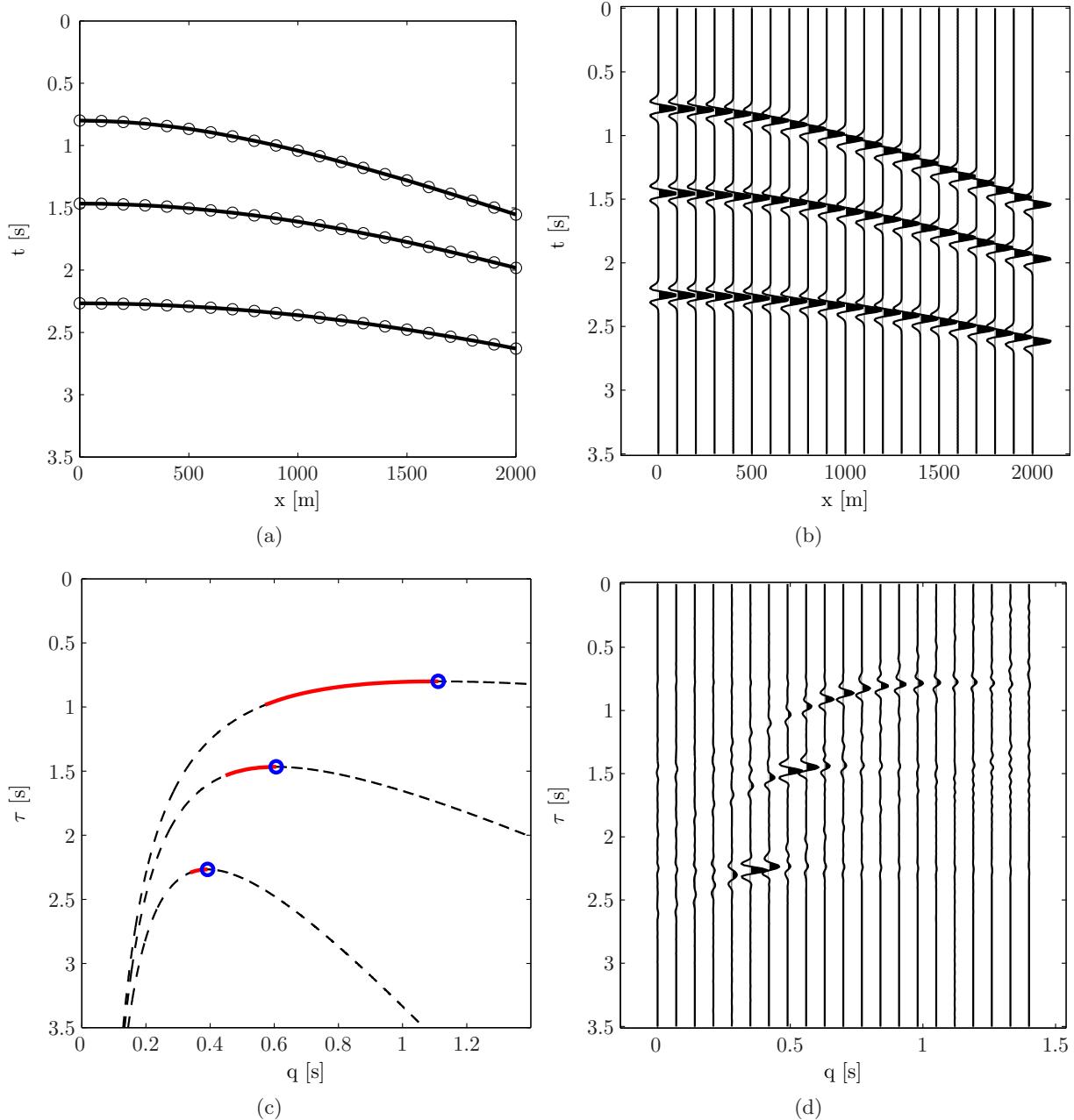


Figure 6: Focusing of the parabolic Radon transform by three traveltimes curves, representing different depths.

CHAPTER 3

Least Squares and Regularization Techniques

Finding a solution to an inverse problem is the process of finding a model, \mathbf{x} , fitting some data, \mathbf{b} . A transform or function is specified from either the laws of physics or derived from mathematics. Further classification defines the forward problem (finding \mathbf{b} given some \mathbf{x}) and the system identification problem (determining the operator A given examples of \mathbf{x} and \mathbf{b}). Inverse problems could be both linear and non-linear, but the linear problem is the motivation for the present work. A linear inverse problem can then be written

$$A(\mathbf{x}) = \mathbf{Ax} = \mathbf{b}.$$

Linear systems obey superposition

$$A(\mathbf{x}_1 + \mathbf{x}_2) = A(\mathbf{x}_1) + A(\mathbf{x}_2)$$

and scaling

$$A(\alpha\mathbf{x}) = \alpha A(\mathbf{x}).$$

It is important to mathematically classify the solution of an inverse problem. There are several issues that should be considered: existence, uniqueness and instability of the solution. Even if an exact solution exist, it may not be unique, it may exist infinitely many solutions. Consider a rank-deficient discrete linear inverse problem; the matrix \mathbf{A} has a non-trivial null space. Let \mathbf{x}_0 be a vector in the null space, then \mathbf{x}_0 is a solution to

$$\mathbf{Ax}_0 = \mathbf{0}.$$

This implies that any linear combination of vectors with basis in the null space can be added to any model without changing its fit to the data. The solution could have infinitely many mathematically acceptable models, because

$$\mathbf{A}(\mathbf{x} + \mathbf{x}_0) = \mathbf{Ax} + \mathbf{Ax}_0 = \mathbf{b}.$$

The process of calculating an inverse solution can be, and often is, numerically unstable. Small changes in discretization or measurements can lead to an enormous change in the estimated model. Inverse problems where this situation arises are referred to as ill-posed in the case of continuous systems, or ill-conditioned in the case of discrete linear systems.

This chapter focuses on the least squares method, and different regularization techniques, applicable to the ill-condition least squares Radon transform (Kostov, 1990; Schonewille and Duijndam, 2001). It is important to note that forward problem in inverse theory means the inverse Radon transform $r(p, \tau) \rightarrow f(x, t)$ (reconstruction). On the other hand, the inverse problem means the forward Radon transform $f(x, t) \rightarrow r(p, \tau)$. The latter involves inversion of a linear operator in the least squares sense.

To clarify notation used in this Chapter, the ℓ_p -norm is defined as

$$\|\mathbf{z}\|_p = \left(\sum_{j=1}^M |z_j|^p \right)^{\frac{1}{p}},$$

where $p \geq 1$, $\mathbf{z} \in \mathbb{C}^M$ and $|z|^2 = zz^*$ is the modulus. As a motivational example, let $\mathbf{x} \in \mathbb{R}^M$ and assume the simplest norm approximation problem as an unconstrained problem of the form

$$(20) \quad \min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_1.$$

This yields an approximate solution $\mathbf{Ax} \approx \mathbf{b}$ unless the residuals $\mathbf{r} = \mathbf{Ax} - \mathbf{b}$ are zero. The norm approximation problem in Equation (20) is a convex problem (Boyd and Vandenberghe, 2004) and there is always at least one optimal solution. Expressing a linear system as

$$\mathbf{Ax} = x_1 \mathbf{a}_1 + \cdots + x_M \mathbf{a}_M$$

where $\mathbf{a}_1, \dots, \mathbf{a}_M$ are the columns of \mathbf{A} . These vectors are called regressors and the vector \mathbf{Ax} , where \mathbf{x} is an optimal solution of the problem, is called the regression of \mathbf{b} .

1. Least Squares and ℓ_p -norm Regularization

Assume a linear overdetermined system $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{x} \in \mathbb{R}^M$, $\mathbf{b} \in \mathbb{R}^N$ and $\mathbf{A} \in \mathbb{R}^{N \times M}$. The least squares solution minimizes the residual sum of squared errors

$$(21) \quad \arg \min_{\mathbf{x}} f(\mathbf{x}) = \sum_{n=1}^N |b_n - \sum_{m=1}^M a_{n,m} x_m|^2 = \|\mathbf{Ax} - \mathbf{b}\|_2^2.$$

This can be seen as a continuous differentiable and unconstrained convex optimization problem. Expressing the objective as a quadratic function

$$(22) \quad f(\mathbf{x}) = \mathbf{x}^T \mathbf{A}^T \mathbf{Ax} - 2\mathbf{b}^T \mathbf{Ax} + \mathbf{b}^T \mathbf{b},$$

and minimizing

$$\nabla f(\mathbf{x}) = 2\mathbf{A}^T \mathbf{Ax} - 2\mathbf{A}^T \mathbf{b} = \mathbf{0},$$

gives the normal equations,

$$(23) \quad \mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b}.$$

The convex optimization problem is reduced to a set of linear equations. This is a closed form solution to the least squares problem. The inverse operator

$$\mathbf{A}^\dagger = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T,$$

is referred to as the left inverse or generalized inverse. In case of an underdetermined problem $M > N$, the operator is given as

$$\mathbf{A}^T (\mathbf{A} \mathbf{A}^T)^{-1},$$

often referred to as the right inverse, yielding the minimum length solution. The present work will focus on the least squares formulation which is used in further discussions on singular value decomposition and regularization techniques.

1.1. Singular value decomposition (SVD) and truncation. For a more in depth analysis of the problem, singular value decomposition (Aster et al., 2005) is used. A valuable tool when analyzing rank-deficient or ill-conditioned systems.

DEFINITION 1 (Singular Value Decomposition). *Let $\mathbf{A} \in \mathbb{R}^{N \times M}$, the singular value decomposition of \mathbf{A} is the factorization*

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T$$

where \mathbf{U} ($N \times N$) and \mathbf{V} ($M \times M$) are both unitary. Σ is an $N \times M$ diagonal matrix with nonnegative diagonal elements called singular values. The singular values along the diagonal are now arranged as $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min(N,M)} \geq 0$.

Suppose $p = \min(N, M)$, the matrix Σ is written

$$\Sigma = \begin{pmatrix} \Sigma_p & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

where Σ_p is a $p \times p$ matrix. Writing

$$\mathbf{A} = (\mathbf{U}_p \quad \mathbf{U}_0) \begin{pmatrix} \Sigma_p & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} (\mathbf{V}_p \quad \mathbf{V}_0)^T,$$

the SVD can be reduced to its compact form

$$\mathbf{A} = \mathbf{U}_p \Sigma_p \mathbf{V}_p^T.$$

Simplicity and computer efficiency are consequences of having the SVD in this form. A common application of SVD is computation of the generalized inverse, also known as the Moore-Penrose-pseudoinverse (Golub and Van Loan, 1989), defined as

$$\mathbf{A}^\dagger = \mathbf{V}_p \Sigma_p^{-1} \mathbf{U}_p^T.$$

The pseudoinverse is implemented in MATLAB as the function *pinv*, in which you can provide a tolerance limit¹, such that singular values smaller than this limit are not included in the computation (Aster et al., 2005). Formulating Equation (23) using SVD, yields

$$\begin{aligned} \mathbf{x} &= (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \\ &= \mathbf{V}_p \Sigma_p^{-2} \mathbf{V}_p^T \mathbf{V}_p \Sigma_p \mathbf{U}_p^T \mathbf{b} \\ &= \mathbf{V}_p \Sigma_p^{-1} \mathbf{U}_p^T \mathbf{b} \\ &= \mathbf{A}^\dagger \mathbf{b}. \end{aligned}$$

With the generalized inverse solution, there are no contributions from the null space. However, the presence of very small singular values in the denominator can lead to a very large coefficient in the corresponding model space basis vector \mathbf{v}_j , which can dominate the solution. In a worst case, this could actually amplify noise (Aster et al., 2005). A common method for dealing with this problem is to truncate very small singular values (decrease p), and thus eliminating model space vectors.

¹By default the tolerance limit in MATLAB is given as $\max(\text{size}(\mathbf{A})) \cdot \text{norm}(\mathbf{A}) \cdot \epsilon$, where ϵ is the ‘computer-epsilon’.

The solution becomes stable, but on the cost of reduced resolution. Suppose \mathbf{u}_j and \mathbf{v}_j are the column-vectors of the matrices \mathbf{U} and \mathbf{V} , the truncated SVD is

$$\begin{aligned}\mathbf{x} &= \sum_{j=1}^{p'(\alpha)} \frac{\mathbf{u}_j^T \mathbf{b}}{s_j} \mathbf{v}_j, \quad p'(\alpha) < p \\ &= \sum_{j=1}^p f_j(\alpha) \frac{\mathbf{u}_j^T \mathbf{b}}{s_j} \mathbf{v}_j,\end{aligned}$$

where

$$f_j(\alpha) = \begin{cases} 1, & s_j \geq \alpha \\ 0, & s_j < \alpha, \end{cases}$$

and α is a parameter chosen by the user, referred to as the regularization parameter. Different techniques have been developed for choosing an optimal regularization parameter, literature on this subject (Hansen, 1998) discusses the L-curve method, the discrepancy principle and generalized cross-validation.

1.2. Regularization by different penalty functions. Another technique, known as ridge regression² or Tikhonov regularization, addresses the problem in a more sophisticated manner. This is referred to as ℓ_2 -norm regularization, an effective and tool for achieving numerical stability. The minimization problem is

$$\arg \min_{\mathbf{x}} f(\mathbf{x}) = \|\mathbf{Ax} - \mathbf{b}\|_2^2 + \|\mathbf{\Gamma}\mathbf{x}\|_2^2,$$

where $\mathbf{\Gamma}$ is some selected Tikhonov matrix. This is often chosen as the identity matrix $\mathbf{\Gamma} = \alpha \mathbf{I}$, where α is the regularization parameter. With these assumptions, the minimization problem is given as another convex quadratic optimization problem

$$f(\mathbf{x}) = \mathbf{x}^T (\mathbf{A}^T \mathbf{A} + \alpha^2 \mathbf{I}) \mathbf{x} - 2 \mathbf{b}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{b}.$$

Finding $\nabla f(\mathbf{x}) = \mathbf{0}$, the closed form solution is

$$\mathbf{x} = (\mathbf{A}^T \mathbf{A} + \alpha^2 \mathbf{I})^{-1} \mathbf{A}^T \mathbf{b}.$$

A more explicit expression for how this stabilizes the solution, is given by the SVD formulation of Tikhonov regularization

$$\mathbf{x} = \sum_{j=1}^{\min(N,M)} \frac{s_j^2}{s_j^2 + \alpha^2} \frac{\mathbf{u}_j^T \mathbf{b}}{s_j} \mathbf{v}_j.$$

The regularization parameter α is adjusted to give the right trade-off between making the loss function $\|\mathbf{Ax} - \mathbf{b}\|_2^2$ small, and the penalty function $\|\mathbf{x}\|_2^2$ not too big (Boyd and Vandenberghe, 2004).

Unfortunately, when experimenting with the Radon transform, none of these techniques prove to be superior. There are small differences, but they could not be considered significant. The transform is computationally expensive and the minimal gain in resolution or exactness cannot support the computational burden required by the techniques for calculating an optimal regularization parameter. Utilizing the matrix (Toeplitz) structure for faster computations is discussed in Chapter 4.

²Bridge regression for generalized ℓ_p -norm.

The Radon transform suffers from different smearing effects (Thorson and Claerbout, 1985), and it would be beneficial to find regularization techniques, able to focus separate events in the Radon panel. Regularization techniques pursuing a sparse Radon panel is presented in Chapter 6. One of the techniques presented is the basis pursuit method, thus, a short introduction to ℓ_1 -norm regularization for real numbers are given here. Consider the minimization problem

$$(24) \quad \arg \min_{\mathbf{x}} f(\mathbf{x}) = \|\mathbf{Ax} - \mathbf{b}\|_2^2 + 2\alpha\|\mathbf{x}\|_1,$$

where the residuals are minimized in the ℓ_2 -norm and the regularization is done by the ℓ_1 -norm. This is an unconstrained convex optimization problem, non-differentiable with respect to \mathbf{x} when $x_m = 0$. The objective function can be expressed as

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{A}^T \mathbf{A} \mathbf{x} - 2\mathbf{b}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{b} + q(\mathbf{x}),$$

where $q(\mathbf{x}) = 2\alpha\|\mathbf{x}\|_1$. Ignoring the fact that $q(\mathbf{x})$ is non-differentiable at any point where one of the elements of \mathbf{x} are zero, the derivatives can be expressed as

$$\nabla q(\mathbf{x}) = 2\alpha \mathbf{R}(\mathbf{x}) \mathbf{x},$$

where $\mathbf{R} = \text{diag}(1/|x_m|)$, for $m = 1, 2, \dots, M$. Minimizing $f(\mathbf{x})$, yields

$$\nabla f(\mathbf{x}) = 2\mathbf{A}^T \mathbf{A} \mathbf{x} - 2\mathbf{A}^T \mathbf{b} + 2\alpha \mathbf{R}(\mathbf{x}) \mathbf{x} = \mathbf{0},$$

which since \mathbf{R} depends on \mathbf{x} leads to the non-linear system of equations

$$\mathbf{x} = (\mathbf{A}^T \mathbf{A} + \alpha \mathbf{R}(\mathbf{x}))^{-1} \mathbf{A}^T \mathbf{b}.$$

To solve this problem an optimization algorithm is required. A simple approach is to use iteratively reweighted least squares (Bjorck, 1996), with some stopping criterion (without guarantees for convergence), noting that some perturbation factor has to be added when $x_m = 0$. The problem with this formulation is that many signal processing problems, including frequency domain methods for time-invariant Radon transforms, are given in the complex plane. In fact, the complex absolute value is continuous everywhere, but (complex) differentiable nowhere. The reminder of this chapter is devoted to problems in the complex plane.

2. Complex Roots and Newton's Method

Finding roots of equations are an important mathematical problem. In the field of applied mathematics, it exists many methods for finding complex (and real) roots. Newton's method is a well established and efficient method, regularly used in optimization problems. Considering real valued functions, the geometric principle of the method is as follows. Draw a tangent line at the initial guess $(x_0, f(x_0))$, and find the intercept at x_1 . If the solution converges, this intercept will be closer to the root than x_0 . In the complex case this geometric principle is not as trivial. However, if $f(z)$ is analytic, the method makes formal sense for complex numbers.

Finding the roots of $f(z) = 0$ require a function $f(z)$ to be differentiable, with an initial guess $z = z_0$. Suppose the elements x and y are the real and imaginary part of the complex number z , defined as $z = x + iy$. The function $f(z)$ is said to be differentiable at z_0 , if its derivative at z_0 exists. For a complex function to be differentiable, it has to satisfy the Cauchy-Riemann equations (Brown and Churchill, 1974).

From this, a function f of a complex variable z is analytic in an open set³, if it has a derivative at each point in that set. The complex Newton method (Devaney, 1994) is given as

$$z_{k+1} = z_k - \frac{f(z_k)}{f'(z_k)},$$

for $k = 0, 1, 2, \dots$. Newton's method could be used for finding minimum or maximum of a function by applying it to its derivative, assuming the function is twice differentiable, this is known as Newton's method in optimization.

Stopping criterions for this algorithm is either when the convergence reaches some tolerance limit where the approximation is acceptable, or when reaching a defined maximum number of iterations. Even though Newton's method is an attractive and much used method, it has drawbacks worth mentioning: 1) Divergence near inflection points. 2) Division by zero. 3) Divergence from oscillations near local maximum and minimum. 4) Root jumping.

Checking the Newton basins (Devaney, 1994) of attraction of a function $f(z)$, with roots outside and inside the unit circle, yields different fractional patterns. As an exercise, Figure 1 demonstrates four different equations and their Newton basins. A red cross marks a root in the complex plane. Different colors in the image indicate how many iterations were required for convergence from an initial guess. The tolerance limit is $\epsilon = 10^{-2}$ and a maximum number of iterations is set to $k = 20$. For the examples in figures 1(c) and 1(d), let $f(z)$ have the roots a and b , in the following form

$$f(z) = (z - a)(z + b)(z^2 - 1) = 0,$$

and the derivative

$$f'(z) = (z^2 - 1)[(z - a) + (z + b)] + 2z(z - a)(z + b).$$

The roots are given as $z = \{a, -b, \pm 1\}$. Figure 1(c) shows $a = 2i$ and $b = 3/2 + i$ and Figure 1(d) shows $a = (2/10)i$ and $b = 1/8 + (3/2)i$.

2.1. Multivariable Newton's method. Considering a real-valued function $f(\mathbf{x})$ of the vector $\mathbf{x} \in \mathbb{R}^M$. A generalization of Newton's method to higher dimensions is straightforward by introducing a gradient and the inverse of the Hessian matrix, i.e.

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (\mathbf{H} f(\mathbf{x}_k))^{-1} \nabla f(\mathbf{x}_k), \quad k = 0, 1, 2, \dots$$

with $\det(\mathbf{H} f(\mathbf{x}_k)) \neq 0$. This is also known as the pure Newton's method with a fixed step size of one, and with the Newton decrement

$$\lambda(\mathbf{x}) = [\nabla f(\mathbf{x})^T (\mathbf{H} f(\mathbf{x}))^{-1} \nabla f(\mathbf{x})]^{1/2},$$

at \mathbf{x} as a stopping criterion based on some tolerance limit (Boyd and Vandenberghe, 2004). Calculations of a M -dimensional quadratic function will converge in only one step from any starting point, thus the algorithm applied to a system of this kind does not require any stopping criterion with respect to convergence. For a complex Newton method, definitions of a complex gradient and Hessian operator is needed. This is used as a motivation for the following analysis.

³A set is open if it contains none of its boundary points.

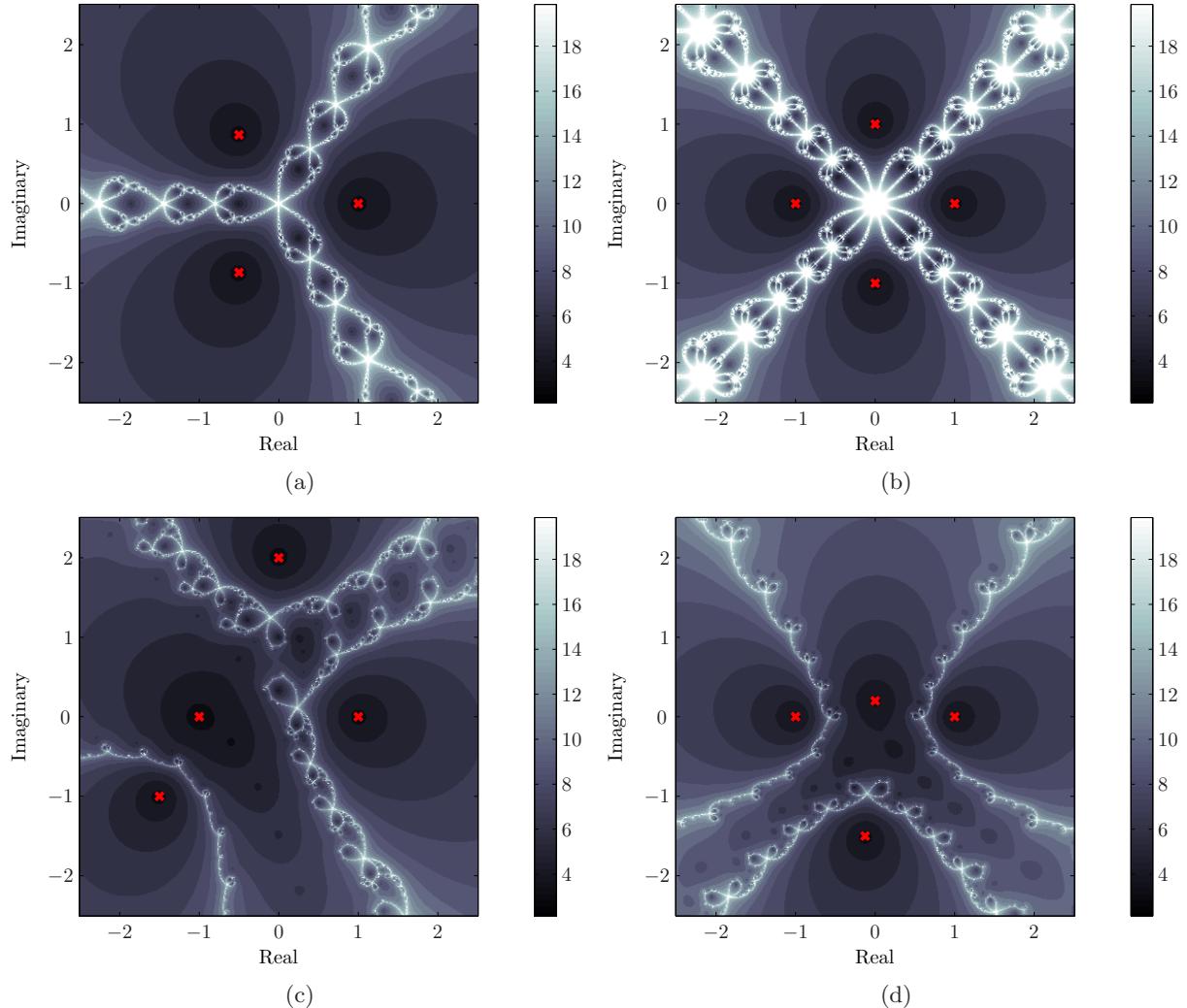


Figure 1: Newton basins of attraction with roots indicated by red crosses. Colors represent the number of iterations before reaching a given tolerance limit of $\epsilon = 10^{-2}$, with a maximum iteration count of $k = 20$. Roots of a) $f(z) = z^3 - 1$. b) $f(z) = z^4 - 1$. c,d) $f(z) = (z - a)(z + b)(z^2 - 1)$.

3. Complex Differential Operators Using CR-Calculus

In signal processing, complex-valued data are encountered and is often subject to different engineering shortcuts. Prior to generalizing the complex Newton method to multiple variables, the complex gradient and Hessian are discussed. It is important to note the difference between the complex and real gradient operators, and recognize that the complex results are not obtained by simply replacing real variables with complex ones. Utilizing the full power of the complex domain with real and imaginary parts leads to a more complicated analysis, although in some cases, it could yield the same end result.

In many optimization problems, the objective functions are typically $f : \mathbb{C} \rightarrow \mathbb{R}$, which imply that f is not complex analytic by the Cauchy-Riemann equations. Wirtinger (1927) relaxed the strong requirement for differentiability from the Cauchy-Riemann equations, and introduced Wirtinger calculus using real differentiation rules. Regarding a function $f : \mathbb{C} \rightarrow \mathbb{C}$ as $f : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{C}$, by writing $f(z) = f(x, y)$, thus making use of the underlying \mathbb{R}^2 structure, forms a basis for the following theorem.

THEOREM 1 (Li and Adali (2008)). *Let $f : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{C}$ be a function of real variables x and y such that $g(z, z^*) = f(x, y)$, where $z = x + iy$, and g is analytic with respect to z^* and z independently. Then,*

- i. *the partial derivatives*

$$\frac{\partial g}{\partial z} = \frac{1}{2} \left(\frac{\partial f}{\partial x} - i \frac{\partial f}{\partial y} \right), \quad \frac{\partial g}{\partial z^*} = \frac{1}{2} \left(\frac{\partial f}{\partial x} + i \frac{\partial f}{\partial y} \right)$$

can be computed by treating z^ as a constant in g and z as a constant, respectively;*

- ii. *a necessary and sufficient condition for f to have a stationary point is that $\partial g / \partial z = 0$. Similarly, $\partial g / \partial z^* = 0$ is also a necessary and sufficient condition.*

This makes up the framework for further development using several complex variables. A complex gradient and Hessian (Brandwood, 1983; Van Den Bos, 1994) is summarized in a note by Kreutz-Delgado (2005). A note from the electrical engineering community, reviewing development within the field of complex multivariable calculus. The present section contains a brief summary of some important points which are used to derive the linear least squares problem and the complex Newton update.

Complex vectors $\mathbf{z} = \mathbf{x} + iy$ can be represented in three different vector spaces. First, the canonical M -dimensional complex vector space $\mathbf{z} \in \mathbb{C}^M$. A second, is the canonical real vector space

$$\mathbf{r} = \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \in \mathcal{R} \triangleq \mathbb{R}^{2M},$$

and a third given by a set of conjugate coordinate vectors

$$\mathbf{c} = \begin{pmatrix} \mathbf{z} \\ \mathbf{z}^* \end{pmatrix} \in \mathcal{C} \subset \mathbb{C}^{2M},$$

where \mathcal{C} is a subset of the $4M$ dimensional complex vector space \mathbb{C}^{2M} . The subset \mathcal{C} can be viewed as a complex subset of \mathbb{C}^{2M} or as a real vector space isomorphic to \mathbb{R}^{2M} (Fraleigh, 1994). With these definitions, the parameters \mathbf{c} , $(\mathbf{z}, \mathbf{z}^*)$, \mathbf{z} , (\mathbf{x}, \mathbf{y}) , all denote the same point

$$f(\mathbf{c}) = f(\mathbf{z}, \mathbf{z}^*) = f(\mathbf{z}) = f(\mathbf{x}, \mathbf{y}) \in \mathbb{C}^N.$$

The cogradient and conjugate cogradient operators are then defined respectively as row operators

$$\frac{\partial}{\partial \mathbf{z}} \triangleq \left(\frac{\partial}{\partial z_1}, \dots, \frac{\partial}{\partial z_M} \right) \quad \text{and} \quad \frac{\partial}{\partial \mathbf{z}^*} \triangleq \left(\frac{\partial}{\partial z_1^*}, \dots, \frac{\partial}{\partial z_M^*} \right).$$

The formalism in Theorem 1 is also at true for the multivariable case. When applying the cogradient operator $\partial/\partial \mathbf{z}$, then \mathbf{z}^* is formally treated as a constant, and when applying the cogradient operator $\partial/\partial \mathbf{z}^*$, then \mathbf{z} is formally treated as a constant.

The linear mapping between the coordinate systems \mathcal{R} and \mathcal{C} is defined by

$$\mathbf{c} = \mathbf{J}\mathbf{r} = \begin{pmatrix} \mathbf{I} & i\mathbf{I} \\ \mathbf{I} & -i\mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}$$

where \mathbf{I} is the $M \times M$ identity matrix. It can be shown that $\mathbf{J}^{-1} = (1/2)\mathbf{J}^H$, i.e. the mapping between \mathcal{R} and \mathcal{C} is one-to-one, and both spaces are isomorphic real vector spaces of dimension $2M$, with a Jacobian

$$\mathbf{J}_c \triangleq \frac{\partial}{\partial \mathbf{r}} \mathbf{c}(\mathbf{r}) = \frac{\partial}{\partial \mathbf{r}} \mathbf{J}\mathbf{r} = \mathbf{J}.$$

The Jacobian of the inverse transformation $\mathbf{r} = \mathbf{r}(\mathbf{c})$ is

$$\mathbf{J}_r = \mathbf{J}_c^{-1} = \mathbf{J}^{-1} = (1/2)\mathbf{J}^H.$$

This gives the following first-order relationships

$$\Delta \mathbf{c} = \mathbf{J}_c \Delta \mathbf{r} = \mathbf{J} \Delta \mathbf{r} \quad \text{and} \quad \Delta \mathbf{r} = \mathbf{J}_r \Delta \mathbf{c} = (1/2)\mathbf{J}^H \Delta \mathbf{c}.$$

From this framework the cogradient operators are

$$\frac{\partial(\cdot)}{\partial \mathbf{c}} = \frac{\partial(\cdot)}{\partial \mathbf{r}} \mathbf{J}_r = \frac{1}{2} \frac{\partial(\cdot)}{\partial \mathbf{r}} \mathbf{J}^H \quad \text{and} \quad \frac{\partial(\cdot)}{\partial \mathbf{r}} = \frac{\partial(\cdot)}{\partial \mathbf{c}} \mathbf{J}_c = \frac{\partial(\cdot)}{\partial \mathbf{c}} \mathbf{J}.$$

The purpose of this section is to introduce the basic definitions, enough to define a complex gradient (or cogradient) and a complex Hessian operator. Using these definitions the linear least squares and Newton's method could be presented. In the next subsection a complex gradient operator for real-valued functions is defined as it was first proposed by Brandwood (1983).

3.1. The complex gradient of real-valued functions. The following theorem defines a complex gradient operator for real-valued functions

$$\nabla_z = [\partial/\partial z_1, \partial/\partial z_2, \dots, \partial/\partial z_m]^T, \quad \nabla_{z^*} = [\partial/\partial z_1^*, \partial/\partial z_2^*, \dots, \partial/\partial z_m^*]^T,$$

a definition sufficient for the present problems.

THEOREM 2 (Brandwood (1983)). *Let $f : \mathbb{C}^M \rightarrow \mathbb{R}$ be a real-valued scalar function of a complex vector \mathbf{z} . Let $f(\mathbf{z}) = g(\mathbf{z}, \mathbf{z}^*)$, where $g : \mathbb{C}^M \times \mathbb{C}^M \rightarrow \mathbb{R}$ is a real-valued function of two complex variables and g is analytic with respect to each z_k and z_k^* . Then either of the conditions $\nabla_z g = \mathbf{0}$ or $\nabla_{z^*} g = \mathbf{0}$ is necessary and sufficient to determine stationary points of f .*

Investigating three scalar functions gives some general results which can easily be adapted to different problems. Suppose $\mathbf{a} \in \mathbb{C}^M$ and $\mathbf{R} \in \mathbb{C}^{M \times M}$, then the following is true,

$$\nabla_{z^*}(\mathbf{a}^H \mathbf{z}) = \mathbf{0}, \quad \nabla_{z^*}(\mathbf{z}^H \mathbf{a}) = \mathbf{a}, \quad \nabla_{z^*}(\mathbf{z}^H \mathbf{R} \mathbf{z}) = \mathbf{R} \mathbf{z}$$

and

$$\nabla_z(\mathbf{a}^H \mathbf{z}) = \mathbf{a}^*, \quad \nabla_z(\mathbf{z}^H \mathbf{a}) = \mathbf{0}, \quad \nabla_z(\mathbf{z}^H \mathbf{R} \mathbf{z}) = \mathbf{R}^T \mathbf{z}^* = (\mathbf{R}^H \mathbf{z})^*.$$

3.2. Complex linear least squares. From the previously discussed least squares minimization problem in Equation (21), a complex variant is defined

$$(25) \quad \arg \min_{\mathbf{z}} f(\mathbf{z}) = \|\mathbf{Az} - \mathbf{b}\|_2^2,$$

where $\mathbf{z} \in \mathbb{C}^M$, $\mathbf{b} \in \mathbb{C}^N$ and $\mathbf{A} \in \mathbb{C}^{N \times M}$. The function $f(\mathbf{z})$ generates a real-valued scalar, and from Theorem 2, this is a sufficient criterion to minimize the objective function. The complex gradient operator ∇_z , with respect to a complex M -dimensional vector \mathbf{z} , generates another complex M -dimensional vector from a scalar function of \mathbf{z} . Let \mathbf{A}^H denote the conjugate transpose of \mathbf{A} , and consider the objective function of the least squares problem

$$\begin{aligned} f(\mathbf{z}) &= (\mathbf{Az} - \mathbf{b})^H (\mathbf{Az} - \mathbf{b}) \\ &= (\mathbf{z}^H \mathbf{A}^H - \mathbf{b}^H)(\mathbf{Az} - \mathbf{b}) \\ &= \mathbf{z}^H \mathbf{A}^H \mathbf{Az} - \mathbf{z}^H \mathbf{A}^H \mathbf{b} - \mathbf{b}^H \mathbf{Az} + \mathbf{b}^H \mathbf{b}, \end{aligned}$$

where the matrix $\mathbf{A}^H \mathbf{A} \in \mathbb{C}^{M \times M}$. Stationary points of $f(\mathbf{z})$ are now determined by

$$\nabla_{z^*} f(\mathbf{z}) = \mathbf{A}^H \mathbf{Az} - \mathbf{A}^H \mathbf{b} = \mathbf{0}$$

or

$$\begin{aligned} \nabla_z f(\mathbf{z}) &= (\mathbf{A}^H \mathbf{A})^T \mathbf{z}^* - \mathbf{A}^T \mathbf{b}^* \\ &= [(\mathbf{A}^H \mathbf{A})^H \mathbf{z}]^* - \mathbf{A}^T \mathbf{b}^*. \end{aligned}$$

Because $(\mathbf{A}^H \mathbf{A})^H = \mathbf{A}^H \mathbf{A}$ (Hermitian matrix), the gradient can be written

$$\nabla_z f(\mathbf{z}) = (\mathbf{A}^H \mathbf{Az})^* - \mathbf{A}^T \mathbf{b}^* = \mathbf{0}.$$

From this, the symmetry $\nabla_{z^*} f(\mathbf{z}) = (\nabla_z f(\mathbf{z}))^*$ is revealed and a closed form solution for complex numbers is

$$\mathbf{A}^H \mathbf{Az} = \mathbf{A}^H \mathbf{b}.$$

Not surprisingly, the real equivalent, except for the Hermitian conjugate transpose operator.

3.3. The complex Hessian of real-valued functions. A derivation of the complex Hessian is presented in Kreutz-Delgado (2005), where it is established in relation to Newton's method. The Hessian of a real-valued function on \mathbb{C}^M is given by

$$\mathcal{H}_{\mathbf{cc}}^{\mathbb{C}} = \frac{\partial}{\partial \mathbf{c}} \left(\frac{\partial f}{\partial \mathbf{c}} \right)^H = \begin{pmatrix} \mathcal{H}_{zz} & \mathcal{H}_{z^* z} \\ \mathcal{H}_{zz^*} & \mathcal{H}_{z^* z^*} \end{pmatrix},$$

where the partial Hessians are

$$\begin{aligned} \mathcal{H}_{zz} &\triangleq \frac{\partial}{\partial \mathbf{z}} \left(\frac{\partial f}{\partial \mathbf{z}} \right)^H, & \mathcal{H}_{z^* z} &\triangleq \frac{\partial}{\partial \mathbf{z}^*} \left(\frac{\partial f}{\partial \mathbf{z}} \right)^H \\ \mathcal{H}_{zz^*} &\triangleq \frac{\partial}{\partial \mathbf{z}} \left(\frac{\partial f}{\partial \mathbf{z}^*} \right)^H, & \mathcal{H}_{z^* z^*} &\triangleq \frac{\partial}{\partial \mathbf{z}^*} \left(\frac{\partial f}{\partial \mathbf{z}^*} \right)^H. \end{aligned}$$

According to Kreutz-Delgado (2005), the Newton update $\widehat{\Delta \mathbf{c}}$ is

$$\mathcal{H}_{\mathbf{cc}}^{\mathbb{C}} \widehat{\Delta \mathbf{c}} = - \left(\frac{\partial f(\mathbf{c})}{\partial \mathbf{c}} \right)^H,$$

which can be expanded to

$$(26) \quad \widehat{\Delta \mathbf{z}} = (\mathcal{H}_{zz} - \mathcal{H}_{z^* z} \mathcal{H}_{z^* z^*}^{-1} \mathcal{H}_{zz^*})^{-1} \left[\mathcal{H}_{z^* z} \mathcal{H}_{z^* z^*}^{-1} \left(\frac{\partial f}{\partial \mathbf{z}^*} \right)^H - \left(\frac{\partial f}{\partial \mathbf{z}} \right)^H \right].$$

At this point the complex operators (gradient and Hessian) have been defined and it is possible to evaluate the complex least squares objective function $f : \mathbb{C}^M \rightarrow \mathbb{R}$, given by

$$f(\mathbf{z}) = \mathbf{z}^H \mathbf{R} \mathbf{z} - \mathbf{z}^H \mathbf{A}^H \mathbf{b} - \mathbf{b}^H \mathbf{A} \mathbf{z} + \mathbf{b}^H \mathbf{b},$$

where $\mathbf{R} = \mathbf{A}^H \mathbf{A}$. Finding the explicit cogradients

$$\begin{aligned} \left(\frac{\partial f}{\partial \mathbf{z}} \right)^H &= (\mathbf{z}^H \mathbf{R} - \mathbf{b}^H \mathbf{A})^H = \mathbf{R}^H \mathbf{z} - \mathbf{A}^H \mathbf{b}, \\ \left(\frac{\partial f}{\partial \mathbf{z}^*} \right)^H &= (\mathbf{R} \mathbf{z} - \mathbf{A}^H \mathbf{b})^H = \mathbf{z}^H \mathbf{R}^H - \mathbf{b}^H \mathbf{A}, \end{aligned}$$

and using this to find the partial Hessians

$$\begin{aligned} \mathcal{H}_{\mathbf{z}\mathbf{z}} &= \frac{\partial}{\partial \mathbf{z}} (\mathbf{R}^H \mathbf{z} - \mathbf{A}^H \mathbf{b}) = \mathbf{R} \\ \mathcal{H}_{\mathbf{z}^*\mathbf{z}} &= \frac{\partial}{\partial \mathbf{z}^*} (\mathbf{R}^H \mathbf{z} - \mathbf{A}^H \mathbf{b}) = \mathbf{0} \\ \mathcal{H}_{\mathbf{z}\mathbf{z}^*} &= \frac{\partial}{\partial \mathbf{z}} (\mathbf{z}^H \mathbf{R}^H - \mathbf{b}^H \mathbf{A}) = \mathbf{0} \\ \mathcal{H}_{\mathbf{z}^*\mathbf{z}^*} &= \frac{\partial}{\partial \mathbf{z}^*} (\mathbf{z}^H \mathbf{R}^H - \mathbf{b}^H \mathbf{A}) = \mathbf{R}, \end{aligned}$$

satisfying $\mathbf{R}^H = \mathbf{R}$. The complex multivariable Newton update of $f(\mathbf{z})$ is then

$$\begin{aligned} \widehat{\Delta \mathbf{z}} &= -\mathbf{R}^{-1} (\mathbf{R}^H \mathbf{z} - \mathbf{A}^H \mathbf{b}) \\ (27) \quad &= -(\mathbf{A}^H \mathbf{A})^{-1} (\mathbf{A}^H \mathbf{A} \mathbf{z} - \mathbf{A}^H \mathbf{b}), \end{aligned}$$

which again is the real equivalent, only the Hermitian conjugate transpose operator is used. Similarity to the real case is a consequence of symmetry properties in the least squares objective function.

4. Diagonalization of the Complex Newton Update

The complex multivariable formulation of the Newton method could be further analyzed in scalar form. If \mathbf{R} is diagonalizable, then there exists a basis \mathcal{B} for \mathbb{C}^M , formed by the eigenvectors of \mathbf{R} . The $M \times M$ matrix \mathbf{R} is diagonalizable if and only if \mathbf{R} has M linearly independent eigenvectors. Then, formally

$$\mathbf{R} = \mathbf{P} \boldsymbol{\Lambda} \mathbf{P}^{-1} \quad \text{and} \quad \boldsymbol{\Lambda} = \mathbf{P}^{-1} \mathbf{R} \mathbf{P},$$

where $\boldsymbol{\Lambda}$ is a diagonal $M \times M$ matrix. If \mathcal{B} is the basis for \mathbb{C}^M , formed from the columns of \mathbf{P} , the following diagram describe the mappings.

$$\begin{array}{ccc} \mathbb{C}^M & \xrightarrow{\mathbf{R}^{-1}} & \mathbb{C}^M \\ \mathbf{P}^{-1} \downarrow & & \uparrow \mathbf{P} \\ \mathbb{C}_{\mathcal{B}}^M & \xrightarrow{\boldsymbol{\Lambda}^{-1}} & \mathbb{C}_{\mathcal{B}}^M \end{array}$$

The change of coordinates can be written

$$\mathbf{P}[\mathbf{z}]_{\mathcal{B}} = \mathbf{z} \quad \text{and} \quad [\mathbf{z}]_{\mathcal{B}} = \mathbf{P}^{-1} \mathbf{z}.$$

Suppose the complex multivariable Newton step on the following form

$$\begin{aligned}\mathbf{z}_{k+1} &= \mathbf{z}_k + \widehat{\Delta \mathbf{z}} \\ &= \mathbf{z}_k - \mathbf{R}^{-1} \left(\frac{\partial f}{\partial \mathbf{z}} \right)^H \\ &= \mathbf{z}_k - \mathbf{R}^{-1} \mathbf{u}, \quad k = 0, 1, 2, \dots\end{aligned}$$

where $\mathbf{u} = (\partial f / \partial \mathbf{z})^H$. Diagonalization of this expression yields

$$\begin{aligned}[\mathbf{z}_{k+1}]_{\mathcal{B}} &= [\mathbf{z}_k]_{\mathcal{B}} - \mathbf{P}^{-1} (\mathbf{P} \Lambda \mathbf{P}^{-1})^{-1} \mathbf{u} \\ &= [\mathbf{z}_k]_{\mathcal{B}} - \mathbf{P}^{-1} \mathbf{P} \Lambda^{-1} \mathbf{P}^{-1} \mathbf{u} \\ &= [\mathbf{z}_k]_{\mathcal{B}} - \Lambda^{-1} [\mathbf{u}]_{\mathcal{B}}.\end{aligned}$$

Manipulating the vector $[\mathbf{u}]_{\mathcal{B}}$, yields a more explicit expression

$$\begin{aligned}[\mathbf{u}]_{\mathcal{B}} = \mathbf{P}^{-1} \mathbf{u} &= \mathbf{P}^{-1} (\mathbf{R} \mathbf{z}_k - \mathbf{A}^H \mathbf{b}) \\ &= \mathbf{P}^{-1} (\mathbf{R} \mathbf{P} [\mathbf{z}_k]_{\mathcal{B}} - \mathbf{A}^H \mathbf{b}) \\ &= \Lambda [\mathbf{z}_k]_{\mathcal{B}} - \mathbf{w},\end{aligned}$$

where $\mathbf{w} = \mathbf{P}^{-1} \mathbf{A}^H \mathbf{b}$. With the eigenvalues $\boldsymbol{\lambda} = \text{diag}(\Lambda) = [\lambda_1, \dots, \lambda_M]^T$, the problem is evaluated by scalar operations in the new coordinate system. The recursion is given as

$$\begin{aligned}[z(m)_{k+1}]_{\mathcal{B}} &= [z(m)_k]_{\mathcal{B}} - [u_m]_{\mathcal{B}} \lambda_m^{-1} \\ &= [z(m)_k]_{\mathcal{B}} - ([z(m)_k]_{\mathcal{B}} - w_m \lambda_m^{-1}),\end{aligned}$$

where $z(m)$ denotes an element in \mathbf{z} for $m = 1, 2, \dots, M$. This gives the simple form

$$[z_m]_{\mathcal{B}} = w_m \lambda_m^{-1}.$$

From the invertible matrix theorem (Lay, 2006), the matrix \mathbf{R} is invertible if and only if the number 0 is not an eigenvalue of \mathbf{R} .

It is important to notice that when doing calculations with the ℓ_2 -norm, an M -dimensional quadratic function will converge in only one step from any starting point. Thus, the algorithm does not require any stopping criterion with respect to convergence. It is clear that Equation (27) is just the normal equations, when \mathbf{z} is initialized with zeros.

4.1. Relation between SVD and Eigenvalue Decomposition. Following the theory in this chapter, it is relevant to discuss the relation between singular value decomposition and eigenvalue decomposition (diagonalization). Considering examples with the same structures as exhibited by the Radon transform (discussed more in depth in Chapter 4 and 5). The eigenvalues and eigenvectors of a real symmetric matrix are real and the matrix of eigenvectors can be chosen orthogonal. For a complex Hermitian matrix the eigenvalues are also real, but with complex eigenvectors, these eigenvectors could be chosen such that they form a unitary matrix. For symmetric and Hermitian matrices, singular values and eigenvalues are related; a non-negative eigenvalue is also a singular value $\sigma = \lambda$, and the corresponding eigenvectors are equal. Suppose the matrix $\mathbf{A}^H \mathbf{A}$ is Hermitian symmetric with an eigenvalue decomposition

$$\mathbf{A}^H \mathbf{A} = \mathbf{P} \Lambda \mathbf{P}^H,$$

where \mathbf{P} is a unitary matrix of eigenvectors and Λ is diagonal matrix with the corresponding eigenvalues.

With an objective of finding a relation between the eigenvalues of $\mathbf{A}^H \mathbf{A}$, and the singular values of \mathbf{A} , let $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^H$ be a singular value decomposition such that

$$\mathbf{A}^H \mathbf{A} = \mathbf{V}\Sigma^H \mathbf{U}^H \mathbf{U}\Sigma\mathbf{V}^H = \mathbf{V}\Sigma^H \Sigma\mathbf{V}^H = \mathbf{P}\Lambda\mathbf{P}^H.$$

This implies that the singular values σ of \mathbf{A} are squares of the eigenvalues λ of $\mathbf{A}^H \mathbf{A}$,

$$\sigma = \sqrt{\lambda}.$$

To summarize: It is possible to factorize the matrix \mathbf{A} if its eigenvectors are linearly independent. If matrix \mathbf{A} can be eigendecomposed and none of its eigenvalues are zero, then \mathbf{A} is nonsingular and its inverse is given by $\mathbf{A}^{-1} = \mathbf{P}\Lambda^{-1}\mathbf{P}^{-1}$.

CHAPTER 4

Direct and Iterative Inversion Methods

In high-performance computing, fast, accurate and parallelizable methods for solving linear systems are important. Specific algorithms are developed for systems where the inverse operator possess structure or symmetry. Exploiting these properties is about selecting a suitable regularization technique and computational method.

As an introduction, the method of Cholesky factorization (Golub and Van Loan, 1989) is presented. Using this factorization to solve symmetric positive definite (Lay, 2006) systems, is possibly the most common method for solving least squares problems efficiently. Suppose a linear system $\mathbf{Ax} = \mathbf{b}$. If \mathbf{A} is a symmetric positive definite matrix, there exists a unique lower triangular matrix \mathbf{G} , with positive diagonal entries, such that $\mathbf{A} = \mathbf{GG}^T$. After computing the Cholesky factorization, the following triangular systems are solved

$$\mathbf{Gy} = \mathbf{b} \quad \text{and} \quad \mathbf{G}^T \mathbf{x} = \mathbf{y}.$$

These systems are efficiently solved by the iterative process of forward (for lower triangular matrices) and backward (upper triangular matrices) substitution. Applying this to the (stabilized) least squares problem leads to faster computational times then implementations of the classical Gaussian elimination.

This chapter discuss different methods for solving linear Toeplitz systems efficiently, including numerical experiments. Working with a high-level matrix computation language (MATLAB), much of the underlying structure is hidden and detailed documentation on built-in functions may not be easily available to the user. Generalizations in the built-in algorithms could cause speed-ups or slowdowns unknown to the user; a commonly used operator such as the backslash operator¹ includes routines for checking the structure of a matrix before choosing the fastest method. Although documentation is available (Anderson et al., 1999), a study of all the technical details for a specific algorithm can be a very time consuming task.

Another important aspect is the development within the field of computer science. Recently, geoscientists have started to move from conventional CPUs (central processing unit) to GPUs (graphics processing unit). These processing units deliver huge computing power, sometimes several orders of magnitude larger, but at the expense of specialization in code development. With these elements in mind, a set of conventional approaches for solving large-scale linear problems, are presented in this chapter.

¹Solving a linear system $\mathbf{Ax} = \mathbf{b}$ by the MATLAB syntax $\mathbf{x} = \mathbf{A}\backslash\mathbf{b}$.

1. Toeplitz Structure

Linear systems of the form $\mathbf{T}\mathbf{x} = \mathbf{b}$, where the matrix \mathbf{T} has Toeplitz structure, arise in solutions to differential and integral equations, spline functions, problems and methods in physics, mathematics, statistics, and signal processing. Originally named after the German mathematician Otto Toeplitz, matrices whose entries are constant along each diagonal are called Toeplitz matrices.

DEFINITION (Toeplitz matrix). *Let $\mathbf{T} \in \mathbb{R}^{M \times M}$ matrix. The matrix is Toeplitz if there exist scalars $t_{-M+1}, \dots, t_0, \dots, t_{M-1}$ such that $r_{mn} = t_{n-m}$ for all m and n . Thus, with $M = 4$,*

$$\mathbf{T} = \begin{pmatrix} t_0 & t_1 & t_2 & t_3 \\ t_{-1} & t_0 & t_1 & t_2 \\ t_{-2} & t_{-1} & t_0 & t_1 \\ t_{-3} & t_{-2} & t_{-1} & t_0 \end{pmatrix}$$

is known as a Toeplitz matrix. Suppose $\mathbf{T} \in \mathbb{C}^{M \times M}$, this class of Toeplitz matrices can often be referred to as having Hermitian Toeplitz structure. In that case the strictly lower triangular part is the complex conjugate of the strictly upper triangular part, hence with $M = 4$,

$$\mathbf{T} = \begin{pmatrix} t_0 & t_1 & t_2 & t_3 \\ t_1^* & t_0 & t_1 & t_2 \\ t_2^* & t_1^* & t_0 & t_1 \\ t_3^* & t_2^* & t_1^* & t_0 \end{pmatrix}.$$

where t_m^* denotes the complex conjugate.

A Toeplitz matrix is completely determined by $2M - 1$ entries, this property eliminates the need for explicit matrix storage. Utilizing its structure has led to discoveries of several fast and stable methods for finding inverse solutions.

Solving a linear system by straightforward Gaussian elimination involves $\mathcal{O}(M^3)$ operations, it is desirable to use faster algorithms, often referred to as Toeplitz solvers. The Levinson algorithm (Golub and Van Loan, 1989) solves a symmetric positive definite Toeplitz system with an arbitrary right-hand side, reducing the number of operations to $\mathcal{O}(4M^2)$. A complete outline of the Levinson algorithm is given in Appendix B, Algorithm 3. Note that a modification, introducing complex conjugates, is required for Hermitian Toeplitz systems. A MATLAB implementation for Hermitian Toeplitz systems is included in Appendix B, Listing B.1.

An explicit inverse operator for a Toeplitz system can be calculated in $\mathcal{O}(M^2)$ flops by the Trench algorithm (Golub and Van Loan, 1989), but even faster algorithms with $\mathcal{O}(M \log^2 M)$ complexity have been developed (Chan and Ng, 1996). Unfortunately break-downs still occur with ill-conditioned systems and regularization has to be taken into account. Look-ahead variants of the Levinson recursion exists, but preconditioned iterative methods have gained more attention as the number of operations can be reduced to $\mathcal{O}(M \log M)$ by circulant preconditioners, and the fast Fourier transform.

1.1. Eigenvalues of Toeplitz matrices. A great deal is known about the behavior of Toeplitz matrices, the most common and complete reference being Grenander and Szegö (1958). More accessible references, directed at the applied mathematician, is found in Gray (2006); Chan and Jin (2007), including simplified proofs for important theorems from the original work by Grenander and Szegö. The most common result describing Toeplitz matrices is a theorem for sequences of Toeplitz matrices, dealing with the behavior of the eigenvalues as M goes to infinity. Utilizing this theoretical framework is of significant value when analyzing a specific problem.

Following Chan and Jin (2007), the concept of generating functions is presented. Suppose a Toeplitz matrix represented by the diagonals $\{t_m\}_{m=-M-1}^{M-1}$. Then, formally, this is the Fourier coefficients

$$(28) \quad t_m = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\xi) e^{-im\xi} d\xi,$$

where $f(\xi)$ is called the generating function, given by

$$f(\xi) = \sum_{m=-\infty}^{\infty} t_m e^{im\xi}.$$

Let $\mathcal{C}_{2\pi}$ be a space of 2π -periodic continuous real-valued functions defined on $[-\pi, \pi]$. A generating function f with these properties is then used with the following theorem, giving upper and lower bounds to eigenvalues and determining their distribution.

THEOREM 3 (Grenander-Szegö theorem, Chan and Jin (2007)). *Suppose a Toeplitz matrix \mathbf{T} with a generating function $f \in \mathcal{C}_{2\pi}$. Let λ_{\min} and λ_{\max} denote the smallest and largest eigenvalue of \mathbf{T} , respectively. Then*

$$f_{\min} \leq \lambda_{\min} \leq \lambda_{\max} \leq f_{\max},$$

where f_{\min} and f_{\max} denote the minimum and maximum values of $f(x)$, respectively. In particular, if $f_{\min} > 0$, then \mathbf{T} is positive definite. Moreover, the eigenvalues λ_j , $j = 0, 1, \dots, M - 1$, are equally distributed as $f(2\pi j/M)$, i.e.,

$$\lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=0}^{M-1} [g(\lambda_j) - g(f(2\pi j/M))] = 0$$

for any $g \in \mathcal{C}_{2\pi}$.

Uniform convergence of the Fourier series $f(\xi)$ is ensured if the coefficients t_m are absolutely summable

$$(29) \quad \sum_{m=-\infty}^{\infty} |t_m| < \infty.$$

If this equation holds, the function f is Riemann integrable and is said to be in the Wiener class.

1.2. Example. To illustrate Theorem 3, the eigenvalues of a tridiagonal (one-dimensional discrete Laplacian) matrix are evaluated. Suppose a generating function f on the form

$$f(\xi) = -\cos(-\xi) + 2 - \cos(\xi) = 4 \sin^2(\xi/2).$$

From Equation (28), the diagonals of a Toeplitz matrix with f as a generating function, is given as

$$t_m = \frac{2}{\pi} \int_{-\pi}^{\pi} \sin^2(\xi/2) e^{-im\xi} d\xi = \frac{2}{\pi} \left(\frac{\sin(\pi m)}{m} + \frac{m \sin(\pi m)}{m^2 - 1} \right)$$

Using l'Hôpital's rule, the diagonals t_0 and $t_{\pm 1}$ are, respectively

$$\lim_{m \rightarrow 0} t_m = \lim_{m \rightarrow 0} 2 \cos(\pi m) = 2, \quad \lim_{m \rightarrow \pm 1} t_m = \lim_{m \rightarrow \pm 1} \frac{\sin(\pi m) + m \cos(\pi m)}{m} = -1.$$

For all other values $m = \pm 2, \pm 3, \dots$ the coefficients $t_m = 0$. The framework is established with a 2π -periodic generating function f and its Fourier coefficients t_m . From the Grenander-Szegő theorem it is possible to determine the distribution of eigenvalues λ_j for $j = 0, 1, \dots, M - 1$.

In Figure, 1 the Toeplitz matrix \mathbf{T} is demonstrated using $M = 32$ samples. The generating function $f(\xi) = 4 \sin^2(\xi/2)$ on $[-\pi, \pi]$ has a maximum at $f_{\max} = 4$ and a minimum $f_{\min} = 0$, such that the eigenvalues λ_j are restricted to $0 \leq \lambda_{\min} \leq \lambda_{\max} \leq 4$. The discrete sequence in Figure 1(b) shows the numerically calculated eigenvalues plotted by index j .

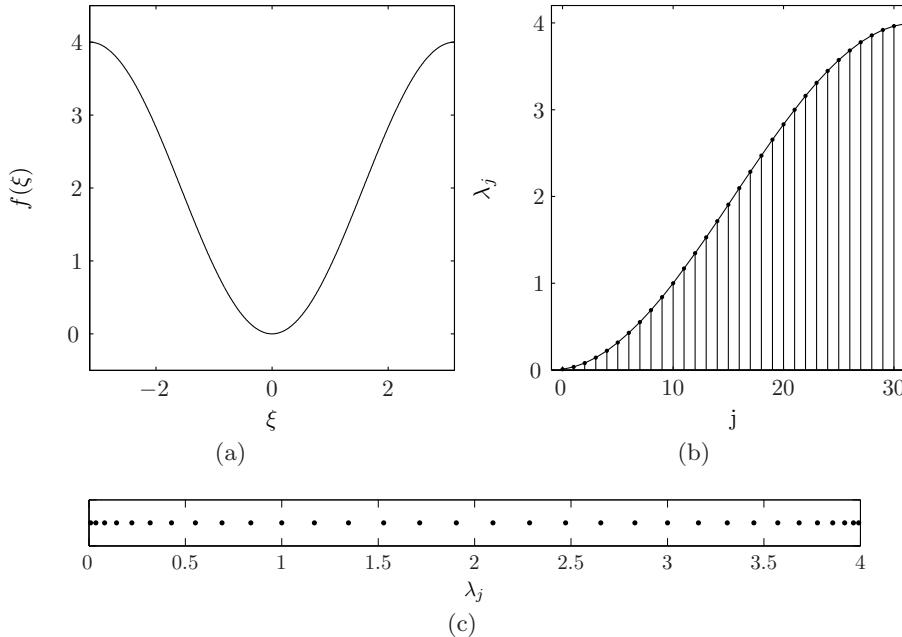


Figure 1: Eigenvalues of a tridiagonal Toeplitz matrix. a) The generating function $f(\xi) = 4 \sin^2(\xi/2)$. b) Numerically calculated eigenvalues of \mathbf{T} . c) One-dimensional plot of the eigenvalue distribution.

2. Quadratic Forms and the Conjugate Gradient Method

Iterative solvers may require several technical conditions to be satisfied to guarantee convergence. The method of conjugate gradient (Hestenes, 1973) is an iterative algorithm for systems with the property of being positive definite. Studying its concepts is a good introduction to mathematical optimization theory, and gives new perspectives on how linear systems can be solved. A central part of the conjugate gradient (CG) algorithm, is the method of gradient descent (often referred to as method of steepest descent), backed up by quadratic forms. An introduction to these algorithms is presented in this section.

Suppose a quadratic form $f : \mathbb{R}^M \rightarrow \mathbb{R}$ as

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{A}\mathbf{x} - \mathbf{b}^T \mathbf{x} + c,$$

where $\mathbf{b} \in \mathbb{R}^M$, c is a constant and the matrix $\mathbf{A} \in \mathbb{R}^{M \times M}$. From this equation, it can be shown that $f(\mathbf{x})$ is minimized by the solution to $\mathbf{Ax} = \mathbf{b}$ when \mathbf{A} is symmetric and positive definite.

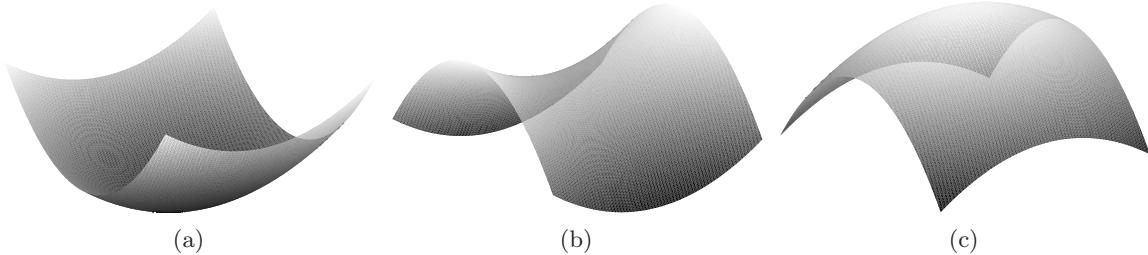


Figure 2: Different classification of quadratic forms. a) Positive definite by equation $f(x, y) = 3x^2 + 7y^2$. b) Indefinite by equation $f(x, y) = 3x^2 - 7y^2$. c) Negative definite by equation $f(x, y) = -3x^2 - 7y^2$.

2.1. The method of gradient descent. To derive the conjugate gradient method, the first step is to introduce the theory of gradient descent. The gradient of the quadratic form is

$$\nabla f(\mathbf{x}) = \frac{1}{2}\mathbf{A}^T \mathbf{x} + \frac{1}{2}\mathbf{Ax} - \mathbf{b},$$

which by symmetry in \mathbf{A} , is reduced to

$$\nabla f(\mathbf{x}) = \mathbf{Ax} - \mathbf{b}.$$

Solving $\nabla f(\mathbf{x}) = \mathbf{0}$ gives a stationary point (and solution to the inverse problem), representing a local minimum for a positive definite system. Positive definiteness is defined as $\mathbf{x}^T \mathbf{A}\mathbf{x} > 0$, but it is also possible to evaluate this property by looking at the operators eigenvalues, i.e. a symmetric matrix is positive definite if and only if its eigenvalues are all positive.

Following Shewchuk (1994), an intuitive understanding of the gradient descent algorithm is best described in three-dimensions by thinking of it as sliding down a positive definite paraboloid, starting from an arbitrary point \mathbf{x}_0 and then continue iterating until reaching the bottom at some \mathbf{x}_k , which gives the solution to the linear system. The iterative procedure is given as

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \nabla f(\mathbf{x}_k), \quad k = 0, 1, 2, \dots$$

where α is the step size. Choosing the α which minimizes f along a line is called a line search. This is done by differentiating $f(\mathbf{x}_{k+1})$ with respect to α , and finding its minimum

$$\frac{d}{d\alpha} f(\mathbf{x}_{k+1}) = \nabla f(\mathbf{x}_{k+1})^T \frac{d}{d\alpha} \mathbf{x}_k = -\nabla f(\mathbf{x}_{k+1})^T \nabla f(\mathbf{x}_k) = \mathbf{0}.$$

This implies that α must be chosen such that the gradients $\nabla f(\mathbf{x}_{k+1})$ and $-\nabla f(\mathbf{x}_k)$ are orthogonal. Golub and Van Loan (1989) remark that the function f decreases most rapidly in the direction of the negative gradient

$$\mathbf{r}_k = -\nabla f(\mathbf{x}_k) = \mathbf{b} - \mathbf{A}\mathbf{x}_k.$$

To isolate α and get an explicit expression, further algebraic manipulations are needed

$$\mathbf{r}_{k+1}^T \mathbf{r}_k = (\mathbf{b} - \mathbf{A}\mathbf{x}_{k+1})^T \mathbf{r}_k = (\mathbf{b} - \mathbf{A}(\mathbf{x}_k + \alpha \mathbf{r}_k))^T \mathbf{r}_k.$$

This gives the exact line search as

$$\alpha = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k}.$$

The gradient descent method now consists of a initialization step where an initial guess of x is done, before the residual is calculated. The recursive part then consists of three steps: 1) Calculate the exact line search based on the residual vectors. 2) Calculate the new point \mathbf{x}_{k+1} . 3) Recalculate the new residual vector. The complete procedure is outlined in Appendix B, Algorithm 2.

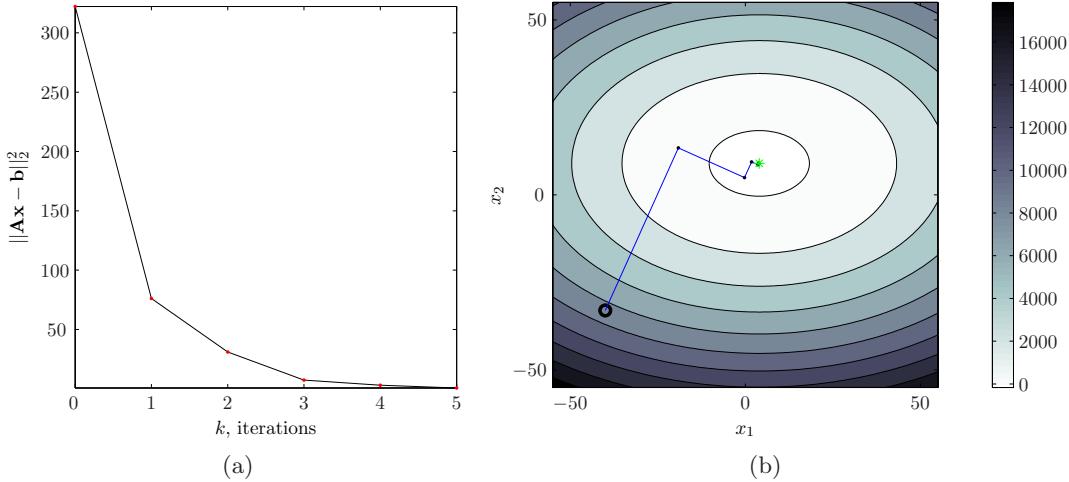


Figure 3: The method of gradient descent. a) Residual norm after k iterations. b) Contour plot of the positive definite linear system.

To illustrate the gradient descent method, a simple three-dimensional example is presented. The sample problem is chosen as

$$\mathbf{A} = \begin{pmatrix} 3 & 0 \\ 0 & 7 \end{pmatrix} \quad \text{and} \quad \mathbf{b} = \begin{pmatrix} 12 \\ 63 \end{pmatrix}.$$

This gives the following quadratic form

$$f(\mathbf{x}) = \frac{1}{2} (3x_1^2 + 7x_2^2) - (12x_1 + 63x_2),$$

shown as a contour plot in Figure 3(b). Initializing with $\mathbf{x}_0 = [-40, -33]^T$ (black circle), Figure 3(a) shows how the norm of the residuals decrease when \mathbf{x}_k , converges against the true solution $\mathbf{x} = [4, 9]^T$. From the illustration in 3(b), it is shown how each step (blue lines) is orthogonal to the previous, and how the step size decrease.

A thorough investigation on the convergence properties of the gradient descent method is presented in Shewchuk (1994) by use of eigenvectors and eigenvalues. Further progress against understanding the complete CG algorithm requires a definition of conjugate directions.

2.2. The conjugate gradient method. The gradient descent method often steps in the same direction. It would be desirable to only step in the same direction once. Selecting a new set of orthogonal search directions $\{\mathbf{p}_1, \dots, \mathbf{p}_M\}$, such that

$$\mathbf{p}_i^T \mathbf{A} \mathbf{p}_j = 0, \quad i \neq j$$

is the ideal choice for this purpose. A sequence of vectors with this property is called mutually conjugate with respect to \mathbf{A} , also known as \mathbf{A} -conjugate or \mathbf{A} -orthogonal, this sequence consists of M mutually conjugate directions. A sequence $\{\mathbf{p}_k\}$ forms a basis for \mathbb{R}^M where the solution of $\mathbf{Ax} = \mathbf{b}$ can exist:

$$\tilde{\mathbf{x}} = \sum_{i=1}^M \alpha_i \mathbf{p}_i.$$

Assuming the new search directions are known, the procedure of finding α_k is straightforward by multiplying with $\mathbf{p}_k^T \mathbf{A}$

$$\mathbf{p}_k^T \mathbf{A} \tilde{\mathbf{x}} = \mathbf{p}_k^T \mathbf{b} = \sum_{i=1}^M \alpha_i \mathbf{p}_k^T \mathbf{A} \mathbf{p}_i = \alpha_k \mathbf{p}_k^T \mathbf{A} \mathbf{p}_k,$$

and finding the corresponding step size in that specific direction

$$\alpha_k = \frac{\mathbf{p}_k^T \mathbf{b}}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k}.$$

By these assumptions, the iteration is given as

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k,$$

with the residuals

$$\begin{aligned} \mathbf{r}_{k+1} &= \mathbf{b} - \mathbf{A} \mathbf{x}_{k+1} \\ &= \mathbf{b} - \mathbf{A}(\mathbf{x}_k + \alpha_k \mathbf{p}_k) \\ &= (\mathbf{b} - \mathbf{A} \mathbf{x}_k) - \alpha_k \mathbf{A} \mathbf{p}_k \\ &= \mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k. \end{aligned}$$

The next step to complete the CG algorithm, is to construct the basis vectors \mathbf{p}_k . This is done in an iterative manner (as the CG algorithm progresses) using a method similar to the Gram-Schmidt process (Lay, 2006). The formalism for deriving the mutually conjugate basis vectors differs in the literature (Golub and Van Loan, 1989; Shewchuk, 1994; Aster et al., 2005), and is well understood. For a complete derivation a fair bit of algebraic manipulations, not covered in the present work, are required. In theory, the algorithm will find an exact solution to the linear system in M iterations, however numerical round-off errors could produce problems and it may require more. In practice, the algorithm will run until some tolerance limit for convergence is reached. Using the same sample system as in the previous example, the CG method is demonstrated in Figure 4.

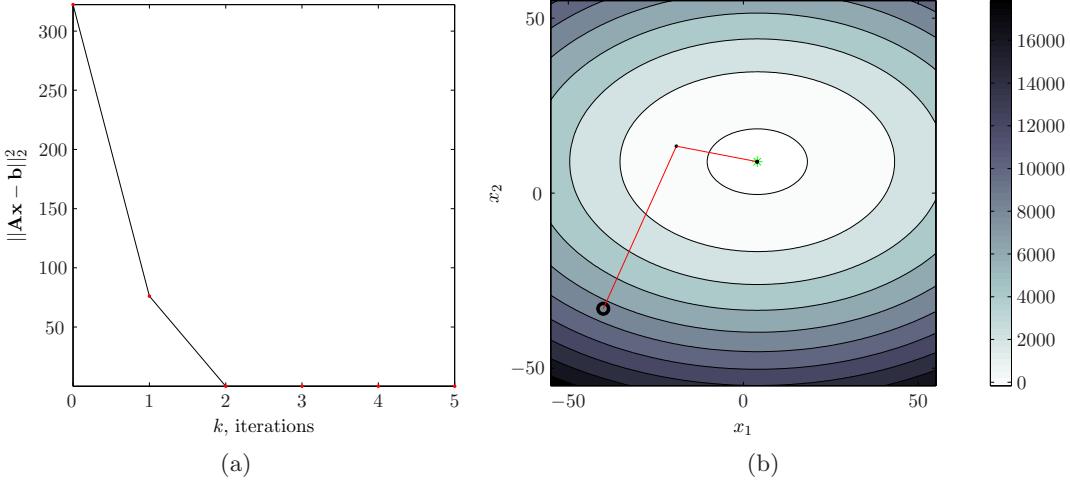


Figure 4: The method of conjugate gradient. a) Residual norm after k iterations. b) Contour plot of the positive definite linear system.

2.3. The complex biconjugate gradient method. In the preceding derivations, the conjugate gradient method was derived for real positive definite and symmetric systems. For asymmetric systems, the biconjugate gradient method (Fletcher, 1975) has been developed. This method is generalized to a complex form called the complex biconjugate gradient method (Jacobs, 1986; Joly and Meurant, 1993), a generalization of both biconjugate gradient and conjugate gradient.

Suppose a system $\mathbf{A}\mathbf{z} = \mathbf{b}$, with $\mathbf{A} \in \mathbb{C}^{M \times M}$, $\mathbf{b} \in \mathbb{C}^M$ and an initial estimate \mathbf{z}_0 . Following Jacobs (1986), define a residual $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{z}_0$ and an initial search direction $\mathbf{p}_0 = \mathbf{r}_0$. In parallel, define a biresidual $\hat{\mathbf{r}}_0 = \mathbf{r}_0^*$ and a bidirection $\hat{\mathbf{p}}_0 = \mathbf{r}_0^*$, where $*$ denotes the complex conjugate. With the inner product denoted $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^H \mathbf{y}$, the step length is given as

$$\alpha_k = \frac{\langle \hat{\mathbf{r}}_k, \mathbf{r}_k \rangle}{\langle \hat{\mathbf{p}}_k, \mathbf{A}\mathbf{p}_k \rangle},$$

with a new solution obtained by $\mathbf{z}_{k+1} = \mathbf{z}_k + \alpha_k \mathbf{p}_k$. The new residual and biresidual are

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k, \quad \hat{\mathbf{r}}_{k+1} = \hat{\mathbf{r}}_k - \alpha_k^* \mathbf{A}^H \hat{\mathbf{p}}_k,$$

and the biconjugacy coefficient is

$$\beta_k = -\frac{\langle \mathbf{A}^H \hat{\mathbf{p}}_k, \mathbf{r}_{k+1} \rangle}{\langle \hat{\mathbf{p}}_k, \mathbf{A}\mathbf{p}_k \rangle}.$$

The next direction vector and bidirection vector is then

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k, \quad \hat{\mathbf{p}}_{k+1} = \hat{\mathbf{r}}_{k+1} + \beta_k^* \hat{\mathbf{p}}_k.$$

With these generalizations, the complete method is presented in Algorithm 1.

ALGORITHM 1 (Complex Biconjugate Gradient Method). *Given a system of equations $\mathbf{Az} = \mathbf{b}$, where the matrix $\mathbf{A} \in \mathbb{C}^{M \times M}$ and $\mathbf{b} \in \mathbb{C}^M$. Initialize the following variables $\mathbf{r}_0 = \mathbf{b} - \mathbf{Az}_0$, $\mathbf{p}_0 = \mathbf{r}_0$, $\hat{\mathbf{r}}_0 = \mathbf{r}_0^*$, $\hat{\mathbf{p}}_0 = \mathbf{p}_0^*$ and $k = 0$, then begin the iteration:*

```

while  $\mathbf{r}_k \neq \mathbf{0}$  do
     $\alpha_k = \langle \hat{\mathbf{r}}_k, \mathbf{r}_k \rangle / \langle \hat{\mathbf{p}}_k, \mathbf{A}\mathbf{p}_k \rangle$ 
     $\mathbf{z}_{k+1} = \mathbf{z}_k + \alpha_k \mathbf{p}_k$ 
     $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k$ 
     $\hat{\mathbf{r}}_{k+1} = \hat{\mathbf{r}}_k - \alpha_k^* \mathbf{A}^H \hat{\mathbf{p}}_k$ 
     $\beta_k = -\langle \mathbf{A}^H \hat{\mathbf{p}}_k, \mathbf{r}_{k+1} \rangle / \langle \hat{\mathbf{p}}_k, \mathbf{A}\mathbf{p}_k \rangle$ 
     $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$ 
     $\hat{\mathbf{p}}_{k+1} = \hat{\mathbf{r}}_{k+1} + \beta_k^* \hat{\mathbf{p}}_k$ 
     $k = k + 1$ 
end while

```

The main holdup in the algorithm is the matrix-vector multiplication. However, by exploiting the matrix structure further improvements can be done. The CG method can be used directly on the least squares problem by applying it to the normal equations. The system still has to be well-conditioned to obtain a stable solution, i.e. applying the CG algorithm to a Tikhonov regularized problem is a common approach if the system is ill-conditioned, although more elegant methods using preconditioners have been developed.

3. Preconditioned Conjugate Gradient

The conjugate gradient method applies to matrices that are either well-conditioned or have just a few distinct eigenvalues. If this is not the case, it is possible to precondition a linear system to take one of these forms. The preconditioned conjugate gradient (PCG) involves applying the CG method to the transformed system

$$\tilde{\mathbf{T}}\tilde{\mathbf{z}} = \tilde{\mathbf{b}},$$

where $\tilde{\mathbf{T}} = \mathbf{C}^{-1}\mathbf{T}\mathbf{C}^{-1}$, $\tilde{\mathbf{z}} = \mathbf{C}\mathbf{z}$, $\tilde{\mathbf{b}} = \mathbf{C}^{-1}\mathbf{b}$ and \mathbf{C} is a symmetric positive definite matrix, known as a preconditioner. The preconditioner is chosen with respect to the problem, i.e. the matrix $\tilde{\mathbf{T}}$ should be well-conditioned or have clustered eigenvalues. The use of PCG with circulant matrices as preconditioners for solving Toeplitz systems, was proposed independently by Strang (1986) and Olkin (1986). It resulted in an efficient inversion method, using the fast Fourier transform.

A preconditioner is used both for well- and ill-conditioned systems, having the effect of promoting convergence and stabilizing the problem. A wide range of Toeplitz preconditioners have been developed (Chan and Jin, 2007), where many of the more advanced are based on the framework by Grenander and Szegő (1958). The most straightforward approach is to use an existing Toeplitz matrix \mathbf{T} , without knowledge of the generating function f . One of the first preconditioners, called the optimal circulant preconditioner (Chan, 1988), is derived using only the Toeplitz coefficients (with a finite range). This is presented in the next section.

4. Circulant Preconditioning of Toeplitz Matrices

Circulant matrices are a natural choice as preconditioner for Toeplitz matrices because they are inexpensive to construct and use (Hansen, 2002), they are also classified as a special class of Toeplitz matrices.

DEFINITION (Circulant Matrix). *Let $\mathbf{C} \in \mathbb{C}^{M \times M}$ matrix with the form*

$$\mathbf{C} = \begin{pmatrix} c_0 & c_{M-1} & c_{M-2} & \cdots & c_1 \\ c_1 & c_0 & c_{M-1} & \cdots & c_2 \\ c_2 & c_1 & c_0 & \cdots & c_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_{M-1} & c_{M-2} & c_{M-3} & \cdots & c_0 \end{pmatrix},$$

known as a circulant matrix, with the property of being determined by its first column.

The optimal circulant approximation (preconditioner) or T. Chan's preconditioner (Chan, 1988) is presented in this section.

Assume that $\mathbf{T} \in \mathbb{C}^{M \times M}$ is a Hermitian Toeplitz matrix where the t_0 diagonal is real. The preconditioner is defined to be the circulant matrix $\mathbf{C} \in \mathbb{C}^{M \times M}$ which minimizes the ℓ_2 -norm

$$e = \|\mathbf{T} - \mathbf{C}\|_2^2$$

amongst all circulant matrices. The ℓ_2 -norm of a matrix is defined as

$$\|\mathbf{T}\|_2^2 = \sum_{m,n} |t_{m,n}|^2.$$

The residual error e , can be written

$$e = M|t_0 - c_0|^2 + \sum_{m=1}^{M-1} (M-m) \{ |t_m - c_m|^2 + |t_m^* - c_{M-m}|^2 \},$$

or changing index in the last part of the sum to $(M-m) = m$, the result is

$$e = M|t_0 - c_0|^2 + \sum_{m=1}^{M-1} \{ (M-m)|t_m - c_m|^2 + m|t_{M-m}^* - c_m|^2 \}.$$

Minimizing e , by $\partial e / \partial c_m = 0$, yields

$$c_m = \frac{1}{M} [(M-m)t_m + mt_{M-m}^*].$$

The circulant preconditioner could be derived in a more general manner; based on the framework which was presented in the discussion on Toeplitz matrices and eigenvalues. With convolution of the generating function and some given kernel, it is possible to obtain the optimal preconditioner (and a range of others). Details are covered in Chan and Jin (2007).

4.1. Circulant preconditioner for ill-conditioned Hermitian Toeplitz systems. Although the main focus in the present work is the optimal circulant approximation, some of the numerical exercises include results from a more modern circulant approximation. This approximation will be referred to as a preconditioner from the generalized Jackson kernel (Chan and Jin, 2007). Modern theory on circulant approximations discuss preconditioners with respect to a generating function $f(\xi)$. For certain generating functions (classified by the number of zeros), it has been proved that the optimal circulant approximations fail to stabilize the system (Chan et al., 2002). It is possible to construct preconditioners for ill-conditioned systems when the generating function is known (Chan et al., 2001a), but when the function is unknown or not in the Wiener class, a different approach is needed.

To tackle an ill-conditioned problem with an unknown generating function (which is the case in the present work), Chan et al. (2001b) propose to construct a preconditioner by approximating $f(\xi)$. This is done using the convolution product $\mathcal{K} * f$, where \mathcal{K} is the generalized Jackson kernel (Lorentz, 2005).

For very ill-conditioned problems, even a preconditioner based on the generalized Jackson kernel may require regularization of the Toeplitz operator in advance. If this is the case, preconditioners serve a purpose, promoting convergence of the iterative Toeplitz solver. For the optimal circulant preconditioner, a MATLAB program is included in Appendix B, Listing B.6. In Chan and Jin (2007), a range of different preconditioners are easily available, including MATLAB code.

5. Matrix-vector Multiplication by the Fast Fourier Transform

The iterative methods presented in this chapter are dominated by matrix-vector multiplications. In general, a matrix has to be formed in memory and $\mathcal{O}(M^2)$ operations is performed to compute a new vector. Techniques have been discovered for Toeplitz matrices (Strang, 1986), reducing the number of operations substantially. Suppose a Toeplitz matrix \mathbf{T} with $M = 3$

$$\mathbf{b} = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} t_0 & t_{-1} & t_{-2} \\ t_1 & t_0 & t_{-1} \\ t_2 & t_1 & t_0 \end{pmatrix} \begin{pmatrix} z_0 \\ z_1 \\ z_2 \end{pmatrix} = \mathbf{T}\mathbf{z}.$$

Instead of using the conventional method, the multiplication $\mathbf{T}\mathbf{z}$ could be carried out by constructing a $(2M - 1) \times (2M - 1)$ circulant matrix $\widehat{\mathbf{T}}$ on the form

$$\widehat{\mathbf{b}} = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ \vdots \end{pmatrix} = \begin{pmatrix} t_0 & t_{-1} & t_{-2} & t_2 & t_1 \\ t_1 & t_0 & t_{-1} & t_{-2} & t_2 \\ t_2 & t_1 & t_0 & t_{-1} & t_{-2} \\ t_{-2} & t_2 & t_1 & t_0 & t_{-1} \\ t_{-1} & t_{-2} & t_2 & t_1 & t_0 \end{pmatrix} \begin{pmatrix} z_0 \\ z_1 \\ z_2 \\ 0 \\ 0 \end{pmatrix} = \widehat{\mathbf{T}}\widehat{\mathbf{z}}.$$

When embedding a Toeplitz matrix into a circulant matrix, the system size is larger, but the formulation now has a computational advantage which can be accredited the fast Fourier transform.

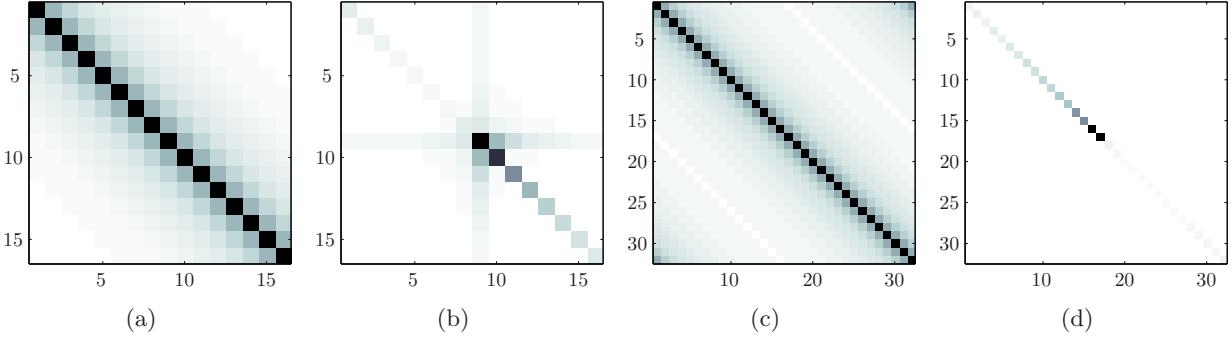


Figure 5: Toeplitz and circulant matrices and their corresponding spectrum. a) Toeplitz matrix \mathbf{T} . b) Diagonalization of \mathbf{T} by the Fourier basis. c) Circulant matrix $\widehat{\mathbf{T}}$ (embedded Toeplitz). d) Diagonalization of $\widehat{\mathbf{T}}$ by the Fourier basis.

Diagonalization of the circulant matrix $\widehat{\mathbf{T}}$ is done by the Fourier basis \mathbf{W} , such that

$$\widehat{\mathbf{T}} = \mathbf{W}^H \boldsymbol{\Lambda} \mathbf{W},$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix, holding the eigenvalues of $\widehat{\mathbf{T}}$. Denoting elementwise product as \odot , the matrix-vector multiplication $\widehat{\mathbf{T}}\widehat{\mathbf{z}}$ can be written

$$\text{diag}(\boldsymbol{\Lambda}) \odot \mathbf{W}\widehat{\mathbf{z}} = \text{diag}(\boldsymbol{\Lambda}) \odot \mathcal{F}\{\widehat{\mathbf{z}}\} = \mathcal{F}\{\widehat{\mathbf{b}}\}$$

In practice, only one column of the circulant matrix is formed and a fast Fourier transform is used for calculating the eigenvalues. The need for matrix storage is eliminated and the number of operations are reduced to $\mathcal{O}[(2M - 1)\log(2M - 1)]$. From this, it is clear that circulant matrices (e.g. the circulant approximation) are efficiently invertible using the FFT.

5.1. Preconditioned conjugate gradient, an example. To illustrate the theory of PCG, a full rank problem is analyzed using the preceding fast computational methods. Suppose a Toeplitz system $\mathbf{T}\mathbf{x} = \mathbf{b}$ with coefficients t_m given as

$$(30) \quad t_m = \begin{cases} (1+i)/(1+m)^{1.1}, & m > 0 \\ 2, & m = 0 \\ t_{-m}^*, & m < 0 \end{cases}.$$

According to Chan and Jin (2007), the coefficients have an underlying generating function f , given by

$$f(\xi) = 2 \sum_{m=0}^{\infty} \frac{\sin(m\xi) + \cos(m\xi)}{(1+m)^{1.1}}.$$

With a tolerance limit of $\epsilon = 10^{-7}$ and \mathbf{b} initialized as ones, the linear system $\mathbf{T}\mathbf{x} = \mathbf{b}$ is solved by the preconditioned conjugate gradient method. Table 1 lists the results for different M . The preconditioners are chosen either as the identity matrix \mathbf{I} , or the optimal preconditioner \mathbf{C} . After preconditioning $\mathbf{C}^{-1}\mathbf{T}$ the eigenvalues are clustered around one, thus promoting convergence. Their distribution is illustrated in Figure 6. From this exercise it is demonstrated how preconditioning promotes convergence and how large linear systems can be solved efficiently using the FFT. In-memory storage is also an asset since it is no need to form any explicit matrices. Figure 5 shows the matrices involved in the computation when $M = 16$.

M	Iterations		CPU-time	
	C	I	C	I
2^4	7	12	n/a	n/a
2^{10}	8	22	0.01 s	0.01 s
2^{12}	8	23	0.03 s	0.07 s
2^{13}	8	23	0.06 s	0.18 s
2^{18}	8	22	3.78 s	9.97 s

Table 1: Solving the linear system $\mathbf{T}\mathbf{x} = \mathbf{b}$ using the preconditioned conjugate gradient method. M denotes the matrix size while **I** and **C** denotes the identity and circulant preconditioner, respectively.

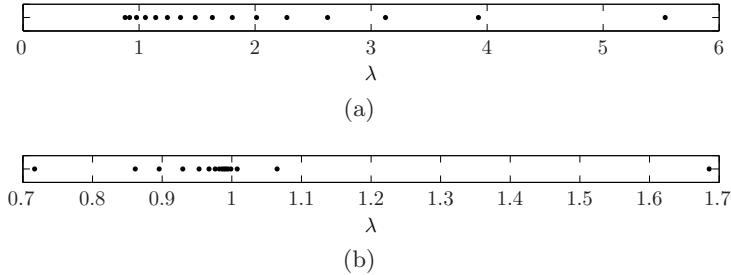


Figure 6: a) Eigenvalues of the original Toeplitz system. b) Preconditioned system $\mathbf{C}^{-1}\mathbf{T}$.

6. Numerical Results

The different methods, presented in this chapter, are tested with the parabolic Radon transform. Motivated by the computational aspects, methods have been compared with respect to the number of operations required to reach convergence and CPU-time. The latter is a common (empirical) measure of algorithm efficiency.

6.1. Computational aspects. It is important to minimize the number of matrix operations when evaluating efficient implementations of the Radon transform. Using the discretization scheme presented in Chapter 1, no explicit matrices are formed in memory. With the least squares formulation of the parabolic Radon transform, $\mathbf{A}^H\mathbf{A}\mathbf{z} = \mathbf{A}^H\mathbf{b}$, the right hand side is evaluated by

$$[\mathbf{A}^H\mathbf{b}]_j = \sum_{n=0}^{N-1} f_n \exp(i\omega x_n^2 q_j),$$

where $j = 0, 1, \dots, M-1$ and f_n is the seismic data for one fixed frequency. The Hermitian Toeplitz matrix $\mathbf{A}^H\mathbf{A} \in \mathbb{C}^{M \times M}$ is given by

$$[\mathbf{A}^H\mathbf{A}]_{j-m} = h_{j-m} = \sum_{n=0}^{N-1} \exp[i\omega x_n^2 (j-m)\Delta q].$$

This matrix is created by keeping either j or m fixed and only building one row or column². If explicit evaluation of the entire matrix is needed, it can easily be built by the MATLAB *toeplitz*

²It is possible to evaluate this expression even faster with the linear Radon transform as an explicit geometric series exists.

method. However, with Toeplitz solvers this is not needed. The fact that these calculations are performed multiple times for a range of frequencies emphasize the importance of optimization. The next step is to choose a linear (Toeplitz) solver. This is also important and can greatly improve the computational speed.

6.2. Benchmarks. In Figure 7, benchmarks (measure of relative performance, computed multiple times and averaged) are plotted for three selected methods: Cholesky factorization, the Levinson algorithm and conjugate gradient. Points represent measured CPU-time in seconds and the lines represent a moving average of the points. For this exercise, the system is defined by $\Delta t = 1/25$, with a period $T = 2.25$. Frequencies range from $1 \leq \nu \leq 12$ and the number of samples in offset is fixed to $N = 122$. A linear transform is computed for each frequency and the problem is stabilized by a regularization parameter $\alpha = 0.1$ (regularization involves one scalar operation, adding α to the first element of the Toeplitz column). It is important to note that the Cholesky factorization is the only method that uses explicit matrices in memory. However, they are formed in a very efficiency manner by just shifting the Toeplitz coefficients (and complex conjugate). Results show that the iterative methods, tailored for Toeplitz systems, does perform better than more conventional methods, especially for large systems.

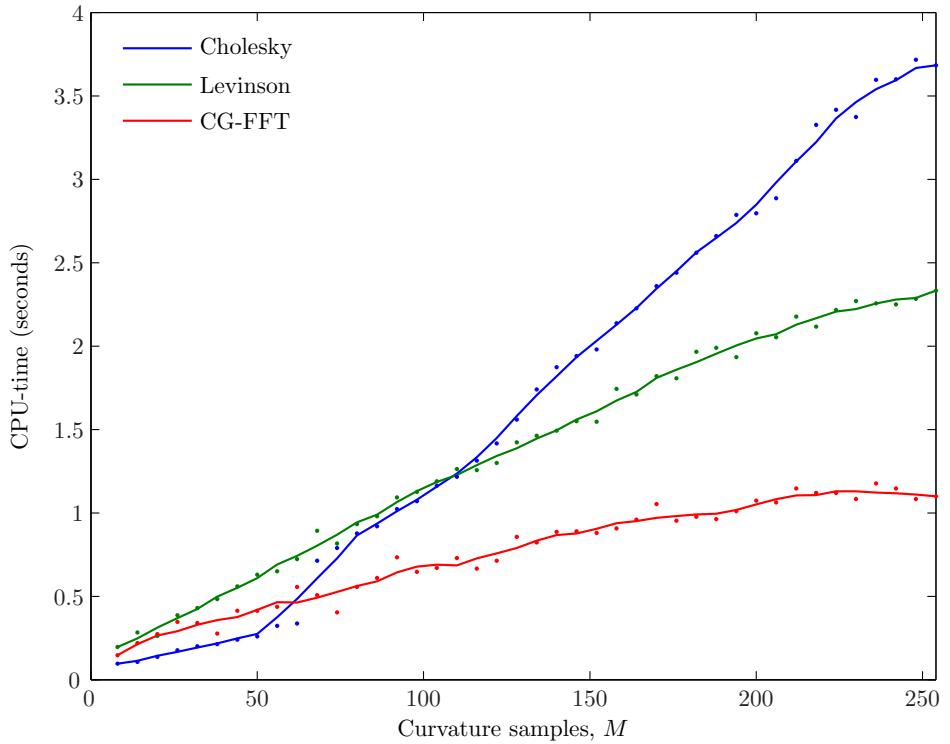


Figure 7: Benchmarks by algorithm CPU-time, plotted for tree selected methods: Cholesky, Levinson and FFT conjugate gradient.

6.3. Stability and condition number. In order to do a systematic numerical exercise, a fixed system is defined and used as a starting point for varying selected parameters. The system is defined by $\Delta t = 1/55$, with a period $T = 2.25$. Frequencies range from $1 \leq \nu \leq 30$ Hz with $N = 35$ samples in offset and $M = 30$ samples for the residual-moveout. Let σ_m denote the singular values (SVD) of a stabilized Toeplitz operator

$$\mathbf{H} = \mathbf{A}^H \mathbf{A} + \alpha \mathbf{I},$$

where α is the regularization parameter. Denoting the condition number κ as the largest singular value σ_{\max} , divided by the smallest σ_{\min} , this is written

$$\kappa_\alpha(\mathbf{H}) = \|\mathbf{H}\|_2 \cdot \|\mathbf{H}^{-1}\|_2 = \sigma_{\max}/\sigma_{\min}.$$

For the fixed sample system, Figure 8 gives an indication of how the condition number is affected by choice of α . Red crosses mark a selected $\hat{\alpha}$, this is the regularization parameter of choice for further analysis. This specific choice stabilizes the solution enough for iterative procedures to succeed, while not introducing too much bias in the solution.

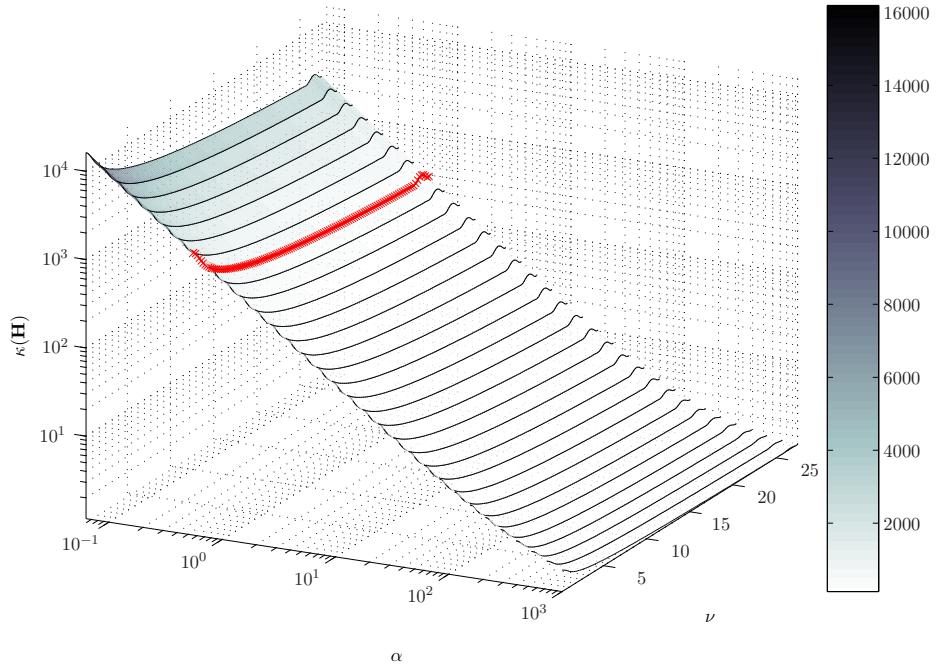


Figure 8: Condition number $\kappa(\mathbf{H})$ for different choices of the regularization parameter α .

For the selected $\hat{\alpha}$, Figure 9(a), shows the distribution of singular values. This can be compared to Figure 9(b), where the values of σ_m are plotted on a logarithmic scale.

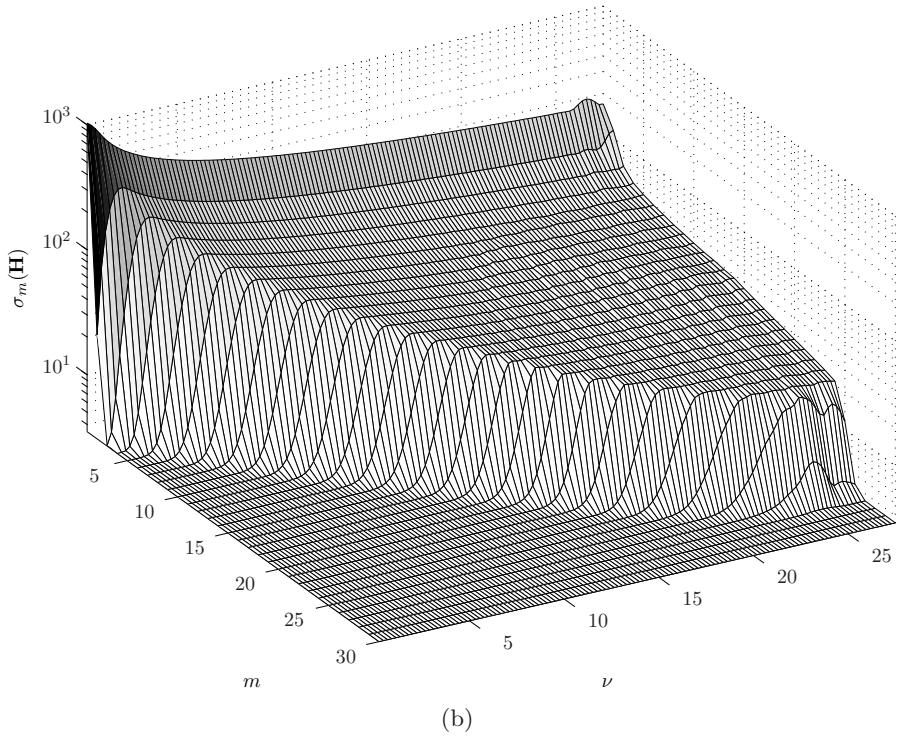
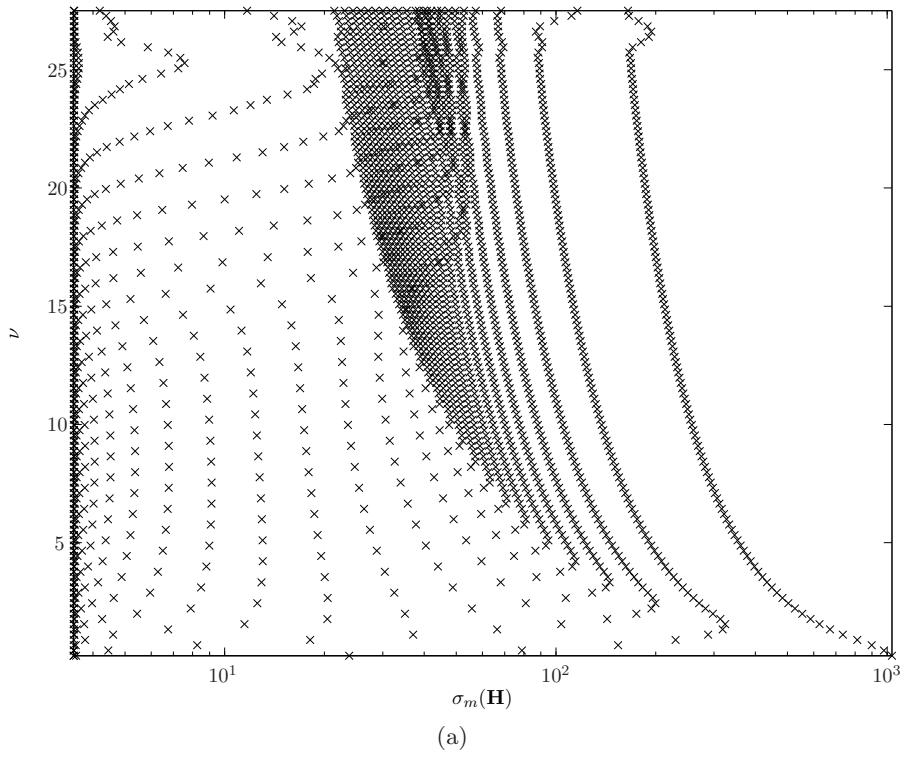


Figure 9: Singular values σ_m , visualized for a selected regularization parameter $\hat{\alpha}$.

6.4. Circulant preconditioning. To increase convergence rate of the conjugate gradient method with Toeplitz systems, circulant preconditioners have been discussed as a suitable choice. This is based on its similarity to the original matrix and computational advantages by the fast Fourier transform. It exists a range of circulant approximations and choosing the correct one depends on the problem at hand. To avoid confusion, the optimal circulant preconditioner is discussed first, in this section. For the preconditioner by the generalized Jackson kernel, results are presented in the end of this chapter.

Let $\mathbf{B} \in \mathbb{C}^{M \times M}$ denote the preconditioned system of the stabilized Toeplitz operator \mathbf{H} . Further, let \mathbf{C} be a circulant preconditioner, such that

$$\mathbf{B} = \mathbf{C}^{-1}\mathbf{H} = \mathbf{C}^{-1}(\mathbf{A}^H\mathbf{A} + \alpha\mathbf{I}).$$

The matrix \mathbf{H} is Hermitian Toeplitz with real eigenvalues (or singular values), however, there is no such guarantee for \mathbf{B} .

In Figure 10, the condition number $\kappa_\alpha(\mathbf{B})$ is evaluated for the preconditioned system. This shows how the preconditioned system responds to different stabilization. From this, it is observed how preconditioning with the optimal preconditioner does not have any radical effect on $\kappa_\alpha(\mathbf{B})$ compared to $\kappa_\alpha(\mathbf{H})$. Preconditioning of a system with $\alpha = 0$ is obviously the preferred approach, but as demonstrated in this figure, smaller α gives a larger κ and the system becomes ill-conditioned for very small α .

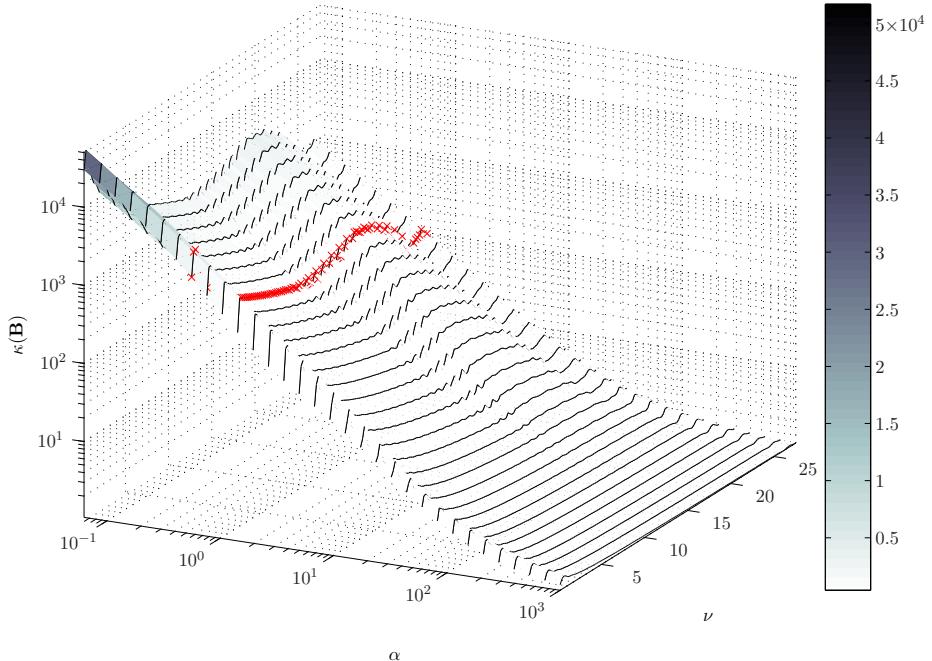


Figure 10: Condition number $\kappa(\mathbf{B})$ for different choices of the regularization parameter α .

6.5. Convergence and the preconditioned conjugate gradient method. By comparing $\kappa_\alpha(\mathbf{B})$ and $\kappa_\alpha(\mathbf{H})$, it is observed that the circulant preconditioner did not have a significant influence on the condition number of the operator \mathbf{B} . The effectiveness of a preconditioner \mathbf{C} is determined by the condition number of $\mathbf{B} = \mathbf{C}^{-1}\mathbf{H}$, and occasionally by its clustering of eigenvalues. For the selected $\hat{\alpha}$, distribution of singular values $\sigma_m(\mathbf{H})$ was shown in Figure 9(a). For comparison, the singular values of the preconditioned operator $\sigma_m(\mathbf{B}^H\mathbf{B})$ is shown in Figure 11(a). Notice that, in order to get real values, the matrix product $\mathbf{B}^H\mathbf{B}$ is formed. The spectra $\sigma_m(\mathbf{H})$ has changed in $\sigma_m(\mathbf{B}^H\mathbf{B})$, and exhibits clustering to some extent. Although the results are far from optimal. To see the effect of this change, Figure 11(b) shows the number of iterations (steps in the PCG method). The number of iterations are plotted for each linear transform at frequencies ν . The red line represents PCG iterations, while the black line represents iterations by the standard CG method. In figures 12 and 13, the same exercise is shown for a preconditioner from the generalized Jackson kernel. This preconditioner is designed for ill-conditioned problems and yields a slightly better result.

To sum up, the following results were identified from this exercise:

- (1) None of the tested circulant preconditioners could stabilize the problem on their own.
- (2) Different frequencies in the signal may require a more dynamic way of selecting the preconditioner. Choice of sampling (and regularization) parameters, also have a significant effect on stability of the operator.
- (3) With the stabilized Toeplitz operator, larger α gives a lower condition number, hence a decrease in number of PCG (and CG) iterations. This is at the expense of bias being introduced in the solution.
- (4) At frequencies where the number of iterations are low (with the PCG method), the clustering properties of the singular values tend to be better. This coincide with the theory on preconditioners.
- (5) The condition numbers of the preconditioned system, based on the generalized Jackson kernel in Figure 13, tend to be lower for higher frequencies. This is a positive effect which is reflected in the number of PCG iterations.

It has been shown that is it possible to compute the parabolic Radon transform by preconditioning with circulant matrices. The average number of iterations by the PCG method was lower then with the standard CG method, although for this parameter choice, the PCG method does not perform well for the lower frequencies. The computational gain (by less iterations) might not be worth the effort; it could sacrifice some of the robustness of the standard methods like just applying the Levinson algorithm to the stabilized problem. The advantage of faster convergence also needs to offset the computational cost of calculating the preconditioner. The conjugate gradient methods uses a tolerance limit for when convergence is reached, adjusting this has a big influence on computational efficiency. A standard limit of $\epsilon = 10^{-7}$ have been used for tests in this section.

6.5.1. Comments. Numerical test with this problem, trying different parameters systematically, might yield a more exact criterion for when to use or not to use a preconditioner. However, the standard methods, without preconditioning, are already very fast and the effort of PCG is questionable. From previous chapters, it is clear that there is no need for oversampling and the system will typically not be as big as demonstrated in Figure 7. On the other hand, the benchmarks shows that the problems scale well, an important asset which can be related to other problems with Toeplitz structure.

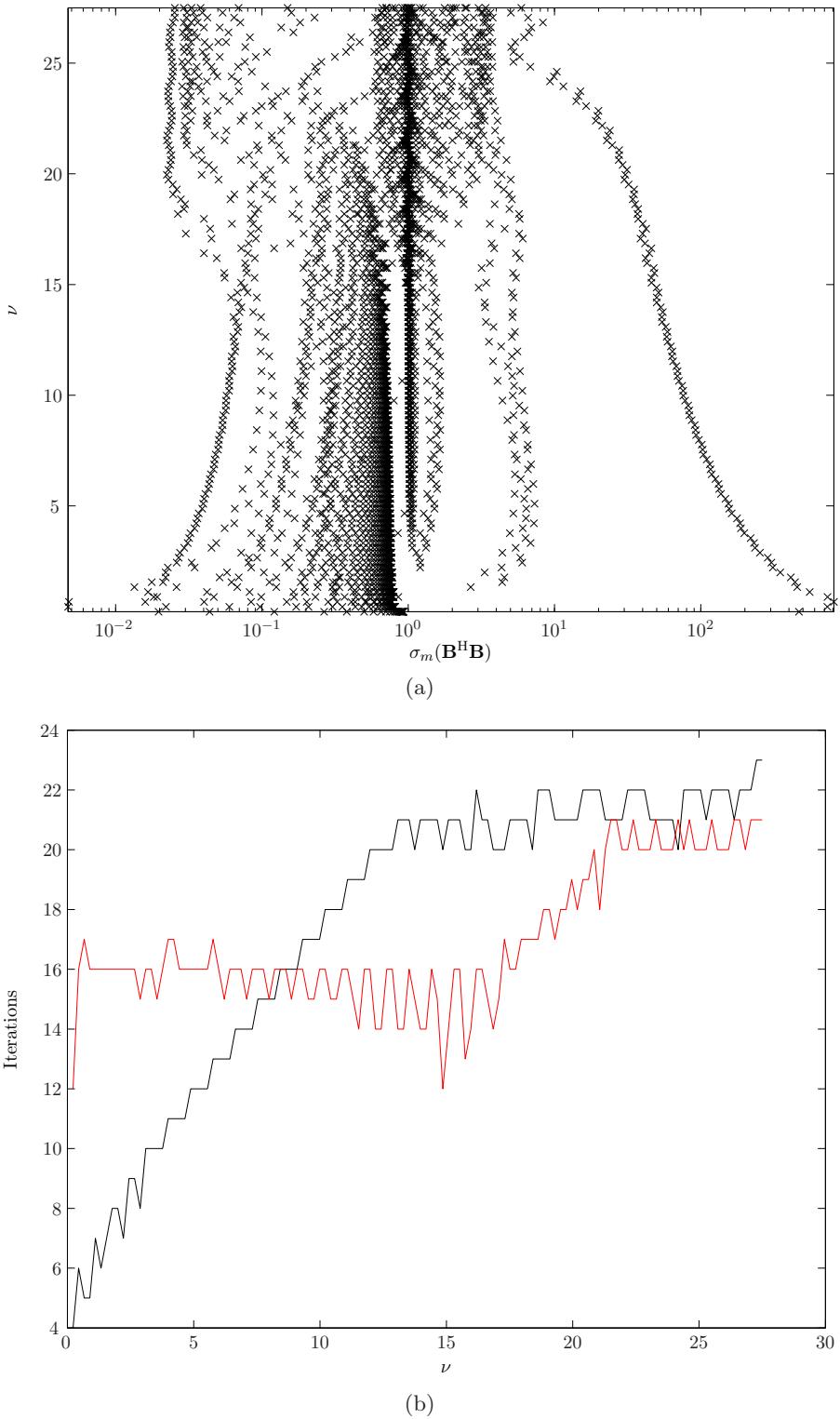


Figure 11: Performance of the circulant preconditioner, stabilized with $\hat{\alpha}$. a) Distribution of the singular values for different frequencies. b) Number of iterations before reaching convergence of the CG method versus the PCG method (red line).

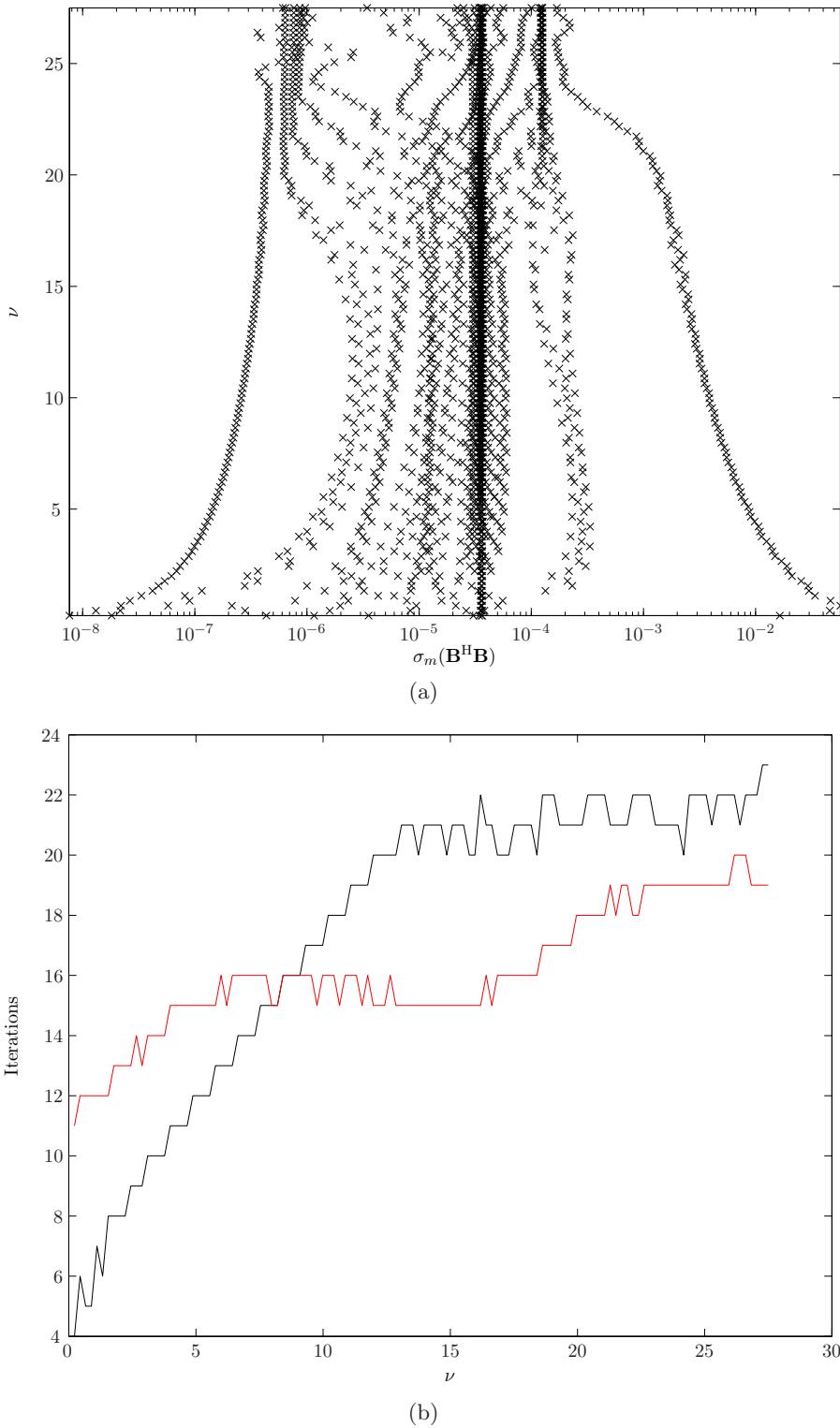


Figure 12: Performance of the circulant preconditioner, with \mathbf{C} from the generalized Jackson kernel, stabilized with $\hat{\alpha}$. a) Distribution of the singular values for different frequencies. b) Number of iterations before reaching convergence of the CG method (red line).

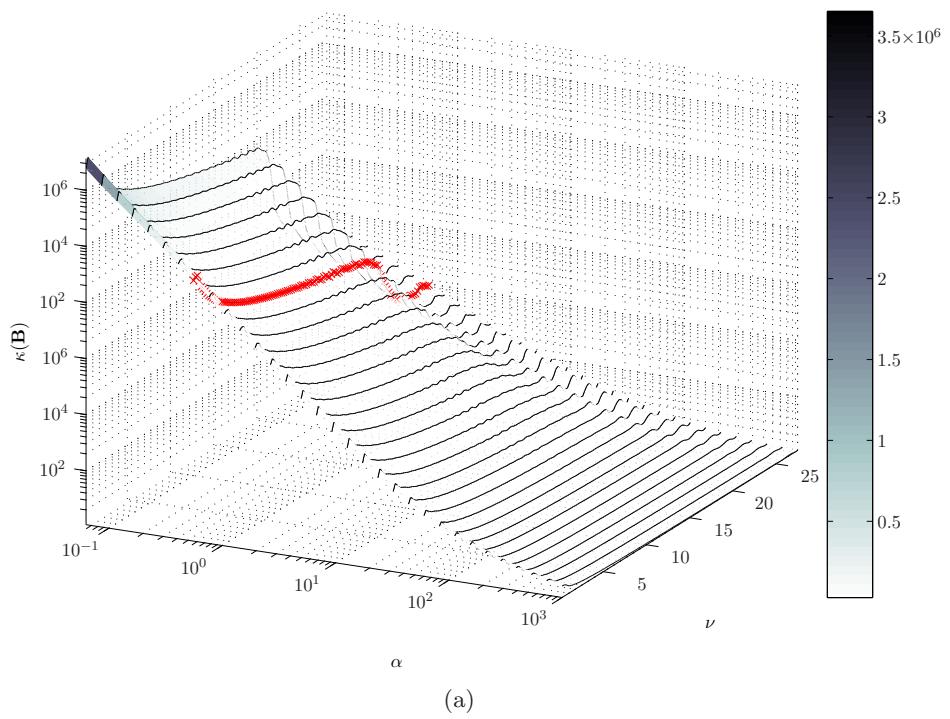


Figure 13: Condition number $\kappa(\mathbf{B})$, with \mathbf{C} from the generalized Jackson Kernel, for different choices of the regularization parameter α .

CHAPTER 5

Analysis of the Low-pass Filter

The preceding chapters have reviewed well established methods for solving and stabilizing the time-invariant Radon transforms. Very small singular values yield a large condition number and the problem is classified as ill-conditioned for a range of frequencies. For a different perspective, this chapter will return to the analysis in Chapter 1 (discretization of the time-invariant Radon transform). The theory on Toeplitz structure and circulant matrices from Chapter 4 is also a prerequisite for the following analysis.

The approximate forward transform $\tilde{\mathbf{z}} = \mathbf{A}^H \mathbf{b}$ is in fact a filtered version of the least squares forward transform $\hat{\mathbf{z}} = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \mathbf{b} = \mathbf{H}^{-1} \tilde{\mathbf{z}}$. Analysing the filter yields more direct insight into the problem. Following the notation from Chapter 1, let \tilde{r}_j be the approximate solution (low resolution Radon panel) and \hat{r}_m the least squares solution (higher resolution Radon panel). Then, the convolution is given as

$$(31) \quad \tilde{r}_j = \sum_{m=0}^{M-1} \hat{r}_m h_{j-m}, \quad j = 0, 1, \dots, M-1.$$

For $M = 4$, this is written in explicit matrix form as

$$(32) \quad \begin{pmatrix} \tilde{r}_0 \\ \tilde{r}_1 \\ \tilde{r}_2 \\ \tilde{r}_3 \end{pmatrix} = \begin{pmatrix} h_0 & h_{-1} & h_{-2} & h_{-3} \\ h_1 & h_0 & h_{-1} & h_{-2} \\ h_2 & h_1 & h_0 & h_{-1} \\ h_3 & h_2 & h_1 & h_0 \end{pmatrix} \begin{pmatrix} \hat{r}_0 \\ \hat{r}_1 \\ \hat{r}_2 \\ \hat{r}_3 \end{pmatrix} = \begin{pmatrix} h_0 & h_1^* & h_2^* & h_3^* \\ h_1 & h_0 & h_1^* & h_2^* \\ h_2 & h_1 & h_0 & h_1^* \\ h_3 & h_2 & h_1 & h_0 \end{pmatrix} \begin{pmatrix} \hat{r}_0 \\ \hat{r}_1 \\ \hat{r}_2 \\ \hat{r}_3 \end{pmatrix},$$

with coefficients h_m forming the diagonals of a Hermitian Toeplitz matrix. This matrix structure has an important property of one column being enough to characterize the entire matrix. Notice that the coefficients does not form a circulant structure, hence, the convolution cannot be evaluated by the discrete Fourier transform.

The low-pass filter $h_m = h_m(\omega \Delta p \Delta x)$ is a geometric series on the form

$$h_m = \sum_{n=0}^{N-1} \exp(i\omega m \Delta p n \Delta x) = 1 + E_m + E_m^2 + \dots + E_m^{N-1} = \frac{1 - E_m^N}{1 - E_m},$$

where $E = \exp(i\omega m \Delta p \Delta x)$. For the more complicated parabolic Radon transform, the filter would contain a squared term for the offset variable; no explicit expression for the series exist. The filter h_m (for the linear Radon transform) is then a natural choice for further analysis.

1. Zero-phase Low-pass Filter

A zero-phase filter has a phase spectrum (imaginary part) consisting of only zeros, and a real and symmetric impulse response. With uniform sampling, it is practical to assume symmetrical sampling about $p = 0$. Let $\mathbf{p} = [p_{-M}, \dots, p_0, \dots, p_M]$ and $x_n = n\Delta x$ for $|n| \leq N$. This gives the well known geometric series in the form of the Dirichlet kernel

$$h_m = \sum_{n=-N}^N \exp(i\omega n\Delta x p_m) = \frac{\sin[(2N+1)\omega\Delta x p_m/2]}{\sin(\omega\Delta x p_m/2)}.$$

The impulse response is real and symmetric, with a corresponding real and symmetric Fourier spectrum. These are properties which simplify interpretation and intuitive understanding of the filter. It is obvious that the sine waves will vary from rapid to more moderate oscillations, depending on the input parameters $\omega\Delta p\Delta x$. Choosing a critical sampling interval (according to the sampling theorem), for $h_m(\omega_{\max}\Delta x\Delta p)$, will generally lead to oversampling of the filter $h_m(\omega_{\min}\Delta x\Delta p)$. This is a consequence a uniform sampling scheme being used for different frequencies (fixed sampling intervals for offset and ray-parameter in the seismic data).

For convenience, the filter is denoted $h_m = h(m\Delta p) = h(p_m)$. With this definition, $h(p_m)$ has a periodicity

$$P_m = \frac{2\pi\ell}{\omega\Delta x}, \quad \ell = 0, \pm 1, \pm 2, \dots$$

which is evaluated as limit problem

$$\lim_{\ell \rightarrow 0, \pm 1, \pm 2, \dots} h(P_m) = \lim_{\ell \rightarrow 0, \pm 1, \pm 2, \dots} \frac{\sin[(2N+1)\pi\ell]}{\sin(\pi\ell)} = 2N + 1.$$

The roots are given by

$$\tilde{p}_m = \frac{2\pi\ell}{\omega\Delta x(2N+1)}, \quad \ell = \pm 1, \pm 2, \dots$$

which gives

$$h(\tilde{p}_m) = \frac{\sin(\pi\ell)}{\sin[\pi\ell/(2N+1)]} = 0,$$

unless $\ell = \pm(2N+1)$ or $\ell = 0$. These expressions are mainly of theoretical interest. It is observed that \tilde{p}_m closely resembles the sampling criterion which is discussed in the next section.

Figure 1(a) shows a Toeplitz matrix from a symmetric impulse response. The matrix is diagonalized by the Fourier basis, yielding a box-car low-pass filter (on the diagonal), shown in Figure 1(b). It is important to notice that diagonalization by the Fourier basis is not a general property for Toeplitz matrices.

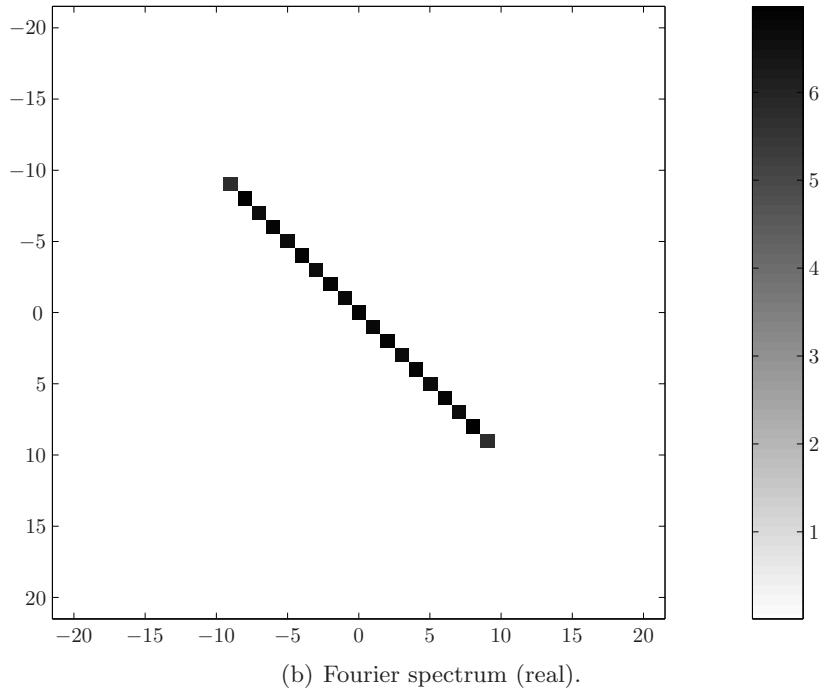
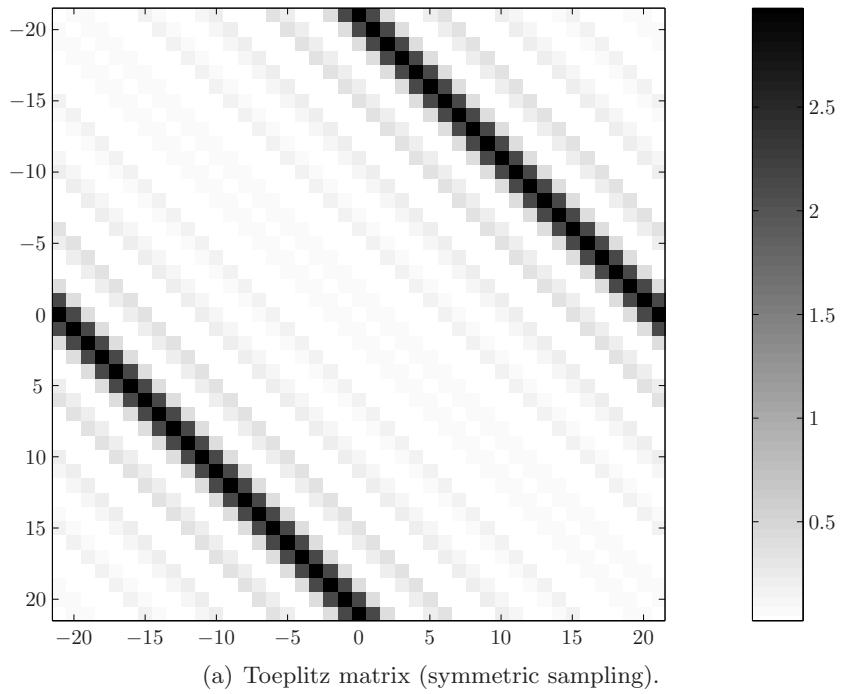


Figure 1: a) The Toeplitz matrix formed by the coefficients h_m (symmetric sampling). b) The corresponding spectrum by Fourier basis decomposition.

2. Sampling and Aliasing

The filter h_m defines a Hermitian Toeplitz matrix which changes characteristics for each frequency ω (or sampling interval Δp and Δx). To simplify the analysis, the parameter β is defined as

$$(33) \quad \omega \Delta p \Delta x = \frac{2\pi\beta}{2N+1}.$$

This gives the filter $h_m = h_m(\beta)$ on the form

$$h_m(\beta) = \frac{\sin(\pi\beta m)}{\sin[\pi\beta m/(2N+1)]}.$$

A low-pass filter, changing characteristics by the parameter β . Increasing β will push the signal towards higher frequencies which will lead to a requirement for more samples. Adjusting the parameter also corresponds to shrinking the sampling period of the discrete signal if $0 < \beta < 1$ or stretching if $\beta > 1$. The discrete Fourier transform H_k of h_m , with $\beta = 1$, gives

$$H_k = \sum_{m=1}^{2M+1} \frac{\sin(\pi m)}{\sin[\pi m/(2N+1)]} e^{2\pi i k m / (2M+1)} = 2N+1,$$

for $k = 1, 2, \dots, (2M+1)$. This is an ‘all-pass’ filter, corresponding to a Dirichlet kernel, sampled at the roots (unless $m = 2M+1$). For $0 < \beta < 1$ the filter $h_m(\beta)$ becomes a low-pass filter where well-known numerical effects like Gibbs phenomenon will influence the result. For $\beta > 1$, aliasing in form of folding will distort the filter, an effect which should be avoided. The behavior when changing β is a consequence of the similarity theorem.

From this discussion, a sampling criterion for Δp could be derived with respect to a fixed offset sampling interval and a maximum frequency ω_{\max} . Suppose $\beta = 1$ such that Equation (33) is given as

$$\omega_{\max} \Delta p \Delta x = \frac{2\pi}{2N+1}.$$

Rearranging this expression, and denoting the offset range $x_r = (2N+1)\Delta x$, a sampling criterion for the linear Radon transform is given as

$$\Delta p \leq \frac{2\pi}{x_r \omega_{\max}}.$$

With this sampling criterion, the filter $h_m(\omega \Delta p \Delta x)$ will not be influenced by aliasing effects for any frequencies in the signal. The same criterion was derived in Chapter 1, where it was based on the local summations in phase principle (Turner, 1990).

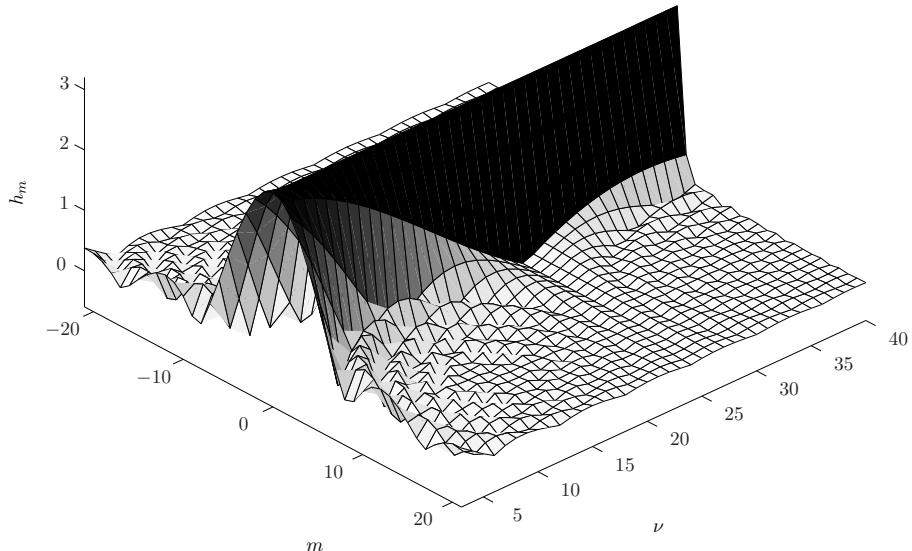
In Figure 2, the symmetric filter h_m and its spectrum H_k are plotted for different frequencies $\nu = \omega/2\pi$. The parameters are defined as $\Delta x = 25$ m and $\Delta p = 1/80$ s/km, with the number of samples given as $N = 63$ and $M = 21$. This gives a range in offset as

$$x_r = (2N+1)\Delta x = 3.175 \text{ km}.$$

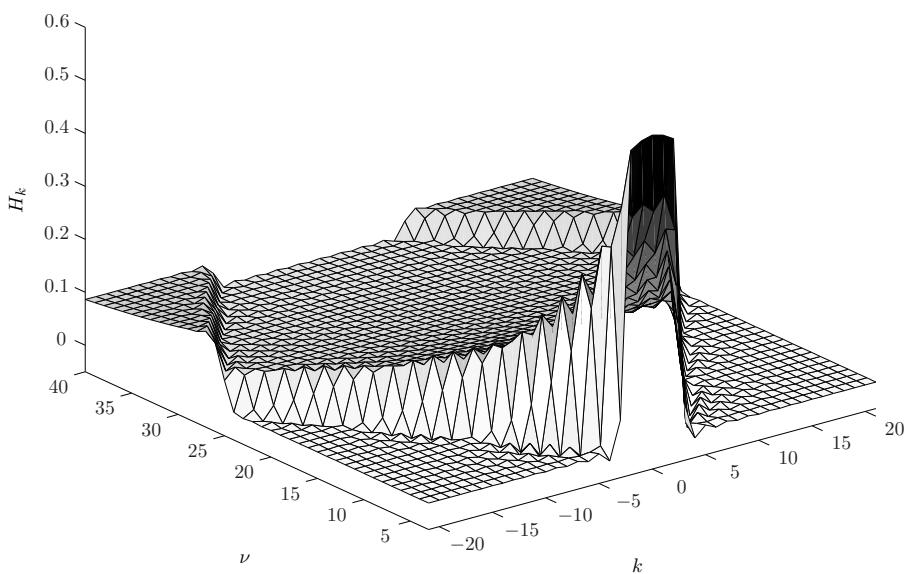
With this specific sample system, the critical frequency ν_c is

$$\nu_c = \frac{1}{x_r \Delta p} = \frac{1}{1/3.175 \text{ km} (1/80) \text{ s/km}} \approx 25 \text{ [1/s](Hz)}.$$

At this frequency, the filter becomes an ‘all-pass’ filter, and above this limit, aliasing in the form of folding is clearly in effect.



(a) Symmetric impulse response h_m .



(b) Zero-phase filter H_k .

Figure 2: Impulse response h_m (symmetric sampling) and frequency spectrum H_k (zero-phase).

3. Optimal Circulant Approximation

The convolution in Equation (32) does not have a circulant structure; it cannot be evaluated by the discrete Fourier transform. With this in mind, a circulant approximation of the convolution is studied. An (optimal) circulant approximation c_m of the filter h_m was derived in the previous chapter as

$$(34) \quad c_m = \frac{1}{M} [(M-m)h_m + mh_{M-m}^*].$$

These coefficients have the important property of Hermitian symmetry

$$c_m = c_{M-m}^*.$$

Replacing the right hand side of Equation (31), gives a new convolution on the form

$$(35) \quad \tilde{r}_j \approx \sum_{m=0}^{M-1} \hat{s}_m c_{j-m} = \sum_{m=0}^{M-1} \hat{s}_m \sum_{\ell=-\infty}^{\infty} c_{j-\ell M}.$$

If the new solution \hat{s}_m is a good approximation to \hat{r}_m or not, is uncertain at this point. However, with this formulation, the problem is given as a circular convolution. A circular convolution has the property $c_{M-m} = c_{-m}$, i.e. when $M = 4$ it is written on the following matrix form

$$\begin{pmatrix} \tilde{r}_0 \\ \tilde{r}_1 \\ \tilde{r}_2 \\ \tilde{r}_3 \end{pmatrix} \approx \begin{pmatrix} c_0 & c_{-1} & c_{-2} & c_{-3} \\ c_1 & c_0 & c_{-1} & c_{-2} \\ c_2 & c_1 & c_0 & c_{-1} \\ c_3 & c_2 & c_1 & c_0 \end{pmatrix} \begin{pmatrix} \hat{s}_0 \\ \hat{s}_1 \\ \hat{s}_2 \\ \hat{s}_3 \end{pmatrix} = \begin{pmatrix} c_0 & c_3 & c_2 & c_1 \\ c_1 & c_0 & c_3 & c_2 \\ c_2 & c_1 & c_0 & c_3 \\ c_3 & c_2 & c_1 & c_0 \end{pmatrix} \begin{pmatrix} \hat{s}_0 \\ \hat{s}_1 \\ \hat{s}_2 \\ \hat{s}_3 \end{pmatrix}.$$

Let the frequency spectrum of \tilde{r}_j , \hat{s}_m and c_m be denoted respectively \tilde{R}_k , \hat{S}_k and C_k , such that the circular convolution is written as a multiplication

$$\tilde{R}_k = \hat{S}_k C_k,$$

for $k = 0, 1, \dots, M - 1$. A formulation with well known computational advantages by the fast Fourier transform.

3.1. Comparison with the Toeplitz low-pass filter. The residual error $e_m = h_m - c_m$, for each coefficient, is given as

$$e_m = \frac{m}{M} (h_m - h_{M-m}^*) = \frac{m}{M} \sum_{n=0}^{N-1} [e^{i\omega \Delta x n \Delta p m} (1 - e^{-i\omega \Delta x n \Delta p M})].$$

The expression might not give any new insight, however it is visualized in Figure 3. The filter C_k is plotted for $\omega \Delta p \Delta x = 1/25$ with $N = 32$ and $M = 41$. The (corresponding) filter H_k is generated from the same set of parameters and plotted with dashed lines.

The Hermitian symmetry $c_m = c_{M-m}^*$ is seen from the symmetric real part $\Re\{c_m\} = \Re\{c_{-m}\}$ in Figure 3(a), and the antisymmetric imaginary part $\Im\{c_{-m}\} = -\Im\{c_m\}$ in Figure 3(b). Figures 3(c) and 3(d) show the real and imaginary frequency spectra, from which it is observed that the filter coefficients C_k are real-valued with zero-phase.

Figure 4 shows the impulse response of (non-symmetric sampling) h_m and its circulant approximation c_m , for the indexes $m = 0, 1, \dots, M - 1$. The parameters are defined as $\Delta x = 25$ m and $\Delta p = 1/45$ s/km, with the number of samples given as $N = 63$ and $M = 21$.

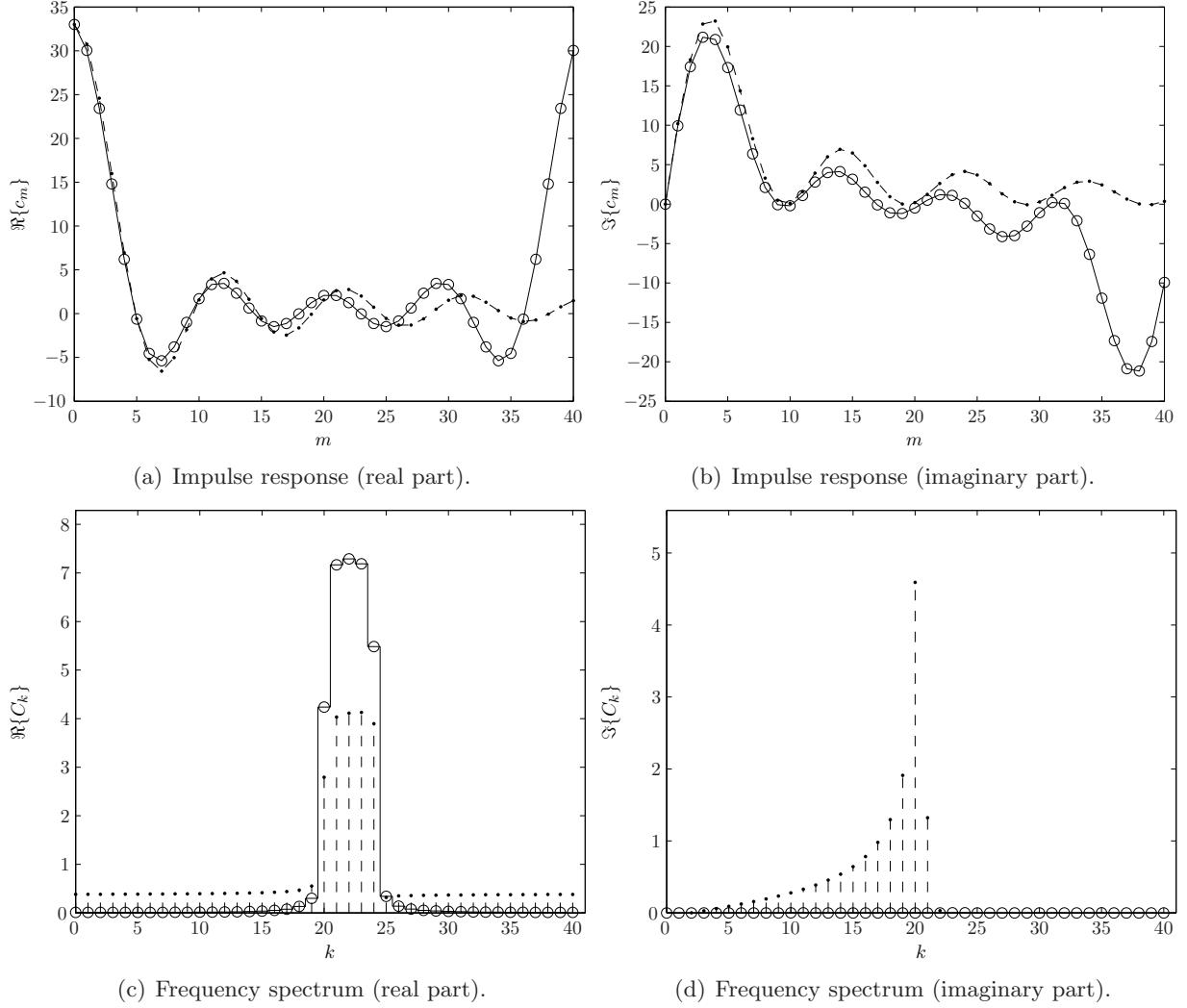
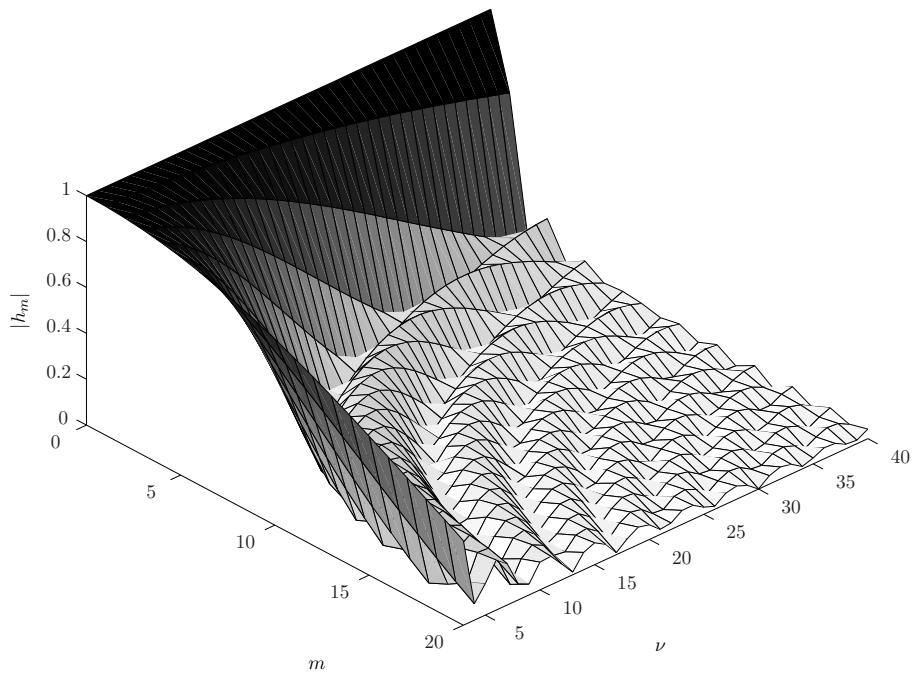
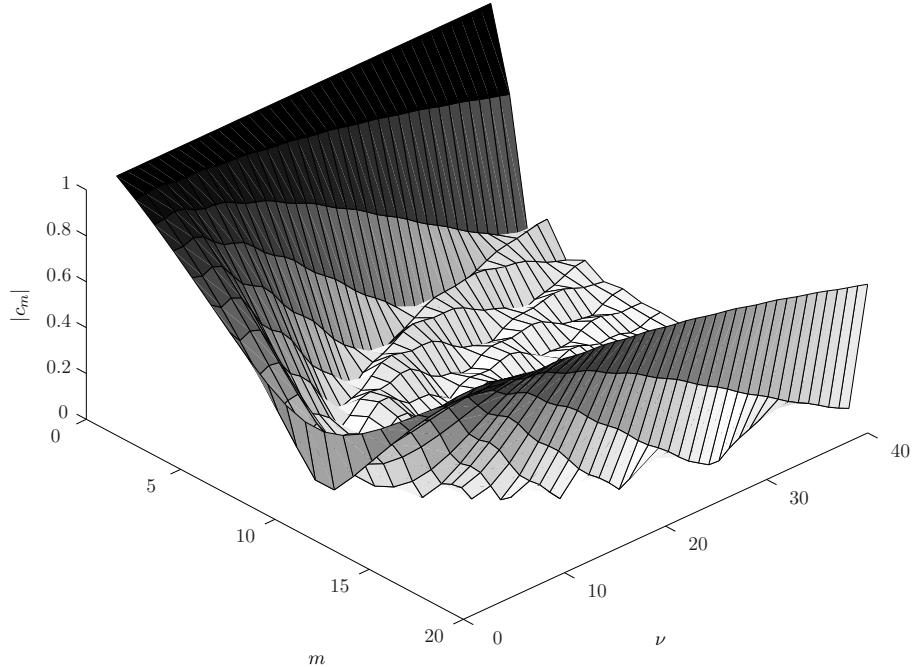


Figure 3: The filter C_k and the Hermitian impulse response c_m . The dashed line represents the corresponding, non-circular filter, H_k and impulse response h_m .



(a) Impulse response $|h_m|$.



(b) Circulant approximation $|c_m|$.

Figure 4: Impulse response of $|h_m|$ (complex, non-symmetric sampling) and $|c_m|$ (circulant approximation).

4. Circulant Approximation and Fixed Point Iteration

In general, the circulant approximation \mathbf{C} should not be used directly as a replacement for the square Toeplitz operator \mathbf{H} . This means that an iterative method, with the ability to converge towards the generalized inverse solution \hat{r}_m , is needed. The method of choice is fixed point iteration.

A number a , satisfying the equation $g(a) = a$, is called a fixed-point of the function g . Suppose that an equation of the type $f(z) = 0$, has a solution z . Adding z on both sides gives

$$g(z) = z + f(z) = z.$$

For certain kinds of functions, fixed points can be found by starting with an initial guess and calculating successive approximations. In general,

$$z_{k+1} = g(z_k), \quad k = 0, 1, 2, \dots$$

where

$$\lim_{k \rightarrow \infty} z_k = z.$$

Fixed points of functions in the complex plane commonly leads to fractal patterns, analogous to what was demonstrated with Newton's method in Chapter 3.

Let the convolution with the original Toeplitz filter, in Equation (31), be denoted $\mathbf{Hz} = \mathbf{w}$, with a solution \mathbf{z} . Adding the circulant approximation \mathbf{Cz} ,

$$\mathbf{Cz} + \mathbf{Hz} = \mathbf{w} + \mathbf{Cz},$$

and defining the Richardson iteration (Kelley, 1995), the fixed point recursion for the circulant approximation is given as

$$(36) \quad \mathbf{Cz}_{k+1} = \mathbf{w} - (\mathbf{H} - \mathbf{C})\mathbf{z}_k, \quad k = 0, 1, 2, \dots$$

The matrix \mathbf{C} is efficiently invertible, and the system can be written

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{C}^{-1}\mathbf{w} - \mathbf{C}^{-1}(\mathbf{H} - \mathbf{C})\mathbf{z}_k \\ &= \mathbf{C}^{-1}\mathbf{w} - \mathbf{Bz}_k, \end{aligned}$$

where $\mathbf{B} = \mathbf{C}^{-1}(\mathbf{H} - \mathbf{C})$ is called the iteration matrix. A solution \mathbf{z} is then found by

$$\lim_{k \rightarrow \infty} \mathbf{z}_k = \mathbf{z}.$$

The advantage of this formulation is that it can be calculated by the fast Fourier transform. To facilitate the FFT-implementation of this recursion, MATLAB code is included in Appendix B, Listing B.5.

4.1. Convergence. For convergence of the recursion, specific spectral properties has to be satisfied. The concept of spectral radius is used to make a complete description. Let λ_m denote the eigenvalues of $\mathbf{B} \in \mathbb{C}^{M \times M}$, this gives the spectral radius of \mathbf{B} as

$$\rho(\mathbf{B}) = \max_m |\lambda_m|.$$

Convergence of the recursion in Equation (36) is now guaranteed for all \mathbf{w} if and only if

$$\rho(\mathbf{B}) < 1.$$

Another important factor is the size of the spectral radius; the smaller $\rho(\mathbf{B})$, the faster the convergence. A detailed discussion, including proofs for the convergence criterions, is found in the much cited book by Kelley (1995).

5. Numerical results

A fundamental theory has been established with respect to convergence of the circulant approximation and fixed point iteration. This gives a starting point for numerical analysis of the problem, applied to the Radon transform. It is clear from the results in Chapter 4, that none of the available circulant preconditioners could stabilize the inversion matrix on their own. Let $\mathbf{B} \in \mathbb{C}^{M \times M}$ denote the iteration matrix with the stabilized Toeplitz operator \mathbf{H} . Further, let \mathbf{C} be the circulant approximation of \mathbf{H} , such that the iteration matrix for the fixed point recursion is given as

$$\mathbf{B} = \mathbf{C}^{-1}(\mathbf{H} - \mathbf{C}) = \mathbf{C}^{-1}(\mathbf{A}^H \mathbf{A} + \alpha \mathbf{I} - \mathbf{C}),$$

where α is the regularization parameter. A big condition number of the matrix \mathbf{B} imply that the spectral radius $\rho(\mathbf{B}) >> 1$, i.e. the circulant approximation method does not converge. This means that regularization is needed in order to test the fixed point iteration method.

Using a constant parameter α , for all frequencies, Figure 5 gives an indication of how to choose α . The figure shows the ℓ_2 -norm of reconstructed seismic, subtracted from the original seismic, versus different α (logarithmic scale). Higher values of the norm means that the solution is influenced strongly by α , i.e. the solution is less accurate. For this exercise, the system is defined by $\Delta t = 1/55$, with a period $T = 2.25$. Frequencies range from $1 \leq \nu \leq 30$ Hz with $N = 35$ samples in offset and $M = 30$ samples for the residual-moveout. Visual inspection shows that $0.1 < \alpha < 1$ are good choices, and that in this example, the best choice is α_b (blue circle). The red and green circle are values of α , respectively denoted α_r and α_g . These values are used in subsequent exercises.

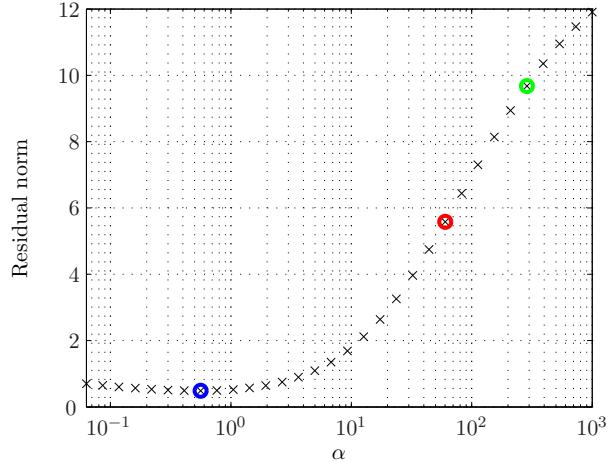
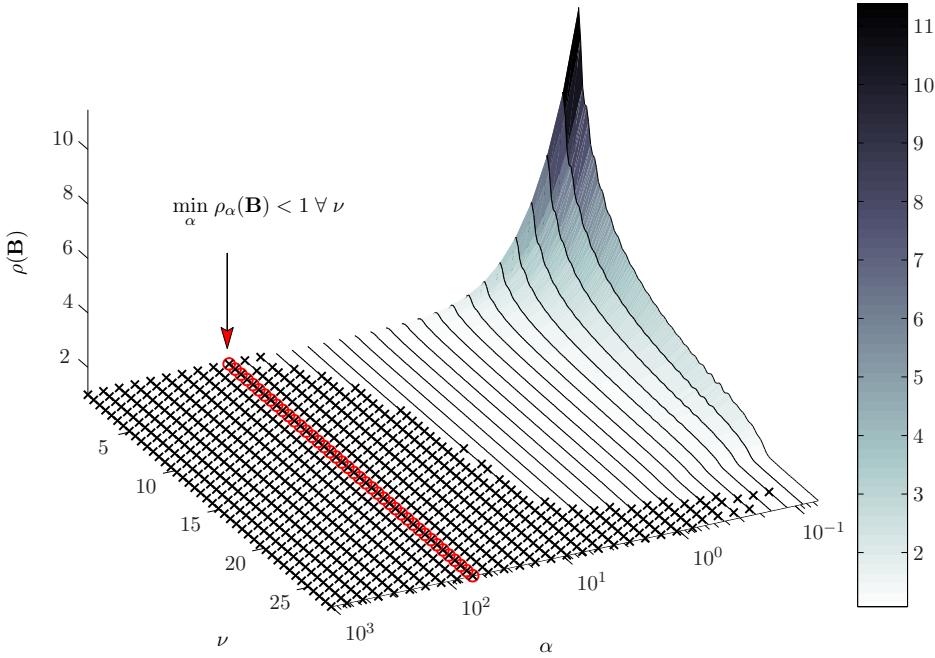


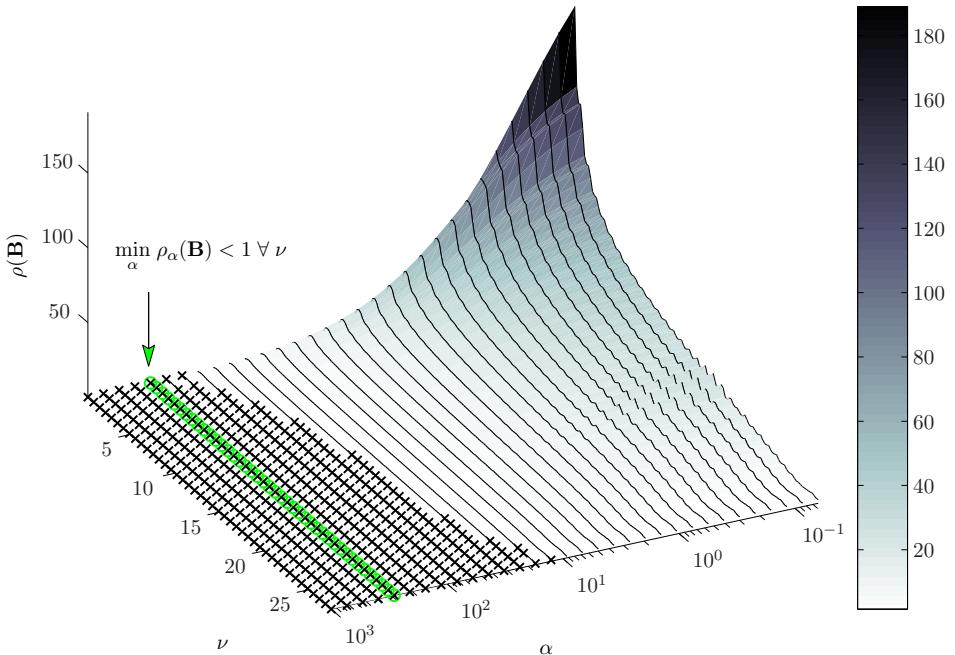
Figure 5: The ℓ_2 -norm of reconstructed seismic, subtracted from the original seismic, versus different α (logarithmic scale). Colored circles indicate α_b (blue), α_r (red) and α_g (green).

To demonstrate the circulant approximation method, a regularization parameter α has to be chosen such that the spectral radius $\rho(\mathbf{B}) < 1$ for all frequencies. The regularization parameter should also minimize the residual norm of the reconstruction, subtracted from the original seismic. Assume this means minimizing α . These criterions are stated in compact form as

$$(37) \quad \min_{\alpha} \rho_{\alpha}(\mathbf{B}) < 1 \quad \forall \nu.$$



(a) Circulant approximation from the generalized Jackson kernel.



(b) Optimal (T. Chan) circulant approximation.

Figure 6: The spectral radius is evaluated for different frequencies ν and regularization parameters α . Crosses mark points where $\rho(\mathbf{B}) < 1$.

In Figure 6, the spectral radius $\rho(\mathbf{B})$ is evaluated for two different circulant approximations. Circulant approximation by the generalized Jackson kernel in 6(a) and the optimal circulant approximation in Figure 6(b). The red (α_r) and green (α_g) circles indicate where Equation (37) is satisfied. These choices of α lead to convergence of the fixed point iteration for all frequencies. However, optimality of the solution (and reconstruction) is sacrificed as a consequence of $\alpha_r \gg \alpha_b$ and $\alpha_g \gg \alpha_b$. That is, both choices of $\alpha = \{\alpha_g, \alpha_r\}$, will lead to a loss of energy in the Radon panel and reconstruction. Notice that the sample system is the same as the one used in Figure 5, where the red, green and blue α -values are indicated.

5.1. Convergence and number of iterations. Another aspect is convergence and number of iterations in the fixed point recursion. Even though convergence is guaranteed if $\rho(\mathbf{B}) < 1$, the size of $\rho(\mathbf{B})$ determines the rate of convergence. This is important since it can have big effects on computational speed. For the fixed point algorithm to qualify as a fast method, a more thorough investigation of the spectral radius is needed.

In Figure 7, the spectral radius $\rho(\mathbf{B})$ is shown for both approximations. In parallel, the number of fixed point iterations are shown. From this, it is clear that although the Jackson approximation does a better job at lowering α , the optimal approximation gives lower values for the spectral radius. Hence, the optimal approximation requires much less iterations to reach convergence when the tolerance limit is $\epsilon = 10^{-7}$.

Figure 8 shows the seismic data and parabolic Radon panels. As a reference, Figure 8(b) shows the Radon panel for α_b , calculated by the standard method (e.g. Levinson algorithm). Further, Figure 8(c) shows the Radon panel calculated by the circulant approximation and fixed point iteration¹ with $\alpha = \alpha_r$. Reconstruction is included in Figure 8(d). All images are plotted at the same scale, this illustrates some loss of amplitude in Figure 8(c). This is a consequence of regularization, i.e. the parameter α_r has too much influence on the solution.

5.2. Comments. Under specific conditions, it is possible to calculate the Radon transform by the fast Fourier transform and fixed point iteration. Imposed regularization is the most important factor for success. The method loses some of its elegance (and advantage) when the problem needs to be stabilized in advance. It is still an interesting approach, applicable to other signal processing problems.

It is important to emphasize that although it is possible to calculate the Radon panel in this manner, it is demonstrated that this method should not be used as a replacement for other algorithms like the Levinson or CG-FFT. These methods have more relaxed criterions for success, an important property for obtaining reliable solutions of problems with varying parameter choices.

¹Implemented with matrix-vector operations performed by the fast Fourier transform.

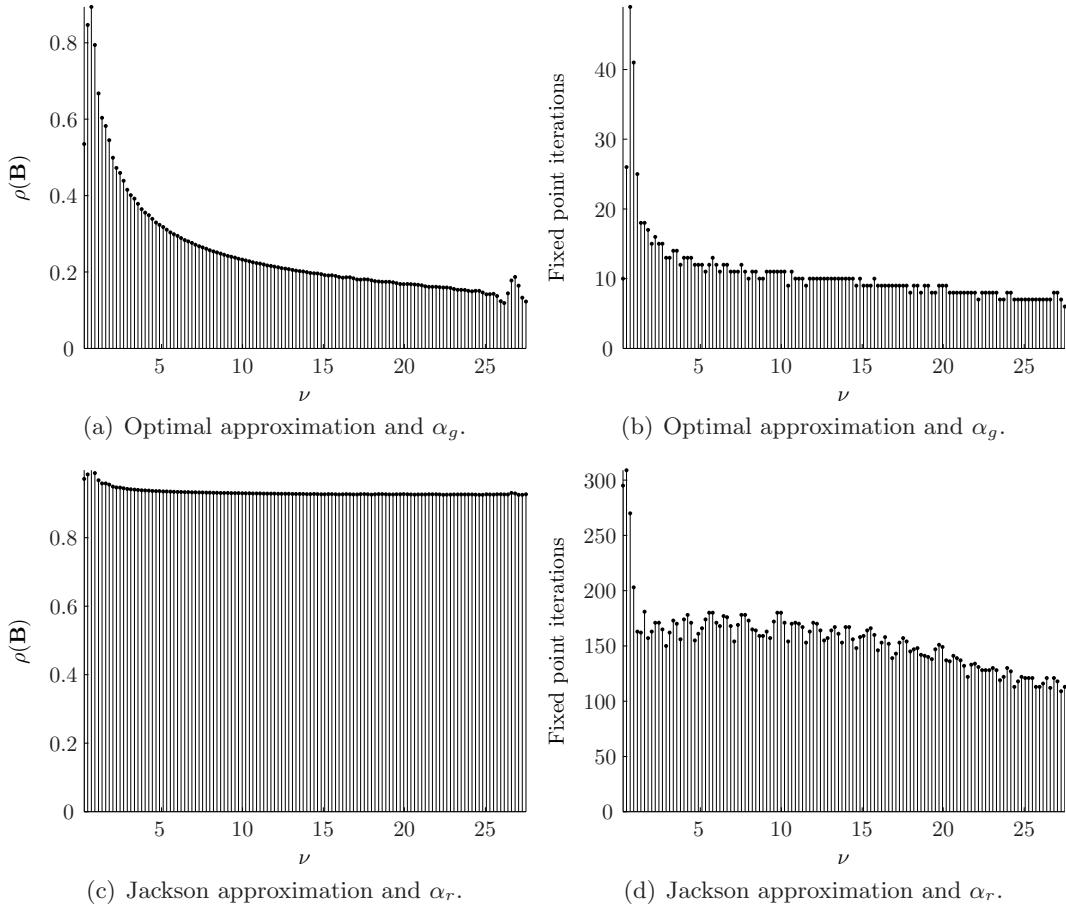


Figure 7: Convergence rate compared to spectral radius, plotted for both approximations.

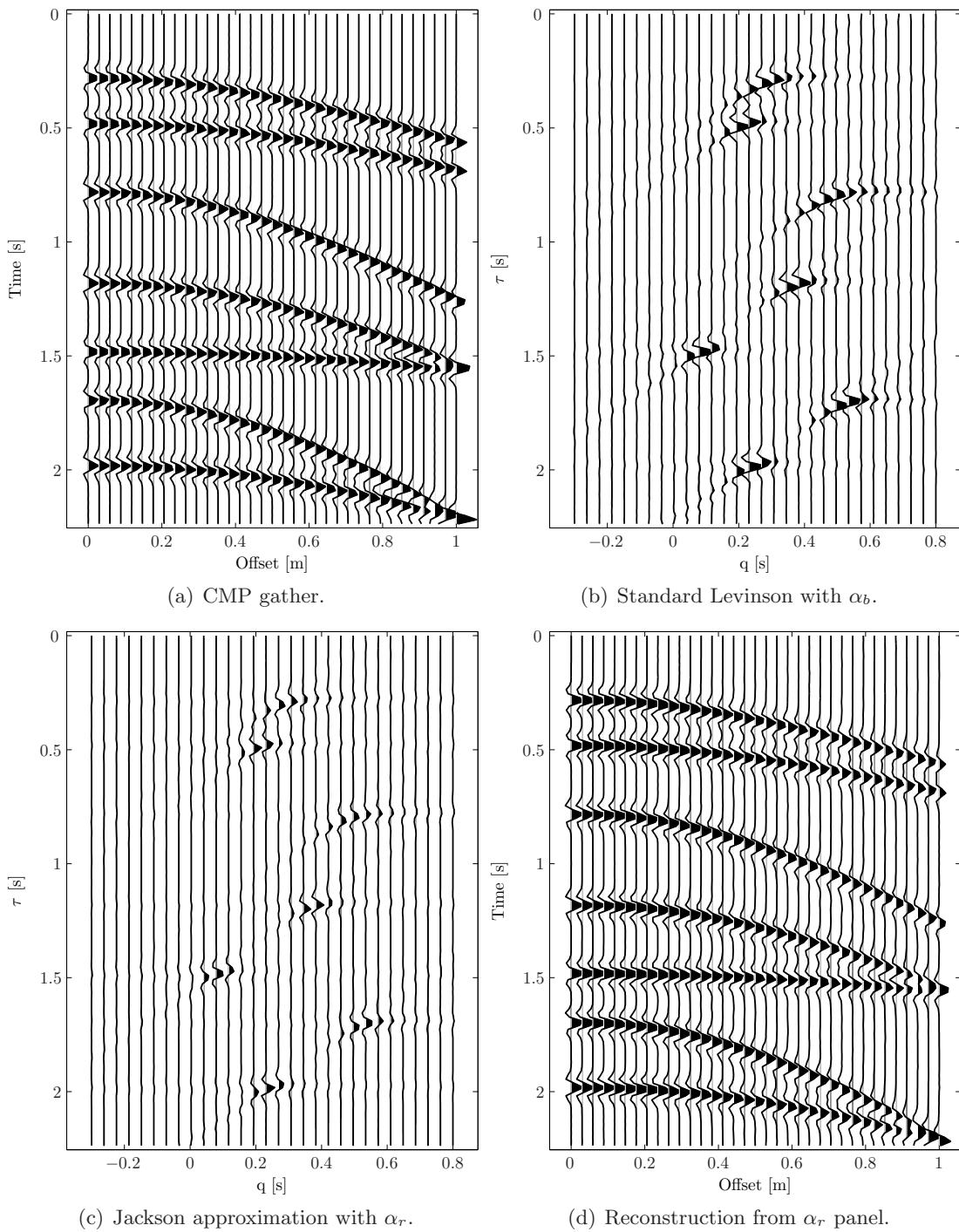


Figure 8: The parabolic Radon transform for different choices of α ($\alpha_r \gg \alpha_b$) and corresponding methods.

CHAPTER 6

Sparse Approximations - High-Resolution Methods

Different regularization techniques such as Tikhonov and TSVD was discussed in Chapter 3. These familiar methods produce stable results, but without the ability to promote sparseness in the solution. Regularization of linear systems with sparsity constraints seeks out to find a sparse approximation (Chen et al., 2001) \mathbf{x} such that $\mathbf{Ax} \approx \mathbf{b}$. Sparsity constraints often require that the solution has few nonzero entries relative to its dimensions. With the Radon transform, it is desirable to have a method able to discriminate fast, short-period multiples under limited offset situations (typically the situation for real data, where events in the Radon panel are clustered together). This is the motivation for the development of different ‘ad-hoc’ methods, serving the purpose of obtaining a clean and simple Radon panel without artifacts from limited aperture and aliasing, referred to as high-resolution Radon transforms. Development of these methods play an important role in industrial directed research. Several different techniques (Tropp and Wright, 2010) exists for computing a sparse approximation problem: greedy pursuit, convex relaxation, Bayesian framework, nonconvex optimization and brute force.

1. The Bayesian Approach

Stochastic inversion for velocity stacking was originally proposed by Thorson and Claerbout (1985). Enforcing sparseness in the hyperbolic Radon panel would fill gaps in CMP gathers after reconstruction. Their method however was expensive to calculate, and the time-invariant parabolic transform (Hampson, 1986) would become the preferred implementation. With this frequency domain formulation Sacchi and Ulrych (1995) introduced a high-resolution Radon transform, also derived in a stochastic framework. The basic concepts from this paper is reviewed in the following sections.

It should be noted that the time-variant hyperbolic Radon transform has been revisited by Trad et al. (2002, 2003), proposing methods which keeps calculation costs of a high-resolution hyperbolic Radon panel low enough for practical implementations.

The relation between model space and data space is given by $\mathbf{Ax} + \mathbf{e} = \mathbf{b}$, where \mathbf{e} is a noise term. Bayes formula is written

$$(38) \quad p(\mathbf{x}|\mathbf{b}) = \frac{p(\mathbf{b}|\mathbf{x}) p(\mathbf{x})}{\int p(\mathbf{b}|\mathbf{x}) p(\mathbf{x}) d\mathbf{x}} \propto p(\mathbf{b}|\mathbf{x}) p(\mathbf{x}),$$

where $p(\mathbf{b}|\mathbf{x})$ is the probability density function of \mathbf{b} given \mathbf{x} , $p(\mathbf{x})$ is the prior probability density function of the model and $p(\mathbf{x}|\mathbf{b})$ is the posterior probability density function of the model. The model \mathbf{x} is a random variable and the solution is a probability density function for the model parameters. Bayes formula relates a priori information and measured data (in many physical applications), constructing the most probable \mathbf{x} . A specific prior could then produce a Radon panel with desired properties such as sparseness.

2. The Principle of Maximum Entropy

Finding the correct posterior probability density function $p(\mathbf{x})$ is a difficult problem in inverse theory. It gives raise to problems like incorporating subjective decisions into the model and possibly producing misleading results. Deriving a posterior distribution with the mathematical method of maximum entropy is in some sense a way to avoid subjective influence on the results. The maximum entropy principle makes inference about the probability law of a random variable, and finds a probability density function which best represents the observed data.

The entropy h is defined as

$$h = - \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} p(\mathbf{x}) \ln p(\mathbf{x}) d\mathbf{x},$$

maximized to the constraints

$$\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} p(\mathbf{x}) d\mathbf{x} = 1 \quad \text{and} \quad \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} p(\mathbf{x}) g_r(\mathbf{x}) d\mathbf{x} = E[g_r(\mathbf{x})] = \mu_r,$$

where $g_r(\mathbf{x})$ defines the constraints and μ_r is the expected value for $r = 1, 2, \dots$. Using only one constraint, the resulting distribution (Karmeshu, 2003) is

$$(39) \quad p(\mathbf{x}) \propto \exp[-\lambda_0 - \lambda_1 g(\mathbf{x})],$$

where λ are Langrangian multipliers determined from the constraints. Pursuing a sparse Radon panel with isolated seismic events, a suitable constraint has to be defined. A constrain which quantifies the amount of sparseness of a vector is chosen as

$$(40) \quad g(\mathbf{x}) = \sum_i \ln(|x_i|^2 + c),$$

where c is a small perturbation factor. The prior probability density function is then given as

$$p(\mathbf{x}) \propto \frac{e^{-\lambda_0}}{\prod_i (|x_i|^2 + c)^{\lambda_1}}.$$

For the present problem, finding the Lagrange multipliers is not of interest. They are chosen by the user, controlling the amount of sparseness in the solution. Assuming the noise vector \mathbf{e} is Gaussian distributed, the likelihood function $p(\mathbf{b}|\mathbf{x})$ is

$$p(\mathbf{b}|\mathbf{x}) = K \exp(-(1/2)(\mathbf{A}\mathbf{x} - \mathbf{b})^H \mathbf{C}^{-1} (\mathbf{A}\mathbf{x} - \mathbf{b})),$$

where K is a normalization constant and \mathbf{C} is the covariance matrix (Johnson and Wichern, 1998) of the noise. The posterior probability density function is then given by Bayes formula

$$p(\mathbf{x}|\mathbf{b}) = p(\mathbf{b}|\mathbf{x}) p(\mathbf{x}) = K \exp(-\lambda_0 - \lambda_1 g_1(\mathbf{x}) - (1/2)(\mathbf{A}\mathbf{x} - \mathbf{b})^H \mathbf{C}^{-1} (\mathbf{A}\mathbf{x} - \mathbf{b})).$$

This formulation also needs instructions for how to select a single model. This is done by finding the maximum a posteriori solution.

3. Maximum a Posteriori Solution

If a single model is the preferred solution, choosing the one which maximizes $p(\mathbf{x}|\mathbf{b})$ is an appropriate choice. This is referred to as the maximum a posteriori (MAP) solution. The MAP solution is found by

$$-\frac{\partial}{\partial \mathbf{x}} \ln p(\mathbf{x}|\mathbf{b}) = \mathbf{0},$$

which leads to a solution on the form

$$\mathbf{x} = (\mathbf{A}^H \mathbf{C}^{-1} \mathbf{A} + \lambda \mathbf{P}(\mathbf{x})^{-1})^{-1} \mathbf{A}^H \mathbf{C}^{-1} \mathbf{b},$$

where $\lambda = 2\lambda_1$ and

$$\mathbf{P}(\mathbf{x}) = \text{diag}(|x_i|^2 + c).$$

This system is solved by the iterative reweighted least squares method. Experiments show that four or five iterations are sufficient for obtaining a sparse solution (Sacchi and Ulrych, 1995). While the model covariance matrix plays an important role when designing high-resolution Radon operators (Trad et al., 2002), only having elements on the diagonal imply no spatial correlation of noise.

A typical multiple removal scheme is used as an exercise with the Bayesian high-resolution method. Using the same sample system as in Chapter 1, section 4, Figure 2(b) shows a NMO corrected and muted CMP gather with its corresponding parabolic Radon transforms in Figure 1. The high-resolution Radon panel in Figure 1(b) is calculated using five iterations with the covariance matrix chosen as the identity matrix and $\lambda = \beta \|\mathbf{x}\|_2^2$ (initiated by the approximate solution and recalculated for each iteration). The parameter λ is now adjusted automatically based on energy levels¹ for each frequency. With $\beta = 2.5$ this yields a result which balances loss of amplitudes and the degree of sparseness. The Radon panel in Figure 1(a) is calculated using the standard damped least squares and for visual reference, the frequency-q domains are included for both variants in Figure 1(c) and 1(d). Filtering of primary events are done using a cut-off set to $[q_{\min}, q_{\text{cut}}]$ where $q_{\text{cut}} = 0.1$ s. Reconstruction of the data in Figure 2(c) then leaves only the multiples, which is subtracted from the original NMO corrected data giving the final image in Figure 2(d).

The Sacchi method works well and is easy to implement using the iterative reweighted least squares method. It is considered the industry standard, occurring with different constraints, i.e. priors. A drawback is the fact that the regularization term breaks the Toeplitz structure leading to increased computational time.

¹This is a choice made for this exercise. Theoretical methods for picking a suitable parameter λ (often referred to as the hyperparameter) is unknown to the author.

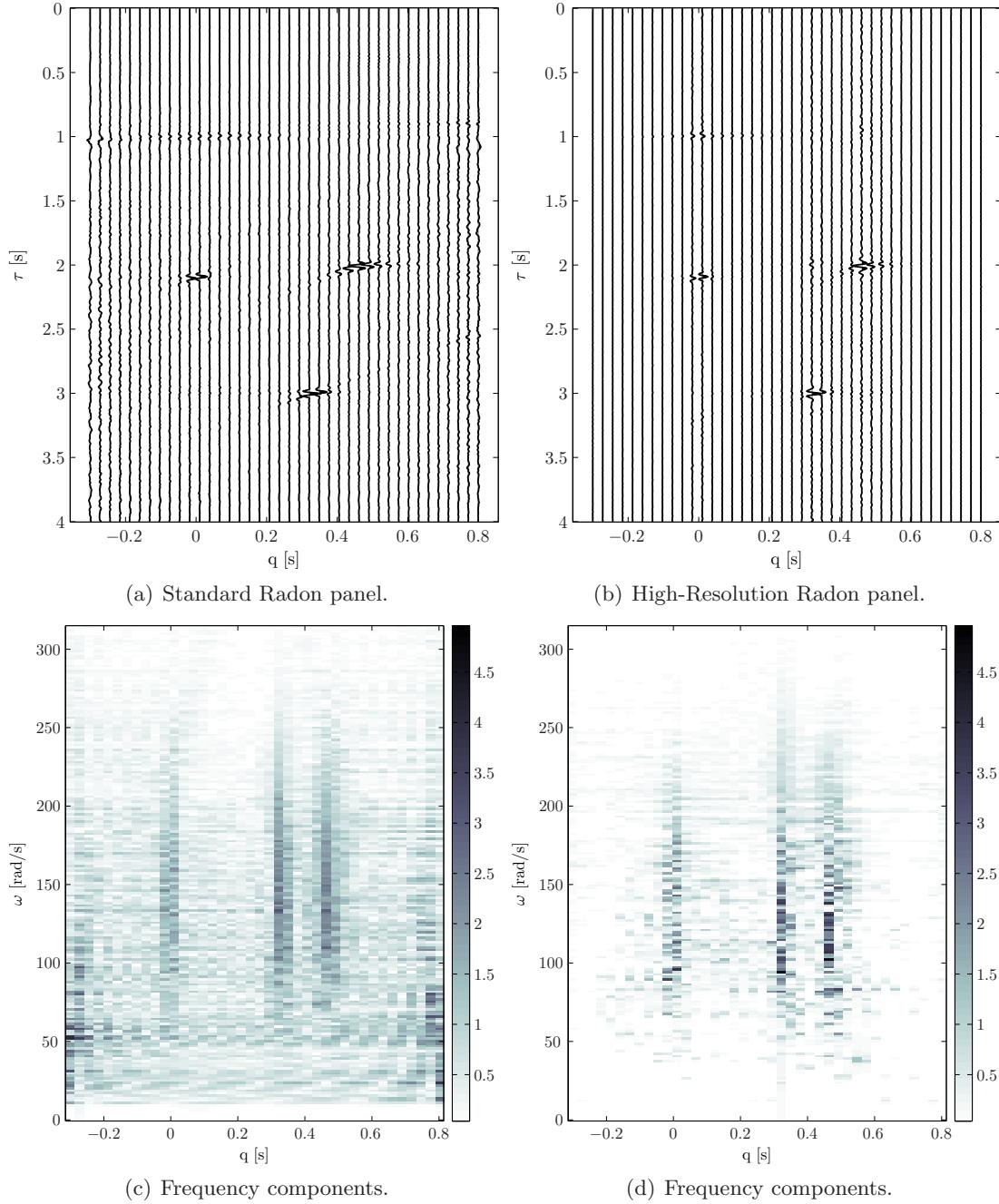


Figure 1: The parabolic Radon transform of a NMO corrected CMP gather given in Figure 2(b). a) Standard damped least squares solution with corresponding frequency components in c). b) Radon panel calculated by Bayesian a priori regularization, exhibits better focusing. Corresponding frequency components shown in d).

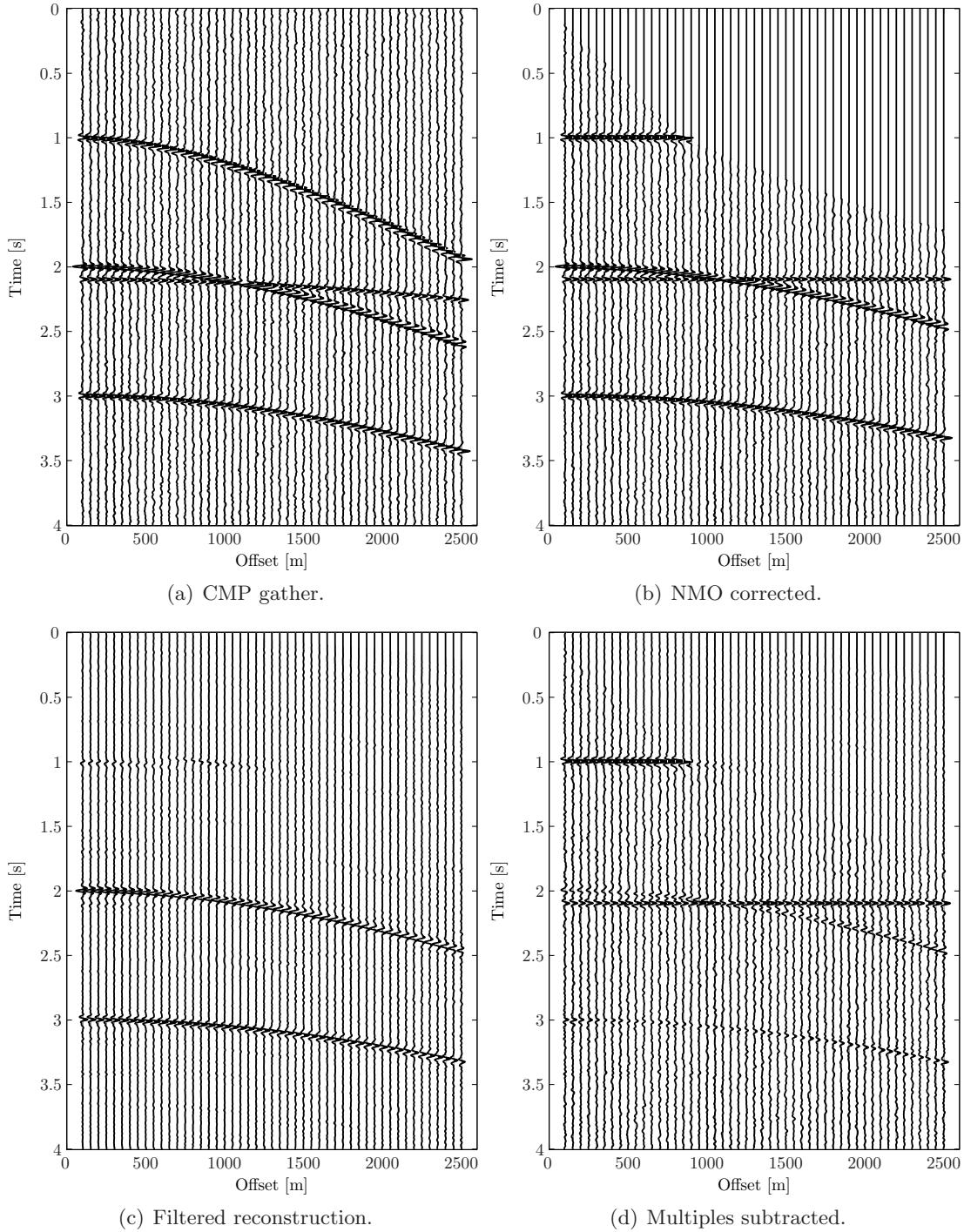


Figure 2: A high-resolution Radon demultiple work flow. a) Original CMP gather from the (demo dataset in the SeismicLab package). b) NMO corrected data (including mute), subject of the Radon transform c) Reconstruction of multiples with primaries removed. d) Multiples subtracted from the original CMP gather, leaving only the primaries.

4. One-norm Regularization and Basis Pursuit Solutions

In this section, ℓ_1 -norm solutions are studied with application to the Radon transform. A vast amount of theory concerning sparse approximations from different ℓ_1 -norm formulations have been developed (Tropp and Wright, 2010), many of whom are awaiting testing on real data applications. The aim is to find a sparse solution of the system of equations $\mathbf{Ax} = \mathbf{b}$. Suppose this is an underdetermined problem, retrieving a maximally sparse solution among the infinite number of solutions to the system is done by minimizing

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ subject to } \mathbf{Ax} = \mathbf{b},$$

where the ℓ_0 -norm is the number of non-zero elements of \mathbf{x} . Unfortunately the ℓ_0 -norm is a non-convex function, making this a difficult problem. Replacing the ℓ_0 -norm with a ℓ_1 -norm yield a convex optimization problem that admit tractable algorithms (Tropp and Wright, 2010). This relaxed formulation is often known as a basis pursuit problem (Chen et al., 2001), given as

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ subject to } \mathbf{Ax} = \mathbf{b}.$$

In the presence of noisy data, it is undesirable to fit this data exactly. Relaxing the constraints gives the basis pursuit denoise problem on the form

$$(41) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_1 \text{ subject to } \|\mathbf{Ax} - \mathbf{b}\|_2 \leq \sigma,$$

where σ is a measure of the noise level. A second formulation is the ℓ_1 -penalized least squares

$$(42) \quad \min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2 + \lambda \|\mathbf{x}\|_1,$$

where λ is the regularization parameter. This formulation is often preferred for its connection to convex quadratic programming (Boyd and Vandenberghe, 2004). It was discussed in Chapter 3, where the iterative reweighted least squares technique was suggested as an algorithm to solve the set of non-linear equations. A third formulation is often known as least absolute shrinkage and selection operator (Tibshirani, 1996) or LASSO. It is given as

$$(43) \quad \min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2 \text{ subject to } \|\mathbf{x}\|_1 \leq \gamma,$$

where γ is a parameter adjusting the trade off between the objectives. For appropriate parameter choices of σ , λ and γ , the solutions of equations (41), (42) and (43) coincide and these problems are in some sense equivalent (Berg and Friedlander, 2008).

In the scientific computing community, a great deal of research has gone into developing algorithms for solving these optimization problems. Picking one for testing with the present work is based on the following criterions:

- (1) Suitable for large-scale problems in the complex domain.
- (2) Easily available routines for implementation and testing.

Berg and Friedlander (2008) presents an algorithm for solving the basis pursuit denoise problem, satisfying the desired criterions. The authors provide a MATLAB implementation named SPGL1 (Spectral Projected Gradient for ℓ_1 -minimization), available under the GNU General Public License. Focusing on the basis pursuit denoise problem, with an approximately known σ , their algorithm is attractive for several different signal processing problems.

5. Basis Pursuit Denoise and the Radon Transform

The basis pursuit denoise and SPGL1 toolkit have already found applications to seismic data (Hennenfent and Herrmann, 2005, 2008). Testing of a ℓ_1 -penalized Radon transform is unknown to the author. Computations are done in the complex ω - q domain, typically on a large-scale dataset with an explicitly known linear operator \mathbf{A}_ω . For selection of the parameter σ , some dynamic scheme is needed as the basis pursuit denoise procedure is done for each frequency in the signal. Initiating \mathbf{x}_ω , using either the damped least squares or approximate solution, gives a measure of the energy in the signal. Hence, σ_ω can be written

$$\sigma_\omega = \beta \|\mathbf{x}_\omega\|_1, \quad \beta > 0.$$

This gives a dynamic parameter, where β quantifies the amount of sparseness. Numerical experiments (not included in this thesis) shows that keeping σ fixed for all frequencies, tend to create distortion of the image.

5.1. Results. A set of exercises will demonstrate the merits of the basis pursuit denoise method applied to the parabolic Radon transform. Emphasis has been made to include a complete multiple removal scheme, in order to show how loss of amplitudes may introduce a problem with applications in seismic processing.

Using the same sample system as the previous exercise (Figure 2), the parabolic Radon panel in Figure 3 is calculated by the basis pursuit denoise method for $\beta = \{0.6, 0.8, 1, 1.2\}$. In Figure 4, the frequency components are shown for the same panels, revealing more of how the regularization directly affects the (frequency- q) Radon panel. The dashed red line at $\omega = 158$ rad/sec represents a selected frequency which is inspected in Figure 5. In these figures, the stairs plot represent the basis pursuit denoise solution while the red dashed line show the standard least squares solution. This exercise show how an increase in β gives an increase in sparseness, however at the cost of loss in amplitude. For comparison, the standard damped least squares solution is the same as in Figures 1(a) and 1(c).

In Figure 6, the NMO corrected CMP gather from Figure 2(a) is transformed by the basis pursuit denoise method using $\beta = 0.5$. Filtering, reconstruction and multiple removal reveals a Radon panel with better resolution, but at the expense of some loss of amplitude. Using the same algorithms and work flow with $\beta = 0.35$, a NMO corrected CMP gather from the Gulf of Mexico (courtesy of the signal analysis and imaging group, University of Alberta) is shown in Figure 7. A comparison between the standard Radon panel and the basis pursuit solution for this real dataset, including their frequency components are shown in Figure 8. A typical deep water CDP gather from the Knipovich Ridge² is shown in Figure 9. Its basis pursuit denoise Radon panel is calculated with $\beta = 0.35$.

²A joint survey by the Institute of Seismology and Vulcanology, Hokkaido University and Institute of Solid Earth Physics, University of Bergen. The survey was conducted by the research vessel ‘Haakon Mosby’ in 2002.

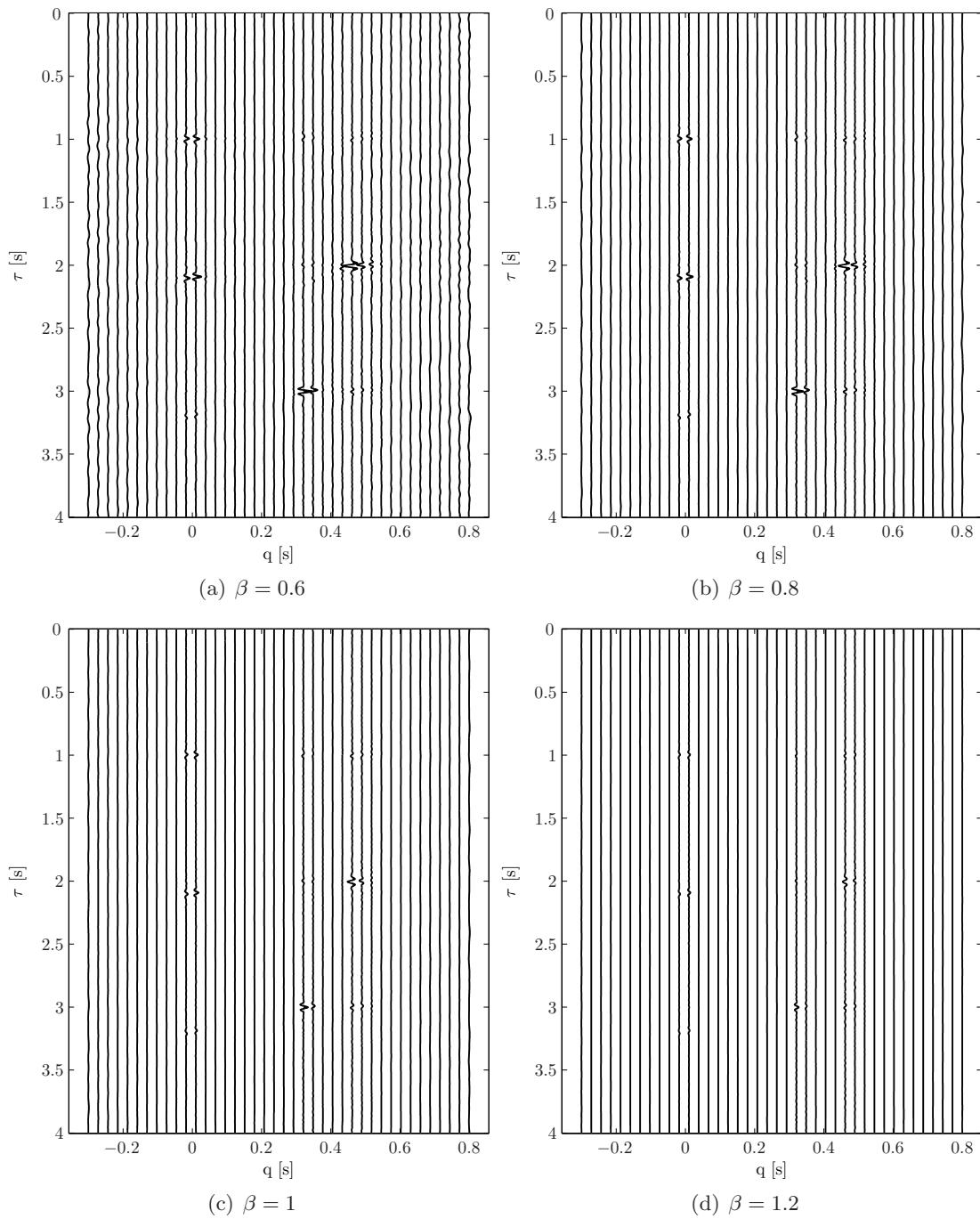


Figure 3: Parabolic Radon panel for the basis pursuit denoise solution with different values of β .

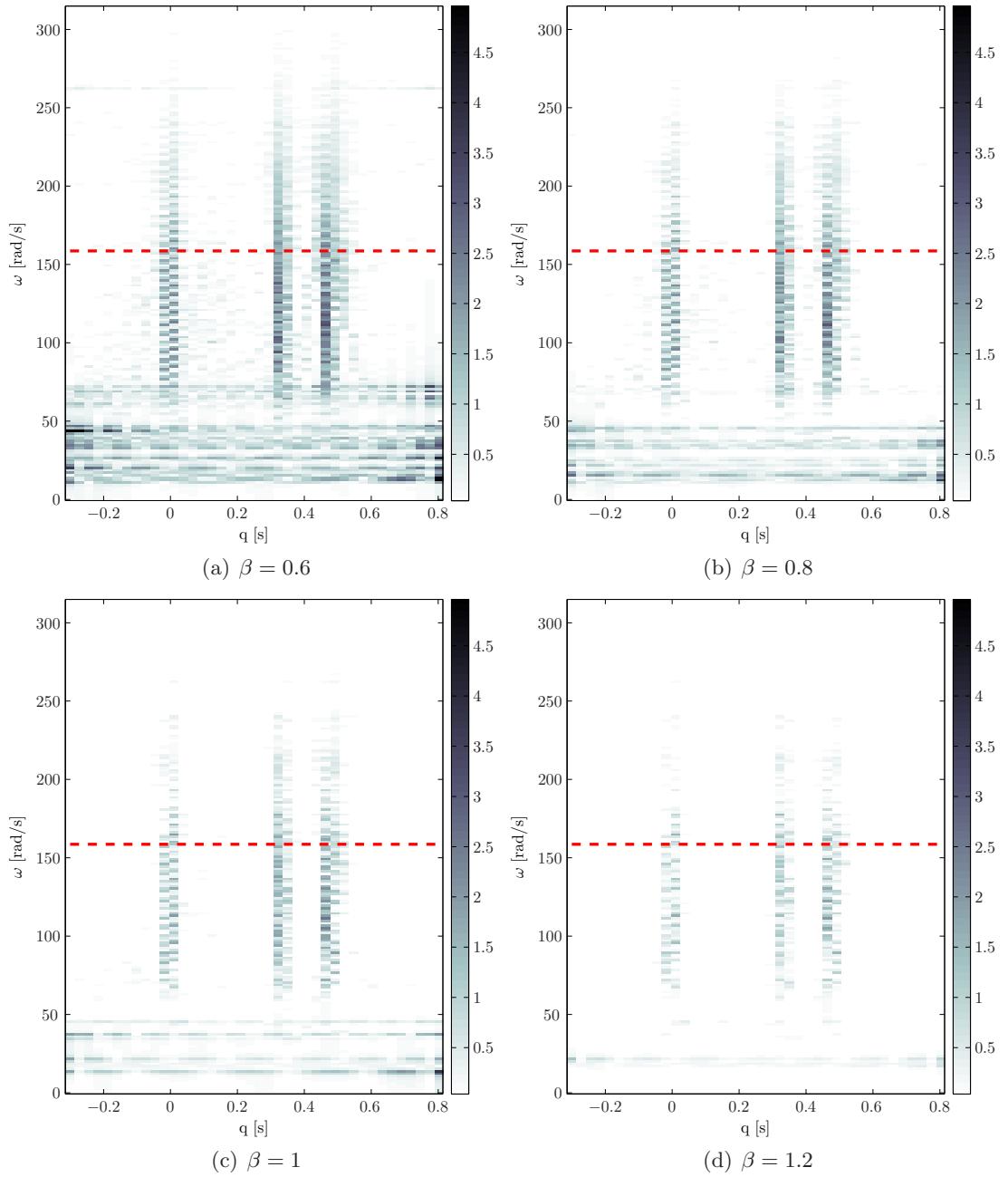


Figure 4: The ω - q domain for different values of β . Dashed red line marks a selected frequency, inspected in Figure 5.

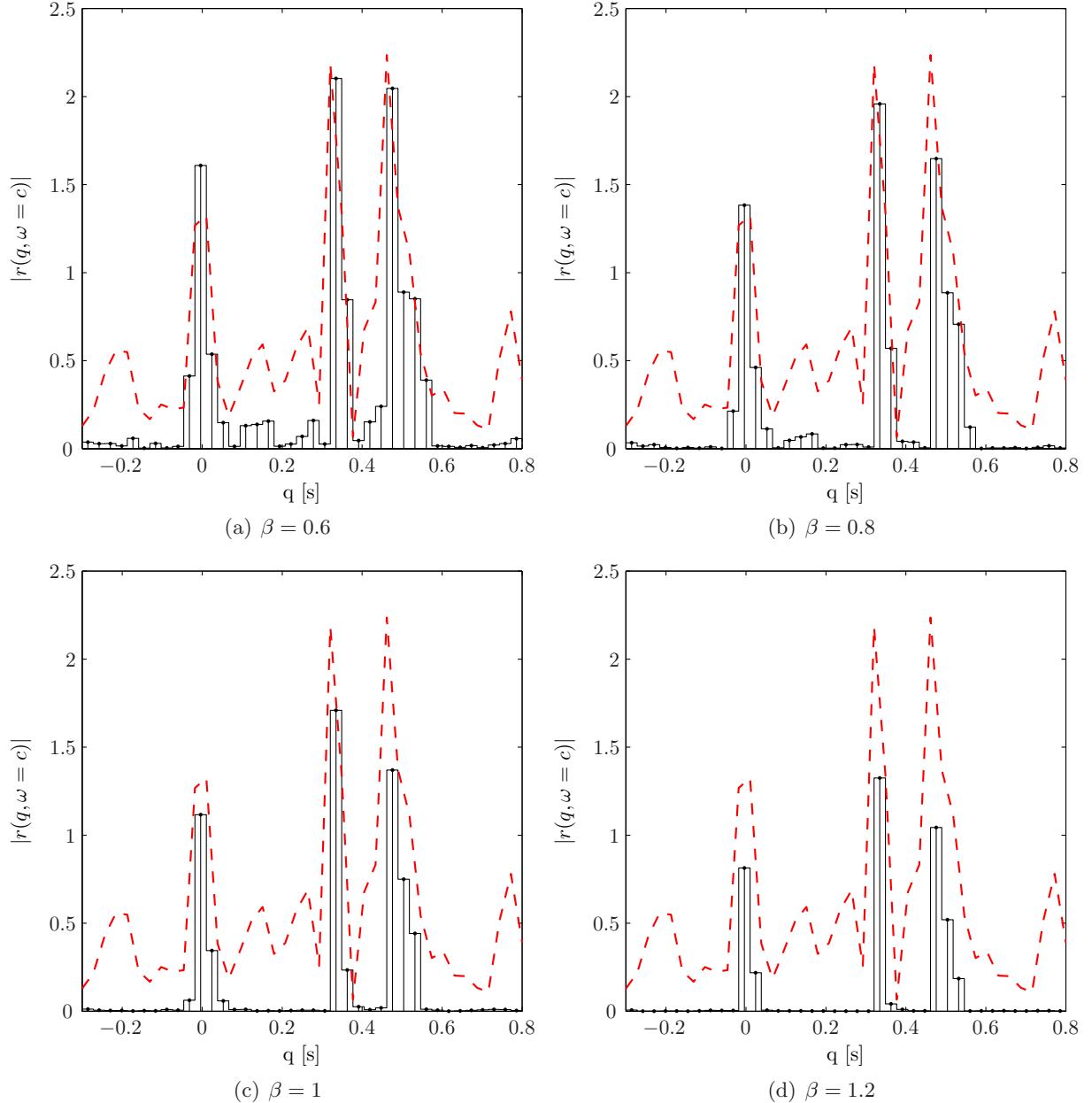


Figure 5: Selected frequency at the constant $c = 158$ rad/sec for different values of β . Dashed red line represents the damped least squares panel, while the stairs show the basis pursuit denoise solution.

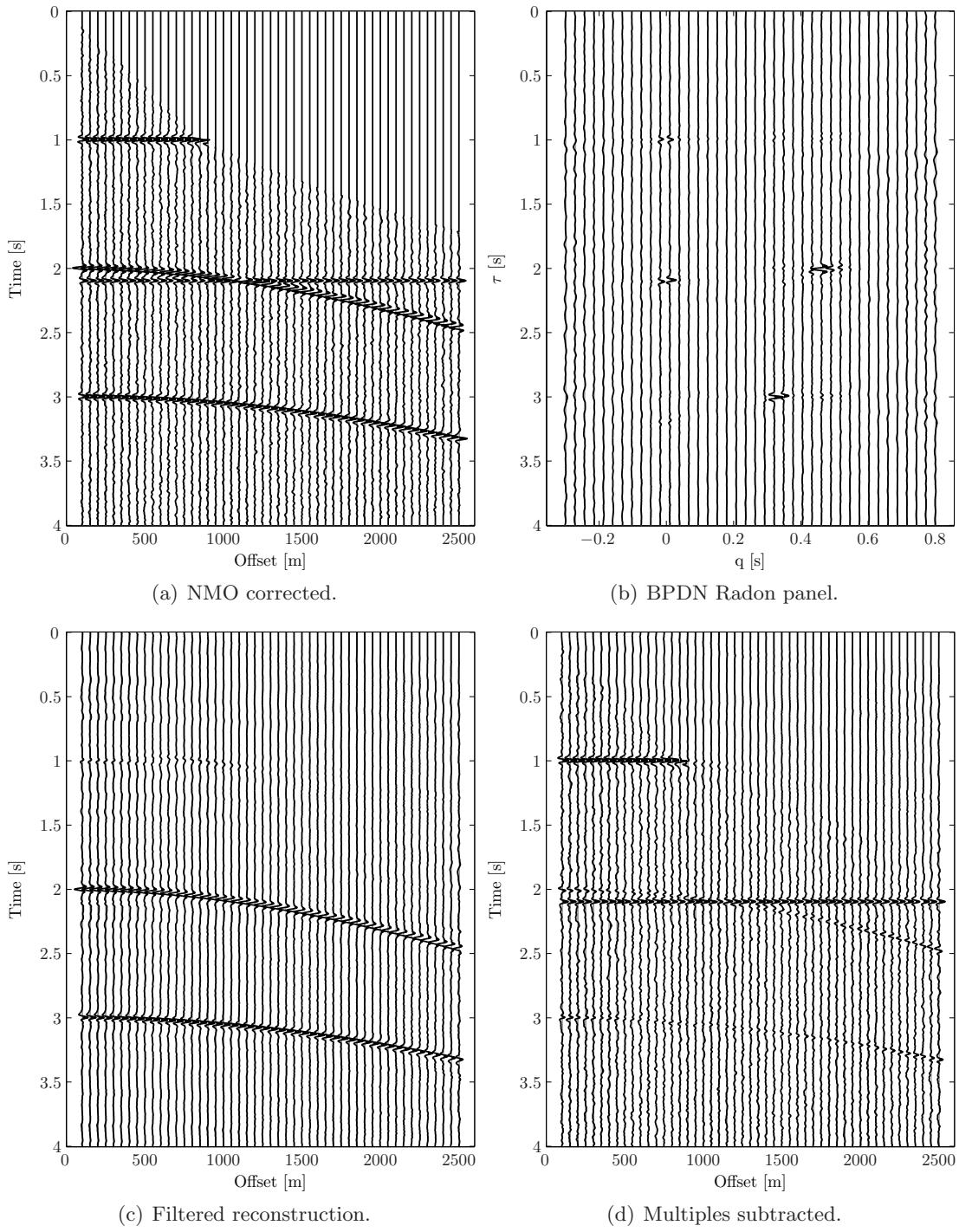


Figure 6: A Basis pursuit denoise (BPDN) Radon demultiple work flow. a) Original CMP gather from the (demo dataset in the SeismicLab package). b) NMO corrected data (including mute), subject of the Radon transform c) Reconstruction of multiples with primaries removed. d) Multiples subtracted from the original CMP gather, leaving only the primaries.

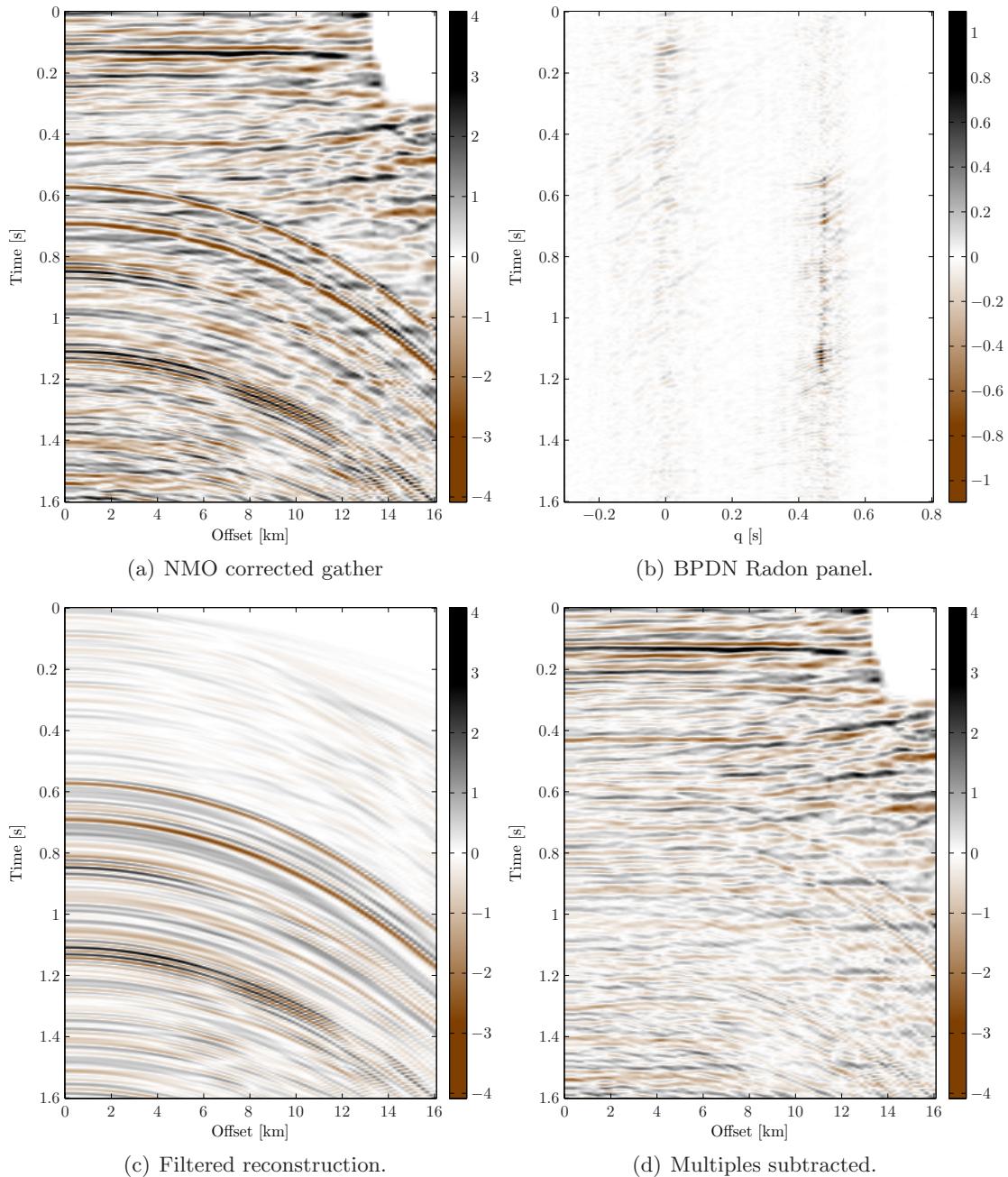


Figure 7: A real data example from the Gulf of Mexico.

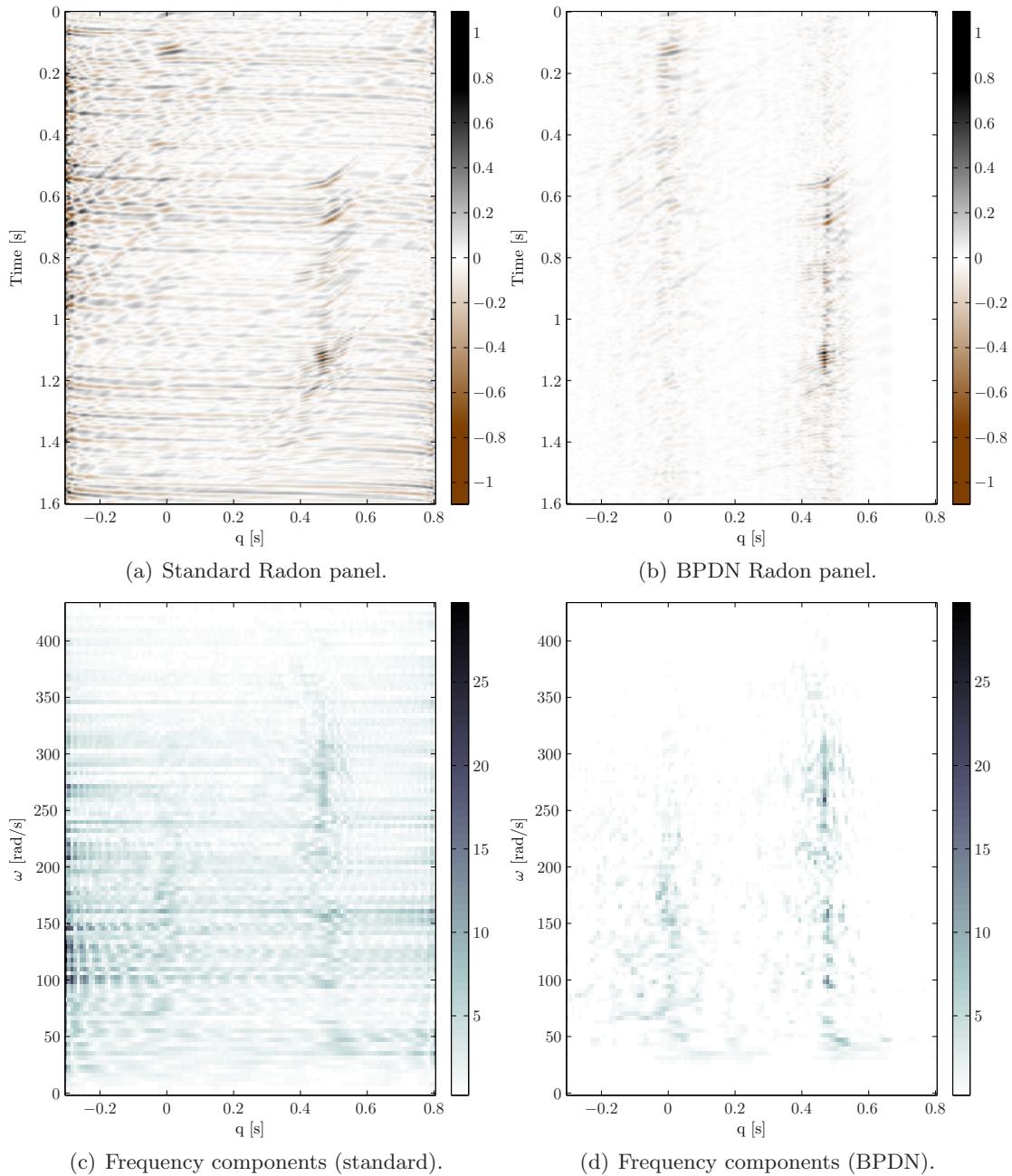


Figure 8: Real data from the Gulf of Mexico, Radon panels and their corresponding frequency components.

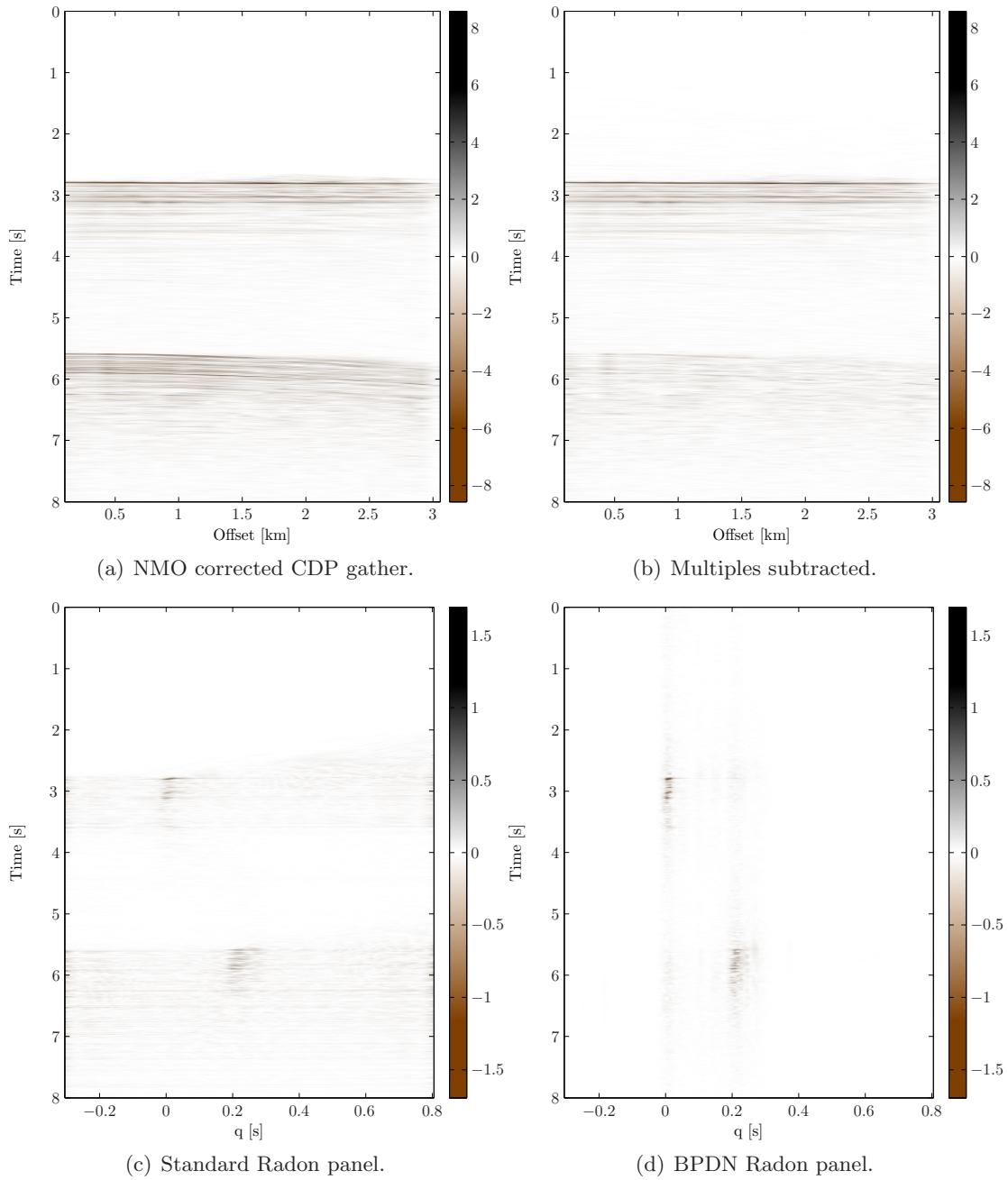


Figure 9: Real data from the Knipovich Ridge, showing clustered sea-bed (or close to) multiples.

6. Recent Developments and Greedy Methods

Previous application of the basis pursuit method (or ℓ_1 -norm penalty in general) to the geo-physical Radon transform is unknown to the author. The threshold for understanding many of the technical details, required for implementation, might be one of the reason for this. Recently several libraries have appeared, thus providing the tools for testing the algorithms on real problems without the requirements of being an expert in the field. As the techniques develop, new applications of the ℓ_1 -norm in seismic imaging could see the light of day.

Novel methods, using a greedy approach, have been implemented by the industry. Methods restricting in both q and τ directions, by some authors referred to as ‘super high-resolution’. A time-offset algorithm based on the Gauss-Seidel method and prioritized moveout parameters require quite a bit of graphical material for a good understanding, it was presented at Denver, Colorado 2004 SEG annual meeting (Ng and Perz, 2004). Detailed information on the algorithm is available from the authors in the form of slides from presentations. At the Houston, Texas 2009 SEG annual meeting, Wang and Ng (2009) presented another high-resolution algorithm. The method uses a greedy approach, solving small regions of moveout parameters using conjugate gradients. Implementation and review of these methods are not covered in the present work, but it is clear that the latest trend has been to move away from analytical solutions and instead develop greedy methods which focus energies using time-offset Radon (Yilmaz and Taner, 1994) implementation.

CHAPTER 7

Summary and Conclusions

During the recent years, time-offset implementations of the Radon transform have gained popularity, much in line with the increased computing power. Direct approaches in this manner has algorithmic advantages that can be used to construct e.g. sparse Radon panels. Chapter 1 focused on the time-invariant frequency domain formulation, and discussed a direct implementation of the time-variant hyperbolic Radon transform. The recurring problem of curves not matching the existing seismic data grid is an obvious weakness of the latter method. For the linear Radon transform, this problem has been studied by others, in the interesting setting of switching to a pseudopolar grid.

Chapter 4 focused on inversion using direct and iterative solvers for matrices with Toeplitz structure. Inversion of the time-invariant Radon transform involves computing a set of different Toeplitz systems, i.e. fast methods are essential. Numerical results showed that choice of solver depends on the system size. For relatively large systems, the conjugate gradient methods, combined with the fast Fourier transform, proved to be most efficient. It was shown how circulant preconditioners influence the eigenvalues, promoting convergence of the conjugate gradient method. The fact that none of the preconditioners (listed in the references) are able to stabilize the problem on their own, was demonstrated by a numerical example. Thus, finding a preconditioner with the correct properties to stabilize the problem in a general manner is still unresolved. This is substantiated by not having a generating function (forming the Toeplitz coefficients) in the Wiener class. A generating function of this class is needed to utilize most of the existing Toeplitz framework.

The approximate solution of the time-invariant Radon transform could be seen as a filtering of the least squares solution. In Chapter 5, the low-pass filter was studied, demonstrating how the filter characteristics change with different parameter choices. Evaluating this filter by fixed point iteration with a circulant approximation, yields an inversion method which can be evaluated by the fast Fourier transform. The fixed point recursive scheme with a circulant approximation, was successfully applied to the Radon transform, although, the Toeplitz operator needs stabilization in advance. The imposed regularization is not a desired effect.

In Chapter 6, the basis pursuit denoise method was successfully applied to the parabolic Radon transform. This showed that one-norm regularization techniques can be used to calculate a sparse approximation of the Radon panel. The proposed method was presented as an alternative to the popular Bayesian method. Tests included synthetic data, as well as field data examples from the Gulf of Mexico and Svalbard. Using the basis pursuit denoise method for linear problems in the complex plane has also gained attention in other fields of signal processing, and has interesting applications in geophysics. This is a consequence of rapid development in computing power and research on efficient optimization routines.

APPENDIX A

Alternative Frequency Domain Formulations

The linear Radon transform was not invented in the field of geophysics, however, the term slant stack was introduced by Schultz and Claerbout (1978) as a part of a migration method. A mathematical operation transforming a seismic section $f(x, t)$ to the (p, τ) -domain with the convenience of providing certain geophysical properties. Suppose p is the horizontal ray parameter and τ is the intercept time

$$p = \frac{k_x}{\omega} = \frac{1}{c_x} = \frac{\cos \alpha}{c}, \quad \tau = t - px = \frac{h}{c} \cos \alpha,$$

where α is the angle of a planar wave with respect to the horizontal axis, c is the (phase) velocity, c_x is the apparent (phase) velocity in the offset direction, k_x is the wavenumber, ω is the angular frequency and h is the depth of the plane reflector. Parameters and their physical properties are described in Table 1. Figure 1 shows geometric interpretation of the parameters.

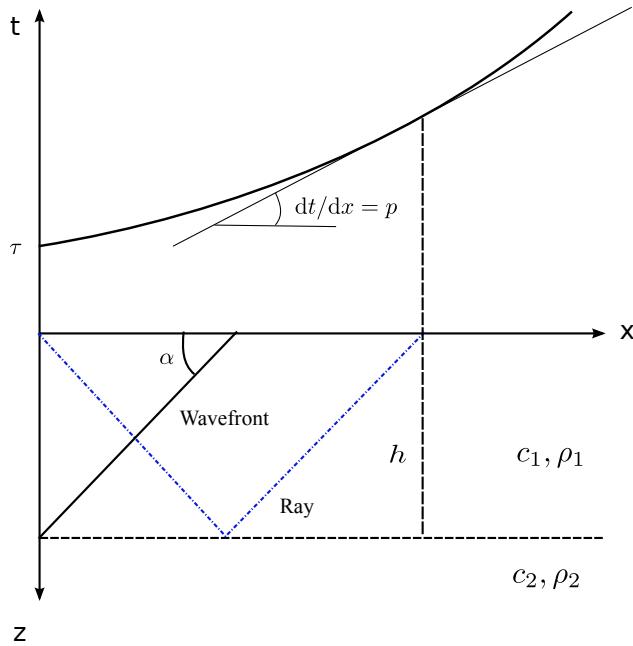


Figure 1: A geometric illustration of the different parameters involved in the linear Radon transform.

α	Angle of a planar wave wrt. the x-axis.
	Dimension, [radians or degrees]
c	The (phase)velocity in the medium.
	Dimension, meter pr. second [m/s].
c_x	Apparent (phase)velocity in the x offset direction.
	Dimension, meter pr. second [m/s].
h	Distance from surface to first horizontal seismic boundary.
	Dimension, meter [m].
$k_x = 2\pi/\lambda$	Wavenumber, the spatial analog of frequency, with wavelength λ .
	Dimension, number of wavelengths per unit distance [1/m].
$\omega = 2\pi f$	Angular frequency. With frequency f measured in hertz.
	Dimension, radians pr. second. [1/s].
p	Horizontal ray parameter, unique parameter for each ray.
	Dimension, seconds pr. kilometer [s/km].
τ	Intercept time.
	Dimension, seconds [s].

Table 1: Description of the parameters involved in the (p, τ) -transform.

1. Linear Radon in the FK-domain

A continuous formulation of the linear Radon transform is given as

$$(44) \quad r(p, \tau) = \int_{-\infty}^{\infty} f(x, t = \tau + px) \, dx.$$

For more insight, the analysis is done in the frequency domain. The two-dimensional Fourier transform of the seismic signal $f(x, t)$ gives

$$q(k_x, \omega) = \iint_{-\infty}^{\infty} f(x, t) e^{i(xk_x - \omega t)} \, dx \, dt.$$

Substituting $k_x = p\omega$, the transform is given as

$$(45) \quad q(p\omega, \omega) = \iint_{-\infty}^{\infty} f(x, t) e^{i\omega(px - t)} \, dx \, dt.$$

By introducing new variables

$$\tau = t - px \quad \text{and} \quad \xi = x,$$

the Jacobian is

$$\mathcal{J} = \frac{\partial(x, t)}{\partial(\xi, \tau)} = \begin{vmatrix} \partial x / \partial \xi & \partial x / \partial \tau \\ \partial t / \partial \xi & \partial t / \partial \tau \end{vmatrix} = 1,$$

i.e. $dx \, dt = \mathcal{J} d\xi \, d\tau$. The integral in Equation (45) is then

$$q(p\omega, \omega) = \iint_{-\infty}^{\infty} f(\xi, \tau + p\xi) e^{-i\omega\tau} \, d\xi \, d\tau.$$

Using Equation (44),

$$q(p\omega, \omega) = \int_{-\infty}^{\infty} r(p, \tau) e^{-i\omega\tau} \, d\tau,$$

and then the inverse Fourier transform of $q(p\omega, \omega)$, yields the Radon panel

$$(46) \quad r(p, \tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} q(p\omega, \omega) e^{i\omega\tau} d\omega$$

Direct numerical implementation of this method will lead to the problems with grids points not matching after the substitution $k_x = p\omega$, this can be resolved using interpolation. Trade-offs for using interpolation is possible distortion of the data and uncertainties which is hard to control. Computational speed may also decrease dramatically when using certain interpolation methods.

1.1. Inverse linear Radon transform. The mapping from (p, τ) to (x, t) is defined by an inverse operator. Starting with a Fourier transform of $f(x, t)$ with respect to the temporal variable

$$s(x, \omega) = \int_{-\infty}^{\infty} f(x, t) e^{-i\omega t} dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} q(k_x, \omega) e^{-ixk_x} dk_x.$$

Substituting $k_x = p\omega$ i.e. $dk_x/dp = \omega$, yields

$$s(x, \omega) = \frac{\omega}{2\pi} \int_{-\infty}^{\infty} q(p\omega, \omega) e^{-ip\omega x} dp.$$

Applying an inverse Fourier transform with respect to the temporal variable and keeping the integration in the conventional sense of negative to positive, the rho-filter on the form $|\omega|$ is introduced

$$f(x, t) = \mathcal{F}^{-1}\{s(x, \omega)\} = \frac{1}{4\pi^2} \iint_{-\infty}^{\infty} |\omega| q(p\omega, \omega) e^{i\omega(t-px)} d\omega dp.$$

The rho-filter is the derivative of the Hilbert transform

$$|\omega| = (i\omega) \times (-i \operatorname{sgn} \omega) = \mathcal{F}\{\rho(t)\},$$

with the filters impulse response denoted $\rho(t)$. From Equation (46) the inverse transform is given as

$$2\pi f(x, t) = \rho(t) * \int_{-\infty}^{\infty} r(p, t-px) dp = \int_{-\infty}^{\infty} \frac{d}{dt} \mathcal{H}\{r(p, t-px)\} dp,$$

where the operator $*$ denotes convolution and \mathcal{H} denotes the Hilbert transform.

2. Parabolic Radon and the Fresnel transform

In optics and diffraction theory the Fresnel transform (Gori, 1981) is a useful tool, but it has not found many significant applications in geophysics. Maeland (2000) suggests to modify the parabolic Radon transform with the substitution $t = \tau + q(x - \xi)^2$, thus introducing the Fresnel transform. If $g(\xi, q, \omega)$ is an image of $r(q, \omega)$ at $\xi = 0$ it is possible to obtain another closed-form derivation of the parabolic Radon transform.

$$g(\xi, q, \omega) = \int_{-\infty}^{\infty} f(x, \omega) e^{i\omega q(x-\xi)^2} dx = \int_{-\infty}^{\infty} f(x, \omega) \mathcal{E}(x - \xi) dx = (f * \mathcal{E})(x)$$

The integral can be evaluated as a convolution in x and at zero offset

$$r(q, \omega) = g(\xi = 0, q, \omega).$$

Applying the spatial Fourier transform yields

$$\hat{g}(k_x, q, \omega) = \int_{-\infty}^{\infty} g(\xi, q, \omega) e^{i\xi k_x} d\xi = \hat{f}(k_x) \hat{\mathcal{E}}(k_x).$$

Taking the DC component, $\xi = 0$ of $\hat{g}(k_x, q, \omega)$ gives in explicit notation

$$\begin{aligned}\hat{g}(k_x, q, \omega) &= \int_{-\infty}^{\infty} f(x, \omega) e^{ik_x x} dx \int_{-\infty}^{\infty} e^{i\omega qx^2} e^{ik_x x} dx \\ &= \sqrt{(i\pi/\omega q)} e^{-ik_x^2/4\omega q} \int_{-\infty}^{\infty} f(x, \omega) e^{ik_x x} dx.\end{aligned}$$

This is a simplification which is possible to implement numerically without any elaborate techniques. An inverse spatial Fourier transform involving only the DC component yields

$$r(q, \omega) = g(\xi = 0, q, \omega) = \int_{-\infty}^{\infty} \hat{g}(k_x, q, \omega) dk_x.$$

The parabolic Radon transform has been evaluated in the spatial wavenumber domain by the Fresnel transform. Maeland (2005) makes use of this result to establish a relation between zero offset migration within the paraxial approximation of the wave equation and the parabolic Radon transform. His paper focuses on structural stability and how the different images (methods) are related.

APPENDIX B

Algorithms, Code Examples and Parameters

1. Algorithms

This section contains different algorithms which was discussed in the text.

ALGORITHM 2 (The Gradient Decent Method). *Given $\mathbf{b} \in \mathbb{R}^m$ and $\mathbf{A} \in \mathbb{R}^{m \times m}$ is assumed to be positive definite and symmetric.*

```

 $k = 0, x_0 = 0, r_0 = b$ 
while  $r_k \neq 0$  do
     $k = k + 1$ 
     $\alpha_k = r_{k-1}^T r_{k-1} / r_{k-1}^T A r_{k-1}$ 
     $x_k = x_{k-1} + \alpha_k r_{k-1}$ 
     $r_k = b - Ax_k$ 
end while

```

ALGORITHM 3 (Levinson, Golub and Van Loan (1989)). *Given $\mathbf{b} \in \mathbb{R}^m$ and real numbers $1 = r_0, r_1, \dots, r_n$ such that $\mathbf{T} = (r_{i-j}) \in \mathbb{R}^{m \times m}$ is positive definite, the following algorithm computes $\mathbf{x} \in \mathbb{R}^m$ such that $\mathbf{T}\mathbf{x} = \mathbf{b}$.*

```

 $y(1) = -r(1); x(1) = b(1); \beta = 1; \alpha = -r(1)$ 
for  $k = 1 : n - 1$  do
     $\beta = (1 - \alpha^2)\beta$ 
     $\mu = (b(k+1) - r(1:k)^T x(k:-1:1)) / \beta$ 
     $v(1:k) = x(1:k) + \mu y(k:-1:1)$ 
     $x(1:k) = v(1:k)$ 
     $x(k+1) = \mu$ 
    if  $k < n - 1$  then
         $\alpha = -(r(k+1) + r(1:k)^T y(k:-1:1)) / \beta$ 
         $z(1:k) = y(1:k) + \alpha y(k:-1:1)$ 
         $y(1:k) = z(1:k)$ 
         $y(k+1) = \alpha$ 
    end if
end for

```

Note that the MATLAB implementation given in Listing B.1 is a modified version for complex Hermitian Toeplitz systems, including correction of a misprint from Golub and Van Loan (1989).

2. MATLAB Code

To facilitate the use of algorithms, discussed in Chapter 4 and 5, this section includes MATLAB functions used for the numerical experiments.

Listing B.1: Levinson.

```
1 % LEVINSON_SOLVER: Solves a Hermitian Toeplitz system  $Tz = b$ .
2 %
3 % [z] = levinson_solver(col, b)
4 %
5 %     IN:    col: the first column of the Toeplitz matrix
6 %            b: the right-hand side vector
7 %     OUT:   z: the complex solution vector
8 %
9 %     Reference: Golub, Van Loan, 1989, Matrix Computations: Second Edition
10 %
11 %     Comments: A modified version of the algorithm in Golub, Van Loan. Works
12 %               for a complex Hermitian Toeplitz systems. Also including a
13 %               misprint correction.
14 %
15 function [z] = levinson_solver(col, b)
16 konst = col(1);
17 col = col ./ konst;
18 b = b ./ konst;
19 r = col(2:end)';
20 n = length(r)+1;
21 z = zeros(n,1); y = zeros(n,1);
22
23 y(1) = -r(1);
24 z(1) = b(1);
25 beta = 1;
26 alpha = -r(1);
27
28 for k = 1:(n - 1)
29     beta = (1 - alpha * conj(alpha)) * beta;
30     mu = (b(k+1) - r(1:k).' * z(k:-1:1)) / beta;
31     z(1:k+1) = [z(1:k) + mu * conj(y(k:-1:1)) ; mu];
32
33 if k < (n - 1)
34     alpha = -(r(k+1) + r(1:k).' * y(k:-1:1)) / beta;
35     y(1:k+1) = [y(1:k) + alpha * conj(y(k:-1:1)) ; alpha];
36 end
37 end
38 end
```

Listing B.2: Conjugate Gradient for a Hermitian Toeplitz system.

```
1 % CG_FFT: The conjugate gradient algorithm with a Hermitian Toeplitz system Tz = b.
2 %
3 % [z] = cg_fft(col, b, tol)
4 %
5 %      IN: col: the first column of the Toeplitz matrix
6 %            b: the right-hand side vector
7 %            tol: tolerance limit for convergence
8 %
9 %      OUT: z: the complex solution vector
10 %
11 % Comments: This algorithm augments a M x M Hermitian Toeplitz matrix into
12 %            a 2M x 2M circulant matrix before solving the matrix-vector
13 %            multiplication by the fast Fourier transform.
14 %
15 function [z] = cg_fft(t, b, tol)
16     n = length(t);
17     z = zeros(n, 1);
18     r = b;
19     d = [r ; zeros(n-1, 1)];
20
21     colhat = fft([t ; conj(t(end:-1:2))]);
22     while norm(r) > tol
23         r_cur = r;
24         g = ifft(colhat .* fft(d));
25
26         alpha = (r' * r) / (d(1:n)' * g(1:n));
27         z = z + alpha * d(1:n);
28         r = r - alpha * g(1:n);
29         beta = (r' * r) / (r_cur' * r_cur);
30         d(1:n) = r + beta * d(1:n);
31     end
32 end
```

Listing B.3: Complex biconjugate gradient method.

```
1 % COMPLEX_BI_CG: The complex-biconjugate gradient algorithm with system  $Az = b$ .
2 %
3 % [z] = complex.bi_cg(A, b, tol)
4 %
5 %      IN: A: the complex operator (matrix)
6 %            b: the right-hand side vector
7 %            tol: tolerance limit for convergence
8 %
9 %      OUT: z: the complex solution vector
10 %
11 %      Reference: Jacobs, D. A. H., 1986, A Generalization of the Conjugate-Gradient
12 %                  method to solve Complex Systems: Journal of Numerical Analysis,
13 %                  6, 447–452
14 %
15 function [z] = complex.bi_cg(A, b, tol)
16     [n m] = size(A);
17     z = zeros(n, 1);
18     r = b - A * z;
19     rhat = conj(r);
20     p = r;
21     phat = conj(p);
22
23 while norm(r) > tol
24     g = A * p;
25     ghat = A' * phat;
26     alpha = (rhat' * r) / (phat' * g);
27     z = z + alpha * p;
28     r = r - alpha * g;
29     rhat = rhat - conj(alpha) * ghat;
30     beta = - (ghat' * r) / (phat' * g);
31     p = r + beta * p;
32     phat = rhat + conj(beta) * phat;
33 end
34 end
```

Listing B.4: Preconditioned (circulant, fft) gradient method.

```
1 % PCG.FFT.CIRCULAR: The preconditioned conjugate gradient algorithm solving a
2 % Hermitian Toeplitz system  $Tz = b$ , using a circulant preconditioner.
3 %
4 % [z] = pcg_fft_circular(col, b, tol, c)
5 %
6 % IN: col: the first column of the Toeplitz matrix
7 %      b: the right-hand side vector
8 %      tol: tolerance limit for convergence
9 %      c: the first column of the circulant approximation.
10 %
11 % OUT: z: the complex solution vector
12 %
13 % Comments: This algorithm augments a  $M \times M$  Hermitian Toeplitz matrix into
14 %            a  $2M \times 2M$  circulant matrix before solving the matrix-vector
15 %            multiplication by the fast Fourier transform. Matrix-vector
16 %            division (inversion) with the circulant approximation is also
17 %            done by FFT, without the need for augmentation.
18 %
19 function [z] = pcg_fft_circular(t, b, tol, c)
20 m = length(t);
21 z = zeros(m, 1);
22 r = b;
23 chat = fft(c);
24 p = ifft(fft(r) ./ chat);
25 d = [p ; zeros(m-1, 1)];
26
27 colhat = fft([t conj(t(end:-1:2))]).';
28
29 while norm(r) > tol
30     g = ifft(colhat .* fft(d));
31     p_prev = p;
32     r_prev = r;
33     alpha = p' * r / (d(1:m)' * g(1:m));
34     z = z + alpha * d(1:m);
35     r = r - alpha * g(1:m);
36     p = ifft(fft(r) ./ chat);
37     beta = p' * r / (p_prev' * r_prev);
38     d(1:m) = p + beta * d(1:m);
39 end
40
41 end
```

Listing B.5: Circulant approximation, classical iteration.

```
1 % CAFFT_SOLVER: Solves the classical iteration on the form  $C z(n+1) = b - (H-C) z(n)$ ,  
2 % where  $C$  is a circulant approximation to  $H$ .  
3 %  
4 % [z] = ca_fft_solver(t, c, b, tol)  
5 %  
6 % IN: t: first column of Toeplitz vector  $Q(:,1)$ ,  
7 % from the residual,  $Q = H - C$   
8 % b: right hand side of the linear system  
9 % c: first column of the circulant approximation  
10 % tol: tolerance limit for convergence  
11 %  
12 % OUT: the complex solution vector  
13 %  
14 % Comments: This function is tested with the Generalized Jackson  
15 % kernel. Other circulant approximation might yield a  
16 % different structure, hence the Toeplitz augmentation  
17 % could require modification (generalization).  
18 %  
19 function [z] = ca_fft_solver(t, b, c, tol)  
20 m = length(c);  
21 z = zeros(m,1);  
22 r = b;  
23 chat = fft(c);  
24 cb = fft(b) ./ chat;  
25 that = fft([t conj(t(end: -1:2))]).';  
26  
27 while norm(r) > tol  
28 z_p = z;  
29 % Interpolate the Fouier domain by padding zeros.  
30 zpad = fft(z, 2*m-1);  
31 % Matrix-vector multiplication in the Fourier domain.  
32 g = ifft(that .* zpad);  
33 % Fourier transform only the solution.  
34 sol = fft(g(1:m));  
35 % Compute the fixed point iteration.  
36 z = ifft(cb - sol ./ chat);  
37 r = z_p - z;  
38 end  
39 end
```

Listing B.6: Circulant approximation.

```
1 % CIRC_GEN: Generate a circulant preconditioner based on a Toeplitz column.
2 %
3 % [c] = circ_gen(t)
4 %
5 %      IN: t: the first column of the Toeplitz matrix
6 %
7 %      OUT: c: circulant approximation with Hermitian symmetry (vector)
8 %
9 %      Reference: Chan, T. F., 1988, An optimal circulant preconditioner for
10 %                  Toeplitz systems: SIAM Journal on Scientific and Statistical
11 %                  Computing, 9, 766-771
12 %
13 %      Comments: The MATLAB command circshift (shifts an array circularly) could be
14 %                  used to generate the complete circulant matrix.
15 %
16 function [c] = circ_gen(t)
17     m = length(t);
18     c = zeros(m, 1);
19     c(1) = t(1);
20     k = 1:m-1;
21     c(k+1) = ((m - k).*t(k+1) + k.*conj(t(m+1-k))) / m;
22 end
```

Listing B.7: Cholesky factorization.

```
1 % CHOLESKY_SOLVER: Solves a linear system  $Tz = b$  by Cholesky factorization.
2 %
3 % [z] = cholesky_solver(t, b)
4 %
5 %      IN:    t: the first column of the Toeplitz matrix
6 %            b: the right-hand side vector
7 %
8 %      OUT:   z: the complex solution vector.
9 %
10 function [z] = cholesky_solver(t, b)
11     T = toeplitz(t);
12     C = chol(T);
13     z = C \ (C' \ b);
14 end
```


Bibliography

- Anderson, E., Bai, Z., Bischof, C., and Blackford, 1999, Lapack users' guide: Society for Industrial Mathematics.
- Apeland, G., 2001, En analyse av den parabolske radon transformasjonen: Master's thesis, University of Bergen.
- Aster, R. C., Borchers, B., and Thurber, C. H., 2005, Parameter estimation and inverse problems: Academic Press.
- Averbuch, A., Coifman, R. R., Donoho, D. L., Israeli, M., and Walden, J., 2001, Fast slant stack: A notion of radon transform for data in a cartesian grid which is rapidly computable, algebraically exact, geometrically faithful and invertible: SIAM Scientific Computing.
- Berg, E. v. d., and Friedlander, M. P., 2008, Probing the pareto frontier for basis pursuit solutions: SIAM Journal on Scientific Computing, **31**, 890–912.
- Beylkin, G., 1987, Discrete radon transform: IEEE Transactions on Acoustics Speech and Signal Processing, **35**, 162–172.
- Bjorck, A., 1996, Numerical methods for least squares problems: Society for Industrial Mathematics.
- Boyd, S. P., and Vandenberghe, L., 2004, Convex optimization: Cambridge University Press.
- Brandwood, D. H., 1983, A complex gradient operator and its application in adaptive array theory: IEE Proceedings H Microwaves, Optics and Antennas, **130**, 11–16.
- Brown, J. W., and Churchill, R. V., 1974, Complex variables and applications: McGraw-Hill New York.
- Chan, R. H., and Jin, X. Q., 2007, An introduction to iterative toeplitz solvers: Society for Industrial and Applied Mathematics.
- Chan, R. H., and Ng, M. K., 1996, Conjugate gradient methods for toeplitz systems: SIAM Review, **38**, 427–482.
- Chan, R. H., Ng, M. K., and Yip, A. M., 2001a, A survey of preconditioners for ill-conditioned toeplitz systems: Contemporary Mathematics, **281**, 175–192.
- 2001b, The best circulant preconditioners for hermitian toeplitz systems: SIAM Journal on Numerical Analysis, **38**, 876–896.
- Chan, R. H., Ng, M. K., and Yip, A. M., 2002, The best circulant preconditioners for hermitian toeplitz systems ii: The multiple-zero case: Numerische Mathematik, **92**, 17–40.
- Chan, T. F., 1988, An optimal circulant preconditioner for toeplitz systems: SIAM Journal on Scientific and Statistical Computing, **9**, 766–771.
- Chen, S. S., Donoho, D. L., and Saunders, M. A., 2001, Atomic decomposition by basis pursuit: SIAM review, **43**, 129–159.
- Deans, S. R., 1983, The radon transform and some of its applications: A Wiley-Interscience Publication, New York.
- Dev, A., and McMechan, G. A., 2009, Spatial antialias filtering in the slowness-frequency domain: Geophysics, **74**, 35–42.
- Devaney, R. L., 1994, Complex dynamical systems: the mathematics behind the mandelbrot and julia sets: Amer Mathematical Society.

- Fletcher, R., 1975, Conjugate gradient methods for indefinite systems: Numerical Analysis, **506**, 73–89.
- Folland, G. B., 2009, Fourier analysis and its applications: AMS Bookstore.
- Fraleigh, J. B., 1994, A first course in abstract algebra: Addison-Wesley.
- Golub, G. H., and Van Loan, C. F., 1989, Matrix computations: The Johns Hopkins University Press, second edition.
- Gori, F., 1981, Fresnel transform and sampling theorem.: Optics Communications, **39**, 293–297.
- Gray, R. M., 2006, Toeplitz and circulant matrices: A review: Now Pub.
- Grenander, U., and Szegö, G., 1958, Toeplitz forms and their applications: Chelsea Publishing Company.
- Hampson, D., 1986, Inverse velocity stacking for multiple elimination: J. Can. Soc. Expl. Geophys., **22**, 44–55.
- Hansen, P. C., 1998, Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion: Society for Industrial Mathematics.
- Hansen, P. C., 2002, Deconvolution and regularization with toeplitz matrices: Numerical Algorithms, **29**, 323–378.
- Helgason, S., 1999, The radon transform: Birkhauser.
- Hennenfent, G., and Herrmann, F. J., 2005, Sparseness-constrained data continuation with frames: Applications to missing traces and aliased signals in 2/3-d: SEG Expanded Abstracts.
- Hennenfent, G., and Herrmann, F. J., 2008, Simply denoise: Wavefield reconstruction via jittered undersampling: Geophysics, **73**, V19–V28.
- Hestenes, M. R., 1973, Iterative methods for solving linear equations: Journal of Optimization Theory and Applications, **11**, 323–334.
- Hugonnet, P., and Canadas, G., 1995, Aliasing in the parabolic radon transform: 65th Meeting, Society of Exploration Geophysicists, Expanded Abstracts, 1366–1369.
- Jacobs, D. A. H., 1986, A generalization of the conjugate-gradient method to solve complex systems: IMA Journal of Numerical Analysis, **6**, 447.
- Jain, A. K., 1989, Fundamentals of digital image processing: Prentice-Hall, Inc.
- Johnson, R. A., and Wichern, D. W., 1998, Applied multivariate statistical analysis: Prentice Hall Englewood Cliffs, NJ.
- Joly, P., and Meurant, G., 1993, Complex conjugate gradient methods: Numerical Algorithms, **4**, 379–406.
- Karmeshu, J., 2003, Entropy measures, maximum entropy principle and emerging applications: Springer-Verlag.
- Kelley, C. T., 1995, Iterative methods for linear and nonlinear equations: Society for Industrial Mathematics.
- Kostov, C., 1990, Toeplitz structure in slant-stack inversion: 60th Meeting, Society of Exploration Geophysicists, Expanded Abstracts, 1618–1621.
- Kreutz-Delgado, K., 2005, The complex gradient operator and the cr-calculus: Citeseer.
- Lay, D. C., 2006, Linear algebra and its applications: Addison-Wesley Reading, Mass, third edition.
- Li, H., and Adali, T., 2008, Complex-valued adaptive signal processing using nonlinear functions: EURASIP Journal on Advances in Signal Processing, **2008**, 1–9.
- Lorentz, G. G., 2005, Approximation of functions: Chelsea Publishing Company, Incorporated.
- Maeland, E., 1998, Focusing aspects of the parabolic radon transform: Geophysics, **63**, 1708–1715.
- Maeland, E., 2000, An overlooked aspect of the parabolic radon transform: Geophysics, **65**, 1326–1329.
- Maeland, E., 2003, Disruption of seismic images by the parabolic radon transform: Geophysics, **68**, 1060–1064.

- Maeland, E., 2004, Sampling, aliasing, and inverting the linear radon transform: *Geophysics*, **69**, 859–861.
- Maeland, E., 2005, On seismic imaging and structural stability: *Journal of Seismic Exploration*, **14**, 295–301.
- Ng, M., and Perz, M., 2004, High resolution radon transform in the tx domain using intelligent prioritization of the gauss-seidel estimation sequence: 74th Meeting, Society of Exploration Geophysicist, Expanded Abstracts, 2160–2163.
- Okin, J., 1986, Linear and nonlinear deconvolution problems: Ph.D. thesis, Rice University, Houston TX.
- Ramm, A. G., and Katsevich, A. I., 1996, *The radon transform and local tomography*: CRC.
- Sacchi, M. D., and Ulrych, T. J., 1995, High-resolution velocity gathers and offset space reconstruction: *Geophysics*, **60**, 1169–1177.
- Schonewille, M. A., and Duijndam, A. J. W., 2001, Parabolic radon transform, sampling and efficiency: *Geophysics*, **66**, 667–678.
- Schultz, P. S., and Claerbout, J. F., 1978, Velocity estimation and downward continuation by wavefront synthesis: *Geophysics*, **43**, 691–714.
- Shewchuk, J. R., 1994, An introduction to the conjugate gradient method without the agonizing pain: Technical Report CMU-CS-94-125.
- Strang, G., 1986, A proposal for toeplitz matrix calculations.: *Stud. Appl. Math.*, **74**, 171–176.
- Thorson, J. R., and Claerbout, J. F., 1985, Velocity-stack and slant-stack stochastic inversion: *Geophysics*, **50**, 2727–2741.
- Thorson, J. R., 1984, Velocity-stack and slant stack inversion methods: Ph.D. thesis, Stanford University.
- Tibshirani, R., 1996, Regression shrinkage and selection via the lasso: *Journal of the Royal Statistical Society. Series B (Methodological)*, **58**, 267–288.
- Trad, D., Ulrych, T., and Sacchi, M., 2002, Accurate interpolation with high-resolution time-variant radon transforms: *Geophysics*, **67**, 644–656.
- Trad, D., Ulrych, T., and Sacchi, M., 2003, Latest views of the sparse radon transform: *Geophysics*, **68**, 386–399.
- Tropp, J. A., and Wright, S. J., 2010, Computational methods for sparse solution of linear inverse problems: *Proceedings of the IEEE*.
- Turner, G., 1990, Aliasing in the tau-p transform and the removal of spatially aliased coherent noise: *Geophysics*, **55**, 1496–1503.
- Van Den Bos, A., 1994, Complex gradient and hessian: *IEE Proceedings-Vision, Image and Signal Processing*, **141**, 380–383.
- Wang, J., and Ng, M., 2009, Greedy least-squares and its application in radon transforms: 79th Meeting, Society of Exploration Geophysicists, Expanded Abstracts.
- Wirtinger, W., 1927, Zur formalen theorie der funktionen von mehr komplexen veränderlichen: *Mathematische Annalen*, **97**, 357–375.
- Yaroslavsky, L. P., 1997, Efficient algorithm for discrete sinc interpolation: *Applied Optics*, **36**, 460–463.
- Yilmaz, O., and Taner, M. T., 1994, Discrete plane-wave decomposition by least-mean-square-error method: *Geophysics*, **59**, 973–982.
- Yilmaz, O., 1988, Velocity-stack processing: SEG Technical Program Expanded Abstracts, **7**, 1013–1016.
- Zhou, B., and Greenhalgh, S. A., 1994, Linear and parabolic tau-p transforms revisited: *Geophysics*, **59**, 1133–1149.