NTNU
Innovation and Creativity

Spatial variability in seismic depth tomography

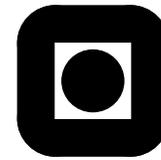Frédéric-Nicolas Dumont-Kristiansen

Master of Science in Physics and Mathematics
Submission date: January 2007
Supervisor: Jo Eidsvik, MATH

Problem Description

Estimation of seismic velocity parameters though reflection tomography is generally an ill-conditioned problem that needs to be constrained. We want to impose constraints on the spatial variability of the estimated velocity parameters through the incorporation of directional penalty functions that increase in value as the local spatial wavenumbers of the solution increase, consistent with available information on the location and dip of major reflection interfaces (where rapid changes in velocity can be assumed to occur).

Assignment given: 20. August 2006
Supervisor: Jo Eidsvik, MATH

MASTER'S THESIS

FOR

STUD.TECHN. Frederic Dumont-Kristiansen

FACULTY OF INFORMATION TECHNOLOGY, MATHEMATICS AND ELECTRICAL ENGINEERING

NTNU

Date due: January 14, 2007

Discipline: Mathematical sciences

Title: "Spatial variability constraints in seismic depth tomography"

Purpose of the work: Estimation of seismic velocity parameters through reflection tompography is generally an ill-conditionned problem that needs to be constrained. We want to impose constraints on the spatial variability of the estimated velocity parameters through the incorporation of directional penalty functions that increase in value as the local spatial wavenumbers of the solution increase, consistent with available information on the location and dip of major reflection interfaces (where rapid changes in velocity can be assumed to occur).

This diploma thesis is to be carried out at the Department of Mathematical Sciences under guidance of Associate Professor Jo Eidsvik.

Trondheim, August 20, 2006.

Trond Digernes	Jo Eidsvik
Instituttleder	Associate Professor
Dept. of Mathematical Sciences	Dept. of Mathematical Sciences

1   Preface

This thesis is a part of my Master degree and has been written at NTNU, Faculty of Information Techonology, Mathematics and Electrical Engineering at the departement of mathematics for Holberg Exploration AS (Oslo). Even if the thesis cover a wide range of subjects from my study, I want to inform the reader that I did not have any background within seismic sciences. The work on this thesis has also been challenging but rewarding since I learned a lot about tomography and seismic velocity data processing.

I would like to thank Jo Eidsvik who supervised my work during the project, Olav Holberg for providing expertise and giving me the opportunity to work with tomography sciences.

Frederic Dumont-Kristiansen

2 Abstract

The location of a reflector or medium in the subsurface is correlated with the high wavenumbers or high frequencies in the velocity field. Indeed, the determination of the high frequencies of the velocity field both normally and laterally is the key step for improving seimic data and then get a better insight of the position of a reflector in the subsurface.

This project focus on the velocity data processing part involved in seismic tomography. We describe, compare and implement several highpass operators based on finite-difference and the Hamming window in order to filter a seismic velocity dataset. In fact, we study their behaviour in the frequency domain by examining their spectrums. The main contribution of this project is to construct two dimensional anisotropic operators by rotating a one dimensional operator based on linear interpolation.

We test all the operators on a synthetic seismic velocity dataset and compare the results obtained between the isotropic filtering method and the anisotropic filtering method. We show that anisotropic filters can be useful in certain geological circumstances.

Finally we attempt to scale the different operators in order to fully incorporate them in the seismic tomography inversion problem by using a Bayesian method. We show that it is possible to decide the strength of the constraint in which we want to filter the seismic dataset by using a regularization parameter.

Contents

3   Introduction

The accurate knowledge of the velocity field is essential in tomography sciences and more specifically seismic tomographic sciences, to determine the locations and construct an image of the shape of the different mediums in the subsurface. The estimation of the velocity parameters is subject to the iterative resolution of an inversion problem (like the backprojection algorithm, the damped least square method) that needs to be constrained.

Without any constraints, the solutions of the inversion problem are often erroneous and the model obtained reflect partially the reality. A lot of methods have been used to constrain the solutions of the inversion problem; among them, the use of penalty functions or filters. The use of filters or penalty functions will help us to parameterize the inversion process in order to get the desired constrained solutions.

We will suppose along this project that the velocity field is isotropic. Our goal in this project is to process the seismic velocity data which are the solutions of the inversion problem. In fact we are going to deal with the final step which is involved in the seismic tomographic sciences: make velocity constraints within the inversion processing. Moreover we want to filter the data such that low wavenumbers will be marked down and high wavenumbers will be accentuated where rapid changes on the velocity field are allowed. Most of the two dimensional (2D) filters are rotation invariant and said to be isotropic (Barner and Arce, 2004). We want to use 1D filter generalized to 2D filter to make the filtering process dependent of the direction of the velocity and afterwards rotation variant. Indeed it has been attempted to construct those directional dependent filters by using anisotropic diffusion (Alvarez, Lions, and Morel, 1992).

We will use operators to determine the rapid variations of the velocity at its location and afterwards we will show that it is possible to parameterize constraints on the velocity process by rotating 1D penalty operators with a certain angle of rotation. Indeed we will construct 2D penalty operators by rotating a 1D penalty operator, using linear interpolation. We will attempt to show the usefulness of those kind of penalty operators in some certain geological circumstances. We will use two operators, the first one is based on finite-difference and the other is based on the Hamming window.

We will first introduce in Section 4 a little insight on the seismic tomography problem and then in Section 5, some signal processing theory on what the project will be based on. We will in Section 6 study and compare six operators, the first three operators based on finite-difference and the three others based on the Hamming window operator, by computing their discrete Fourier transforms for the 1D and 2D wavenumber domain. In Section 7, we will apply rotations on the 1D operator studied in Section 6 by using linear interpolation and check their behaviours. In Section 8, we will use all the different operators studied in the previous sections on a synthetic seismic velocity dataset. Finally in Section 9, we will attempt to give an explanation on the manner to scale those

operators such that they can be fully incorporated into the seismic tomography inversion process.

4   Motivation: seismic tomography

Tomography (from the Greek 'tomos', or 'section') means a picture of a cross section of an object. In our case the object is the earth structure. In practice, the term denotes determining the internal properties of an object from external measurements on rays that pass through the objects. The time (traveltime) for the wave or ray to come to a seismic receiver (geophone) can be used to compute the wave speed through the earth. By combining analyses from many rays sent or seismics from multiple angles, it is possible to construct an image of the structure of the earth. For example, we can see on Figure 1, four rays sent with different angles $\theta_1 = 5°$, $\theta_2 = 10°$, $\theta_3 = 15°$, $\theta_4 = 20°$, refracted through 4 different mediums and reflected on the reflector located at -1400 m(we made the assumption that the different mediums are horizontal in this example). The different
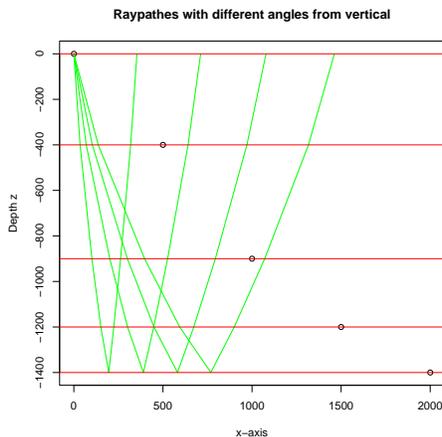

Figure 1: 1D model with angles of reflection from vertical $\theta_1 = 5°$, $\theta_2 = 10°$, $\theta_3 = 15°$, $\theta_4 = 20°$ of the raypathes (represented by green lines) with one reflector located at -1400 m. The crossing between two mediums is represented by a red line.

mediums in the subsurface can be seen as reflectors and then can be located by using Snell's law to find the refraction angle defined by $\theta_{i,j+1} = \arcsin \frac{s_{i,j}}{s_{i,j+1}} \sin \theta_{i,j}$ where $\theta_{i,j+1}$ is the angle from the vertical of the raypath i in the $j+1$th medium and $s_{i,j}$ is the slowness or the inverse of the velocity of the ith raypath in the jth medium. We can work with both the reflection seismic data and the refraction seismic data to determine the locations and the velocity of the different mediums.

More specifically, if we have a rough idea of the velocity propagation of the wave in each mediums in the subsurface given by a so called background model or reference model, we can attempt to determine an estimate of the true velocity by solving an in-

version problem. The inversion problem relies on the resoluton of the following system $\Delta t = L\Delta s$ where Δt is the difference between the travel time of the reference model and the traveltime of the true model, Δs is the difference between the slowness of the reference model and the true model and L is the raypath matrix, in which each element $l_{i,j}$ is the path length in the reflector i for the jth ray. The goal of the inversion problem is to find Δs. As we can see on the latter expression, we need to invert the raypath matrix L which is generally ill-conditioned to find out Δs. Ideally the tomographic inversion should adjust the reference model to look like the true or original model given by the seismic data which means minimizing Δs. Reflection tomography has a large number of parameters (Stork and Clayton, 1991). Only a small subset of the many possible models can be adequately considered. Since it is unlikely that any of the attempted models will be satisfactory, they are used as a starting point which is adjusted with an iterative gradient inversion method to fit the true or original seismic data.

Seismic tomography is considered as an ill-conditioned (which means that the matrix is invertible but close to be non-invertible or that the condition number is high) nonlinear inversion problem in which the seismic data are given and the goal is to determine the properties of a medium. We mean by inversion problem that we have physical data and we wish to find out the model parameters (Stork and Clayton, 1992). In seismic tomography, the solutions of the inversion problem is given by the determination of the wave speed through a medium and implicitly the medium density. The inversion problem has been solved iteratively by different methods like backprojection (Dines and Lytle, 1979), finite gradient, Gauss Newton and damped-least square (Sheriff and Geldart, 1995). The main problem with seismic tomography is that the inversion problem leads to errors especially when it deals with the determination of the lateral (horizontal) velocity.

The velocity and the direction of propagation of a wave in a medium is governed by its wavenumber and more generally by its wave vector. We define a wavenumber k_x by the following relation $k_x = \frac{2\pi u}{v_x}$ where u is the frequency and v_x is the phase velocity. We emphasize that the wave vector is a vector and then have a direction. We notice on the latter expression that for a constant phase velocity, the wavenumber increase as the frequency increases so we can make a correspondance between high wavenumbers and high frequencies. In effect it will be equivalent to say that we work in the frequency domain or the wavenumber domain. We talk about 1D and 2D wavenumber domain if respectively one wavenumber is involved or two wavenumbers are involved in the processing of a seismic dataset.

In this project, we will be interested in the wave speed through a medium. We will only focus on the high frequency solutions of the inversion problem corresponding to the high wavenumbers. This type of tomography is called traveltime tomography and is coupled with tomographic velocity analysis which is a much less nonlinear problem than the full waveform tomography. Since the determination of the lateral velocity often leads to errors, it is also necessary to process the solution of the inversion problem to accentuate

the high frequency solutions and then it will be easier to discriminate the high frequencies from the low frequencies. The high frequencies corresponding to the high wavenumbers are characterized by a rapid change on the velocity field. We also remark that reflections are caused by high wavenumber components on the velocity field and low wavenumber component of the velocity field have no inherent relationship with the reflector position. Such relationships can be imposed as constraints if geologic information in an area warrants it. Indeed, the rapid changes on the velocity can be geophysically interpretated as a crossing between two mediums (only if the dataset is noiseless). It is then interesting to find out where the high frequencies are located on a seismic dataset because it will give us the insight of the location of the different mediums or their shape if we work in more than 1D. Moreover if we work in 2D and if we suppose that the mediums are horizontal or pratically horizontal, then we will only be interested in the rapid changes of the velocity field on the depth direction, which will be denoted later in the project by z-direction. Generally, the horizontal variations are much less rapid than the variations on the vertical direction. It is also clear that if we leave the seismic velocity data unchanged on the z-direction such that rapid changes are allowed (corresponding to high frequencies on the z-direction) and if we process the horizontal direction (x-direction), we will get an optimal insight of the location of the different mediums and their 2D shapes (we penalise the filtering on the z-direction and process the x-direction).

In reality, the seismic dataset is not noiseless and the different mediums are not allways horizontal (Figure 2), that is why it is important to work with a seismic dataset where the noise has been considerably reduced, and have a rought idea of the geometry and deposition of the different mediums given by the reference model. If we work with
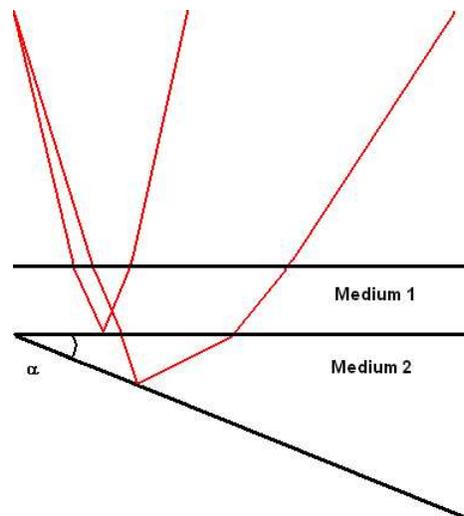

Figure 2: 1D Model with angles of reflection from vertical of the raypathes (represented by red lines) with two reflectors; the first horizontal and the second inclined with a dip-angle α

mediums that are not supposed to be horizontal as shown on Figure 2, for example if some of them are horizontal and others are inclined with a certain angle, it will be necessary to process both directions to determine the exact location or shape of the medium keeping in mind the directionality of the operator which is dependent of the direction of the velocity. In practice the subsurface will be more hetereogenous than the simple sketches in Figure 1 and Figure 2. Nevertheless, geologic information about likely direction of main velocity changes should facilitate the processing of seismic velocity data. In effect, the knowledge about the dip angle can be helpful to adjust the processing of the seismic velocity data as we will see later in this project.

5   Signal processing theory

In this section we are going to introduce some fundamental definitions and properties used in signal processing. In Section 5.1, we will define the discrete Fourier transform and recall some of its most useful properties and in the Section 5.2, we will define and describe some of the main aspects of filter theory.

5.1   Fourier transform and the discrete Fourier transform

We will first present the Fourier transform which can be useful to study the spectrum defined later in this section. Indeed, the Fourier transform permits us to transform a function defined in the spatial domain or time domain to a new equivalent function defined in the frequency domain. For some functions, it is interesting to see what is happening in the frequency domain, especially properties that cannot be seen in the time or spatial domain. In 1D, the Fourier tranform $F(u)$ of a single variable, continuous function, $f(x)$ is defined by the equation

$$F(u) = \int_{-\infty}^{\infty} f(x)e^{-j2\pi ux}dx \tag{1}$$

where $j = \sqrt{-1}$. Conversely, given $F(u)$, we can obtain $f(x)$ by means of the inverse Fourier transform

$$f(x) = \int_{-\infty}^{\infty} F(u)e^{j2\pi ux}du \tag{2}$$

These two equations comprise the Fourier transform pair. They indicate the important fact that a function can be recovered from its transform.

These equations are easily extended to two variables u, v

$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,z)e^{-j2\pi(ux+vz)}dxdz \tag{3}$$

and, for the inverse Fourier transform

$$f(x,z) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(u,v)e^{j2\pi(ux+vz)}dudv \tag{4}$$

The Fourier transform is a continuous transformation. The dicrete Fourier transform (DFT) will be very useful in this project to discretize the Fourier transform in order to implement and compute the frequency content in a spatial image on a regular grid. When we compute the DFT of a function, we assume that we can discretize the function $f(x)$ itself. The discrete Fourier transform of a function of one variable, $f(x)$, $x = 0, 1, 2, ..., M - 1$ is given by the equation

$$F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x) e^{-j2\pi ux/M} \qquad \text{for } u = 0, ..., M - 1 \tag{5}$$

Similarly, given $F(u)$, we can obtain $f(x)$ using the inverse DFT.

$$f(x) = \sum_{u=0}^{M-1} F(u) e^{j2\pi ux/M} \qquad \text{for } u = 0, ..., M - 1 \tag{6}$$

We now define the magnitude or the spectrum of the Fourier transform

$$|F(u)| = \sqrt{\Re(u)^2 + \Im(u)^2} \tag{7}$$

where $\Re(u)$ is the real part of $F(u)$ and $\Im(u)$ the imaginary part. Further

$$\phi(u) = \tan^{-1}\left(\frac{\Im(u)}{\Re(u)}\right) \tag{8}$$

where $\phi(u)$ is called the phase angle or phase spectrum of the transform.

The discrete Fourier transform of a function $f(x, z)$ of size $M \times N$ is given by the equation

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{z=0}^{N-1} f(x, z) e^{-j2\pi(ux/M + vz/N)} \tag{9}$$

and its inverse DFT is defined by

$$f(x, z) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} f(u, v) e^{j2\pi(ux/M + vz/N)} \tag{10}$$

As for a Fourier transform of one variable, we define the spectrum of a Fourier transform of two variables

$$|F(u, v)| = \sqrt{\Re(u, v)^2 + \Im(u, v)^2} \tag{11}$$

Further the phase angle

$$\phi(u, v) = \tan^{-1}\left(\frac{\Im(u, v)}{\Re(u, v)}\right) \tag{12}$$

We will also use another important property of the DFT, the fact that the DFT is separable. This property will help us to gain some running time when we will implement the 2D DFT.

$$F(u,v) = \frac{1}{M}\sum_{x=0}^{M-1} e^{-j2\pi ux/M}\frac{1}{N}\sum_{z=0}^{N-1} f(x,z)e^{-j2\pi vz/N} \tag{13}$$

$$= \frac{1}{M}\sum_{x=0}^{M-1} F(x,v)e^{-j2\pi ux/M} \tag{14}$$

where

$$F(x,v) = \frac{1}{N}\sum_{z=0}^{N-1} f(x,z)e^{-j2\pi vz/N} \tag{15}$$

We see that for each value of x, and for values of $v = 0, 1, 2, ..., N$, this equation is a complete 1D Fourier transform. In other words $F(x,v)$ is the Fourier transform along one row of $f(x,z)$. By varying from 0 to $M-1$, we compute the Fourier transform along all rows of $f(x,z)$ and we notice that the frequency variable u remains constant. To complete the 2D transform we have to vary u from 0 to $M-1$ computing a 1D transform along each column. This result shows that we can compute a 2D transfrom by computing two 1D tranforms. The Fourier transform has also another interesting property, it is periodic. We have

$$F(u,v) = F(u+M,v) = F(u,v+N) = F(u+M,v+N) \tag{16}$$

The inverse transform is also periodic

$$f(x,z) = f(x+M,z) = f(x,z+N) = f(x+M,z+N) \tag{17}$$

This property is useful in a sense that we do not need to plot all the spectrum since we know that the Fourier transform is periodic.

Another property of the spectrum of the Fourier transform will be useful for the 3D plot of the fourier transform in the 2 D wavenumber domain, it is the symmetry of the amplitude spectrum. Indeed we have

$$|F(u,v)| = |F(-u,-v)| \tag{18}$$

5.2   Filter theory

We define a filter by a function, an operator or a mask which is applied on a seismic dataset to enhance or modify the latter. In 1D, we apply a filter $h(m)$ of size m on the signal $f(x)$ by computing the convolution product of a continuous variable, here in the spatial domain or time domain. We denote the result of the convolution product by $g(x)$ in 1D and $g(x,z)$ in 2D.

$$g(x) = (h * f)(x) = \int_{-\infty}^{\infty} h(m)f(x-m)dm \tag{19}$$

Like for the Fourier transform, it is possible to discretize the convolution product for a discrete variable x and m.

$$g(x) = (h * f)(x) = \sum_m h(m)f(x - m) \tag{20}$$

In 2D we get if we denote the filter $h(m, n)$ of size $m \times n$, and the signal by $f(x, z)$, we get the convolution product of a continuous variable

$$g(x, z) = h(x, z) * f(x, z) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(m, n)f(x - m, z - n)dmdn \tag{21}$$

As an image is a function of two (or more) variables it is necessary to define the direction in which the filtering process is taken. For the 2D case we have, the horizontal direction and the vertical direction. We can then express the whole filter by the relation as follow

$$h_\theta(x, z) = \cos\theta h_x(x, z) + \sin\theta h_z(x, z) \tag{22}$$

where

$$\theta = \tan^{-1}\left(\frac{h_z(x, z) * f(x, z)}{h_x(x, z) * f(x, z)}\right) \tag{23}$$

where $h_x(x, z)$ is the mask that filter in the x-direction and $h_z(x, z)$ is the mask that filter in the z-direction and $f(x, z)$ is the input signal. We can see on equation (22) that in the cartesian coordonates, the filter h_θ which filter the direction given by the angle θ can be split in the two components or two projections. We emphasize that if the input signal is a velocity field then it has a direction, so the angle θ will be the direction of the velocity field. In equation (23), we can see that the angle θ is dependent of the processed signal on the z-direction and the processed signal on the x-direction.

For discrete variables x, z, m, n

$$g(x, z) = h(x, z) * f(x, z) = \sum_m \sum_n h(m, n)f(x - m, z - n) \tag{24}$$

We notice that the output $g(x, z)$ of a nonrecursive filter can be seen as a weighted sum (weighted by $h(m, n)$) of all possible values of the input $f(x, z)$. In this project, we are going to scale the operators by using

$$h_{scaled}(x, z) = \frac{h(x, z)}{\sum_m \sum_n h(m, n)} \tag{25}$$

We emphasize that a filter can be seen as a weight matrix of $h(m, n)$. It is important to scale the filter such that the scale between the input signal and the filtered signal is equal. We then scale the operator by dividing by the sum of all its components.

If we work in the frequency domain, it is possible to make correspondence with the

time domain or spatial domain. We emphasize an important result: the convolution theorem (Lu and Antoniou, 1992) that claims that

$$f(x,z) * h(x,z) \leftrightarrow F(u,v)H(u,v) \tag{26}$$

$$f(x,z)h(x,z) \leftrightarrow F(u,v) * H(u,v) \tag{27}$$

This result is very important because we know that it is possible to apply a filter in the frequency domain by making a simple product between the Fourier transform of the signal and the Fourier transform of the filter. Then afterwards, we can apply the inverse DFT of the product result to get the filtered signal in the time or spatial domain as schematized on Figure 3.
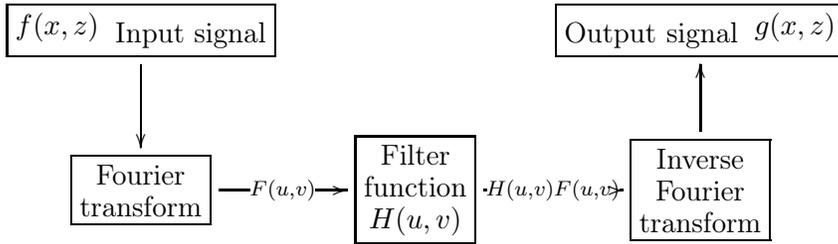

Figure 3: Basic steps for filtering in the frequency domain

In most signal analysis problems, three kind of filters are used, highpass filter, lowpass filter and bandpass filter; We will not use bandpass filters in this project. A lowpass filter is often used to smooth an image or a dataset in order to reduce the noise or just to blur the image by accentuating low frequencies and mark down high frequencies. At the opposite, a highpass filter is used to enhance the details of a picture by accentating high frequencies and mark down low frequencies. We will denote a lowpass filter in the spatial domain by $h_l(x,z)$, $H_l(u,v)$ in the frequency domain and $h(x,z)$, a highpass filter in the spatial domain correponding to $H(x,z)$ in the frequency domain. If we are working in the frequency domain, we can move from a lowpass filter to a highpass filter by using the relation as follows

$$H(u,v) = 1 - H_l(u,v) \tag{28}$$

where $H(u,v)$ is a normalized highpass filter and $H_l(u,v)$ is a normalized lowpass filter. It is also possible to pass from a lowpass filter to a highpass filter in the spatial domain by applying the expression

$$g_h(x,z) = f(x,z) - g_l(x,z) \tag{29}$$

where g_h is the filtered dataset with the highpass filter, $f(x,z)$ is the original dataset and $g_l(x,z)$ is the filtered dataset with the lowpass filter. We notice that the notation defined in equation (22) is not to be confused with the notation defined in equations (28) and (29). We add that the notation in equation (22) can be used both for lowpass and highpass filters.

6   Filters: finite-difference operator, Hamming window

In this Section, we are going to define and study two filters: the finite-difference operator and the Hamming window, both in the 1D/2D wavenumber domain. We are going to compute their Fourier transform and study their spectrum. We notice that for symmetry reasons, we are going to use negative frequencies on the 2D frequency wavenumber domain even if it does not have any physically meaning.

6.1   Finite-difference

We will use a finite-difference operator to determine the variations of the velocity slope. In fact we will see that the finite-difference operator tracks the rapid changes of the velocity field and this property will permit us to make constraints on the manner we wish to process the velocity field later in the project. We will also study its spectrum by computing its Fourier transform and see how the operator behave himself in the frequency domain. We will then introduce the 2D finite-difference in the 2D wavenumber domain and finally, we will study the 1D finite-difference operator in the 2D domain.

6.1.1   1D finite-difference operator in the 1D wavenumber domain

We define the finite-difference by equation (30) and illustrated on Figure 4.

$$h = (1, -2, 1) \tag{30}$$

We get the coefficients of equation (30) by making the approximation: $\frac{\partial^2 f}{\partial x^2} = f(x_{i+1}) - 2f(x_i) + f(x_{i-1})$ where f is any continuous function or the input signal (in our case $f(x)$ represents the velocity field we wish to process.
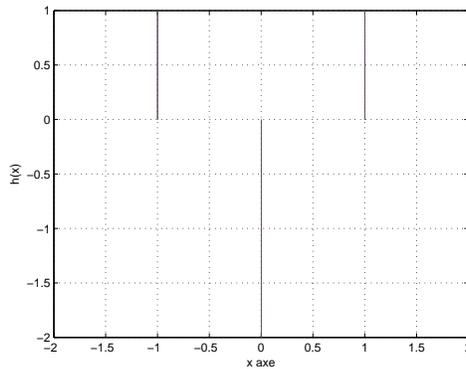

Figure 4: Illustration of the finite-difference operator in the 1D domain with values h=(−1, −2, 1) on the y axis for the respective x=(−1, 0, 1) on the x-axis

The finite-difference operator can be interpreted as an approximation of a second derivative of any continuous function f in a chosen direction. If we apply this operator

to a 1D velocity dataset, then we get all the second derivatives values of this dataset, which is useful to make constraints afterwards on the manner we wish to filter the data. The Fourier transform $H(u)$ of the finite-difference operator h is defined by:

$$H(u) = \int_{-\infty}^{\infty} h(x)e^{-j2\pi ux}dx = \sum_{m=-1}^{1} h(m)e^{-jm2\pi u} \tag{31}$$

where $2\pi u = k_x v_x$. In this case, we can directly compute the Fourier transform $H(u)$. We notice that the finite-difference operator is already a discrete function and then it does not have to be discretize if we wish to compute its DFT.

$$H(u) = 1\exp^{-j2\pi u(-1)} -2\exp^{0} +1\exp^{-j2\pi u(1)} \tag{32}$$
$$= 2(\cos 2\pi u - 1) \tag{33}$$

We notice that the result of equation (33) is a cosine function. We can add that the finite-difference operator have a phase angle equal 0 because $\Im(u) = 0$.
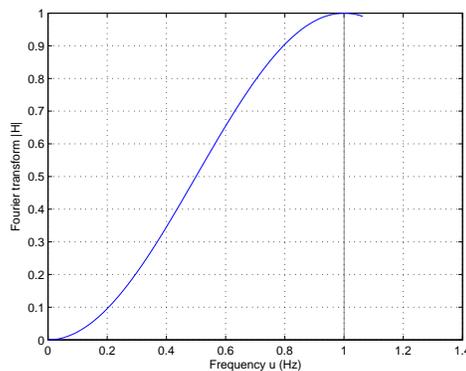

Figure 5: Spectrum $|H|$ of the 1D finite-difference operator against the frequency u in the 1 D wavenumber domain, the black vertical line represent the Nyquist frequency.

We have plotted the modulus of equation (33) on Figure 5 with an angular frequency varying from 0 to π or a frequency varying from 0 Hz to 1 Hz (normalized). In effect the frequency u should vary from 0 Hz to 0.5 Hz if we did not have normalized it. We see on Figure 5 that as the frequency increase, then the norm of the Fourier transform increase. The Fourier transform modulus of the finite-difference operator has been normalized and then will vary from 0 to 1 instead of 0 to 4. The amplitude is 1 for the frequency $u_{nyq} = 1$ Hz. The Nyquist frequency and is represented on Figure 5 by a black vertical line and defined as follow: $u_{nyq} = \frac{u_{sample}}{2}$.

Moreover we know that the frequency is a function of the wavenumber ($k_x = \frac{2\pi u}{v_x}$), then the Figure 5 shows the finite-difference operator will accentuate the high wavenumber (respectively high frequencies) and will mark down the low wavenumbers (respectivly low

frequencies). We can conclude that the finite-difference operator is a highpass filter. It is possible to apply this filter on a seismic velocity dataset by using a convolution product between the operator and the signal we would like to filter (in our case the velocity field).

6.1.2   Discretize Laplacian or 2D finite-difference operator in the 2D wavenumber domain

In the 2D time domain or spatial domain, we can write the finite-difference function $h(x, z)$ as the equation (34) and the operator is illustrated on Figure 6.

$$h = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} \tag{34}$$

The 2D finite-difference operator can be seen as the sum of the two second derivatives.

$$\bigtriangledown^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial z^2} \tag{35}$$

where f is the input signal or the velocity field we wish to process.

We are first going to define a grid $G = \{(x, z) \mid a < x < b, \ c < z < d\}$, then we wish to define steps $h = (b - a)/n$ and $k = (d - c)/m$. We then discretize the variable x such that $x_i = a + ih$ for each $i = 0, 1, ..., n$ and $z_j = c + jk$ for each $j = 0, 1, ..., m$. Then we wish to know at each mesh point the value of $\frac{\partial^2 f(x_i, z_j)}{\partial x^2}$ and $\frac{\partial^2 f(x_i, z_j)}{\partial z^2}$. We are then going to use a finite-difference approximation to estimate those second derivatives (Burden and Faires, 2001).

$$\frac{\partial^2 f(x_i, z_j)}{\partial x^2} = \frac{f(x_{i+1}, z_j) - 2f(x_i, z_j) + f(x_{i-1}, z_j)}{h^2} \tag{36}$$

$$\frac{\partial^2 f(x_i, z_j)}{\partial z^2} = \frac{f(x_i, z_{j+1}) - 2f(x_i, z_j) + f(x_i, z_{j-1})}{k^2} \tag{37}$$

At least, we see that the sum of equation (36) and (37) give the coefficients defined in equation (34) if and only if $h = k$.

We emphasize that the 2D finite-difference in the 2D wavenumber domain is rotation invariant for angle multiples of $\frac{\pi}{2}$. We say that the operator is isotropic. We also recall that the Laplacian operator is a linear operator but the 2D finite-difference in the 2D wavenumber domain is not linear.
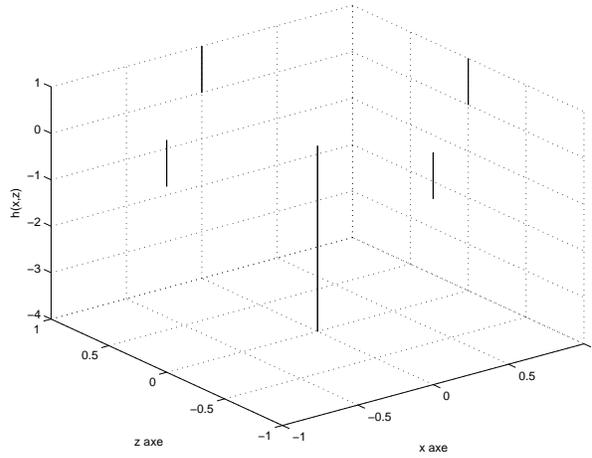
Figure 6: Illustration of the 2D finite-difference operator $h(x, z)$, $h = -4$ at $(x = 0,$ $z = 0)$, $h = 1$ for $(x = 0, z = -1)$, $(x = -1, z = 0)$, $(x = 1, z = 0)$,$(x = 0, z = 1)$

In this case, it is also possible to compute the Fourier transform $H(u, v)$ analytically in the 2D wavenumber domain.

$$H(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x, z) e^{-j2\pi xu - j2\pi yv} dx dz \tag{38}$$

$$= \sum_{m=-1}^{1} \sum_{n=-1}^{1} h(m, n) e^{-j(2\pi mu + 2\pi nv)} \tag{39}$$

$$= e^{2\pi jv} + e^{-2\pi jv} + e^{2\pi ju} + e^{-2\pi ju} - 4 \tag{40}$$

$$= 2\cos 2\pi u + 2\cos 2\pi v - 4 \tag{41}$$

We see that the result of equation (41) is also a cosine function. We notice that as the 1D finite-difference operator, the 2D finite-difference operator does not have any phase because $\Im(u, v) = 0$.

We used the symmetry property of the Fourier transform to plot Figure 7. We have $H(1, 1) = H(-1, -1)$ and $H(1, -1) = H(-1, 1)$, this type of symmetry is said to be quadrantal. We see on Figure 7 that the Fourier transform of the 2D finite-difference operator has a similar shape as a paraboloid with a maximum amplitude at 1 for the respective frequencies $(u, v) = (1, 1)\ Hz$, $(u, v) = (1, 1)\ Hz$. We also see that the dome point to the origin. We see on Figure 7 that the 2D finite-difference operator or filter will accentuate high frequencies (respectively high wavenumbers) and mark down low frequencies (respectively low wavenumbers) in the 2D frequency domain; it is also a highpass filter.

The Nyquist frequencies u_{nyq} in Figure 7 are located at $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$,$(u, v) = (1, -1)\ Hz$. We can also see that the spectrum is rotation invariant for angle multiples of $\frac{\pi}{2}$ with an axis of rotation perpendicular
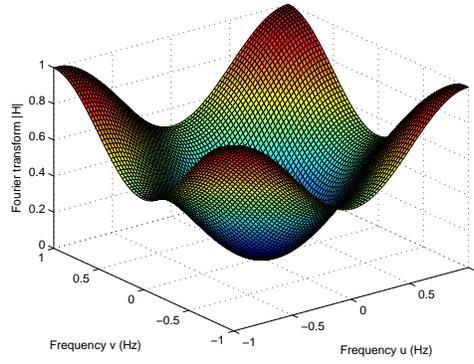
Figure 7: Modulus of the Fourier transform $|H|$ of the highpass 2D finite-difference operator against u (frequency x axis), v (frequency y axis) in the 2 D wavenumber domain with the Nyquist frequencies located at $(u, v) = (1, 1)$ Hz, $(u, v) = (-1, 1)$ Hz, $(u, v) = (-1, -1)$ Hz, $(u, v) = (1, -1)$ Hz

to the u, v plan. This means that the operator will filter both the x and z direction of a seismic velocity dataset. In this example too, the Fourier transform modulus has been normalized and will vary between 0 and 1 instead of 0 and 8.
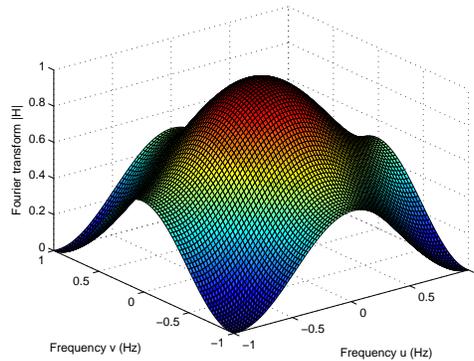

Figure 8: Modulus of the Fourier transform $|H_l|$ of the lowpass 2D finite-difference operator against u (frequency x axis), v (frequency y axis) in the 2 D wavenumber domain with the Nyquist frequencies located at $(u, v) = (1, 1)$ Hz, $(u, v) = (-1, 1)$ Hz, $(u, v) = (-1, -1)$ Hz, $(u, v) = (1, -1)$ Hz

On Figure 8, we have plotted the spectrum of the lowpass 2D finite-difference operator which is found by using equation (28). Like Figure 8, we see that the shape of the amplitude spectrum is the same and the Nyquist frequencies are also unchanged. The only difference remains that the dome do not point to the origin but it points to the point $(u, v, |H_l(u, v)|) = (0, 0, 1)$.

6.1.3   New filter based on 1D finite-difference

We saw in Section 6.1.2 that the 2D finite-difference in the 2D wavenumber domain was rotation invariant and filter the image in both directions. It is possible to modify the filter such that it will only filter the x-direction and leaving the z-direction unchanged. In this section we will show that it is possible to use the 1D expression of the finite-difference (see Section 6.1.1) to filter a 2D seismic velocity dataset by making small changes in the expression of the latter operator and expanding it as a 2D filter. We notice that the operator we will study is nonlinear and nonrecursive and we will not for instance take care of the direction of the velocity.

For the finite-difference, it is equivalent to use only the second derivative defined in equation (36) on the x-direction without making any changes on the z-direction. We wish to use the 1D finite-difference and making of it a 2D filter. Indeed we could have used a 1D filter (3×1) on a 2D dataset but the rotation process is easier with a square mask than for a rectangular mask and the result is the same. We can define the filter as

$$\frac{\partial^2 f}{\partial x^2}(x_i, z_j) = [f(x_{i+1}, z_j) - 2f(x_i, z_j) + f(x_{i-1}, z_j)] = [h(x, z) * f(x, z)] \qquad (42)$$

If we express it in matrix form, we get:

$$h(x, z) = \begin{pmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{pmatrix} \qquad (43)$$

If we compare equation (34) to equation (43), we have the following relationship:

$$h(x, z) = \begin{pmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{pmatrix} \qquad (44)$$

We remark that the matrix in equation (44) can be written as the sum of two matrices if and only if the step of the grid is equal on the x-axis and the z-axis. We see that the matrix has values only on the x-direction and the others values are set to zero. It differs from equation (34) by the two 1 on the z-direction wich have been set to zero and the -4 set to -2.

If we wanted to filter only the vertical direction of the seismic velocity dataset, we would have used the following matrix

$$h(x,z) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{pmatrix} \qquad (45)$$

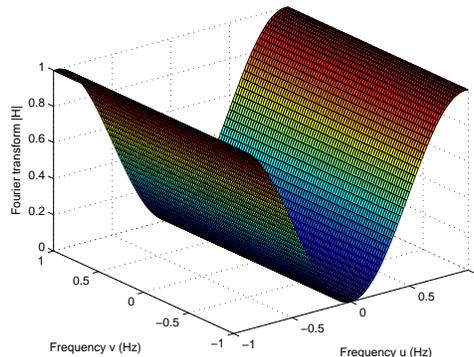We can see on Figure 9 the spectrum of the modified finite-difference operator with

Figure 9: Spectrum of the modified operator based on 1D finite-difference with the Nyquist frequencies located at $(u,v) = (1,1)$ Hz, $(u,v) = (-1,1)$ Hz, $(u,v) = (-1,-1)$ Hz,$(u,v) = (1,-1)$ Hz

its Nyquist frequencies located at $(u,v) = (1,1)$ Hz, $(u,v) = (-1,1)$ Hz, $(u,v) = (-1,-1)$ Hz,$(u,v) = (1,-1)$ Hz. The spectrum is quadrantal because $H(1,1) = H(-1,1) = H(1,-1) = H(-1,-1)$. We notice like for the previous sections, that the Fourier transform $|H|$ and the frequencies has been normalized. We see that the shape of the Fourier transform $|H(u,v)|$ is different from the shape obtained on Figure 7. We see that the u frequency corresponding to the x-direction in the time domain vary as the v frequency corresponding to the z-direction in the time domain remains constant.

We still do not want to filter the z-direction and we want to use the 1D finite-difference as a filter dependent of the direction of the velocity (see equation (22)). In fact we are only going to multiply the convolution product between the filter and the signal by $\cos(\theta)$ when we are filtering on the x-direction corresponding to the projection of the x-component of the velocity on the x axis.

If we look at the spectrum of the operator with $\theta = \frac{\pi}{3}$ on Figure 10, we see that the minimum is located at 0.5 and not 0 compared to Figure 9 where $\theta = 0$. Figure 10 shows that the magnitude of the Fourier transform $|H_\theta(u,v)|$ vary as θ vary. In other words, in the spatial or time domain, the operator $h_\theta(x,z)$ is dependent of the direction of the velocity.
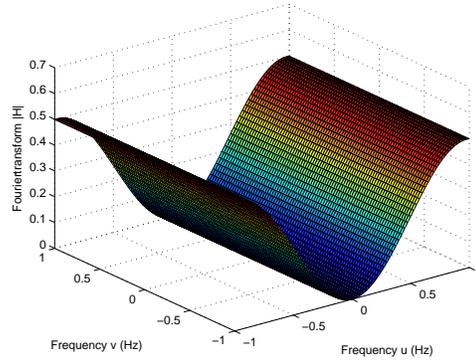
Figure 10: Spectrum of the operator based on the 1D finite-difference with respect to the direction of the velocity, with $\theta = \frac{\pi}{3}$ with Nyquist frequencies located at $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$, $(u, v) = (1, -1)\ Hz$

6.2   The Hamming window

Now we wish to study the DFT in the 1D/2D wavenumber domain on another operator: the Hamming window. In this section we want to use the Hamming window operator instead of the finite-difference operator (see Section 6.1) because of its specific properties. The Hamming window function is interesting because we can decide of the steepness of its slope if we make vary N. This property gives one more degree of freedom in comparison with the finite-difference operator. This property will help us to parameterize the penalty function in order to make constraints on the velocity field. We notice also that the Hamming window is a nonlinear operator because of the cosine function. For simplicity and notation reasons, we did not scale the Hamming window operators.

6.2.1   1 D Wavenumber domain in the 1D wavenumber domain

We define the Hamming window in the 1D time domain by

$$h_l(x) = 0.54 - 0.46 \cos(2\pi \frac{x}{N}) \tag{46}$$

where $-\frac{N}{2} < x < \frac{N}{2}$ We see on Figure 11 that the Hamming window is periodic and symmetric. In our case the period is $x = 2$ and the symmetry axis is located at $x = 0$. We wish to compute the Fourier transform of the Hamming window operator. In this
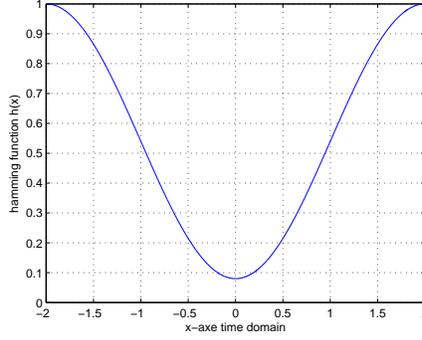
Figure 11: plot of the 1D Hamming window function with $-2 < x < 2$ (time or spatial domain), $N = 4$

case it is possible to compute it directly. We have $-2 < x < 2$.

$$H_l(u) = \int_{-\infty}^{\infty} h(x)e^{-j2\pi xu}dx \tag{47}$$

$$= \int_{-2}^{2} [0.54 - 0.46\cos(\frac{2\pi x}{N})]e^{-j2\pi xu}dx \tag{48}$$

$$= 0.54[\frac{\sin(2\pi xu)}{2\pi u}]_{-2}^{2} + j0.54[\frac{\cos(2\pi ux)}{2\pi u}]_{-2}^{2} - 0.46\int_{-2}^{2}\cos(\frac{2\pi x}{N})\cos(2\pi xu))dx \tag{49}$$

$$- 0.46j\int_{-2}^{2}\cos(\frac{2\pi x}{N})\sin(2\pi xu))dx \tag{50}$$

$$= 0.54[\frac{\sin(2\pi xu)}{2\pi u}]_{-2}^{2} + j0.54[\frac{\cos(2\pi xu)}{2\pi u}]_{-2}^{2} - 0.23[\sin(\frac{2\pi}{N} + 2\pi u)x)]_{-2}^{2}(\frac{2\pi}{N} + 2\pi u))^{-1} \tag{51}$$

$$- 0.23[\sin(\frac{2\pi}{N} - 2\pi u)x)]_{-2}^{2}(\frac{2\pi}{N} - 2\pi u))^{-1} + 0.23j[\cos(2\pi ux + \frac{2\pi x}{N})]_{-2}^{2}(2\pi u + \frac{2\pi}{N})^{-1} \tag{52}$$

$$+ 0.23j[\cos(2\pi ux - \frac{2\pi x}{N})]_{-2}^{2}(2\pi u - \frac{2\pi}{N})^{-1} \tag{53}$$

$$\tag{54}$$

We can also approximate the Fourier transform of the Hamming window with a DFT defined on equation (46) and get the spectrum of the Hamming window as shown on Figure 11. Because the Hamming window is defined as a continuous function, if we wish to compute the DFT of the Hamming window, we also have to discretize the latter. We see on Figure 12 that as the frequency increase, the Fourier transform of the Hamming window $|H_l|$ decrease. We conclude that the Hamming window is a lowpass filter. We also see that the Nyquist frequency is $u_{nyq} = 1 \ Hz$ (normalized frequency). We recall that the modulus of the Fourier transform has also been normalized in this example. Besides
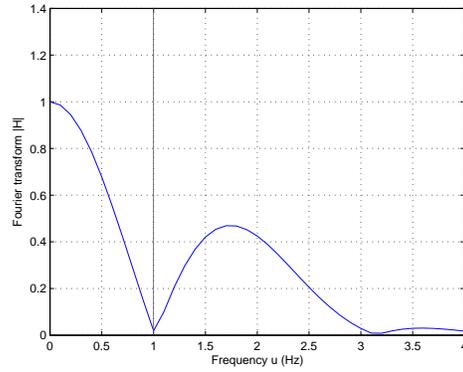
Figure 12: Plot of the lowpass Hamming window spectrum with the Nyquist frequency located at $u = 1\ Hz$

we normalized the modulus $|H_l|$ by dividing it by its highest value: $|H_l|_{norm} = \frac{|H_l|}{\max(|H_l|)}$. In this project, we wish to make use of the Hamming window as a high pass filter. We are then going to use the transfer function of the highpass filter (see equation (28)). We normalize the modulus of the Fourier transform of the Hamming window in order to operate this transformation.
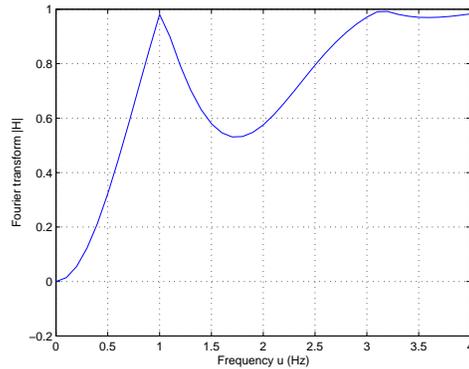

Figure 13: Plot of the spectrum of the highpass Hamming window with the Nyquist frequency located at $u = 1\ Hz$

On Figure 13, we can see that the modulus of the Fourier transform increase as the frequency increase and is bounded by the Nyquist frequency which is located at $u = 1\ Hz$. If we now take the inverse Fourier transform of the spectrum, we will get a highpass 1D Hamming window function in the time or spatial domain. We notice that Figure 13 is like Figure 12 but inverted.

6.2.2   2 D Hamming window in the 2D Wavenumber domain

In this section we are going to deal with the 2D wavenumber domain. We are then going to use a 2D Hamming window. Like the previous section we wish to compute its Fourier transform to see what is happening in the frequency domain, and then study its spectrum. The Hamming window in the 2D domain is defined as follow and illustrated on Figure 14.

$$h_l(x, z) = \begin{cases} [0.54 + 0.46\cos(\pi\frac{x}{N})][0.54 + 0.46\cos(\pi\frac{z}{M})] & \text{if } |x| < \frac{N}{2} \ |z| < \frac{M}{2} \\ 0 & \text{otherwise} \end{cases} \tag{55}$$
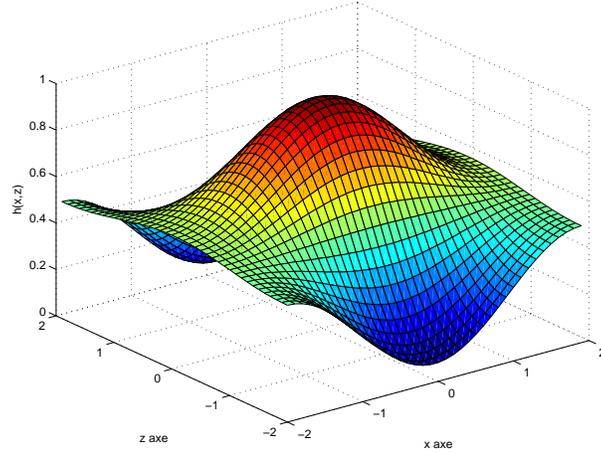

Figure 14: Illustration of the 2D lowpass Hamming window with $-2 < x < 2$ and $-2 < z < 2$, $N = 4$, $M = 4$

In this section we are not going to compute the Fourier transform of equation (55) analytically. Instead we are going to directly compute its DFT in the 2D domain as defined in equation (56).

$$DFT(u, v) = \sum_{m=-2}^{2} \sum_{n=-2}^{2} h_l(m, n)e^{-j(2\pi um + 2\pi vn)} \tag{56}$$
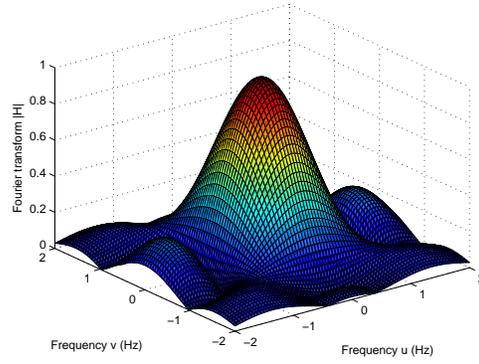
Figure 15: Illustration of the spectrum of the 2D lowpass Hamming window function with Nyquist frequencies located at $(u, v) = (1, 1)$ Hz, $(u, v) = (-1, 1)$ Hz, $(u, v) = (-1, -1)$ Hz, $(u, v) = (1, -1)$ Hz

The modulus of the Fourier transform of the 2D Hamming window is represented on Figure 15. We see that as the frequencies increase in the 2D wavenumber domain, then the modulus of the Fourier transform decrease. The 2D hamming window is also a lowpass filter. Its dome or max amplitude is located at frequencies $(u, v) = (0, 0)$ Hz. We recall that the modulus of the Fourier transform has also been normalized in this example in order to get the transfer function of a highpass 2D Hamming window from the transfer function of a lowpass 2D hamming window. We also notice that the amplitude response of the filter is quadrantal with $H_l(1, 1) = H_l(-1, 1) = H_l(1, -1) = H_l(-1, -1)$. Its Nyquist frequencies are located at $(u, v) = (1, 1)$ Hz, $(u, v) = (-1, 1)$ Hz, $(u, v) = (-1, -1)$ Hz, $(u, v) = (1, -1)$ Hz.
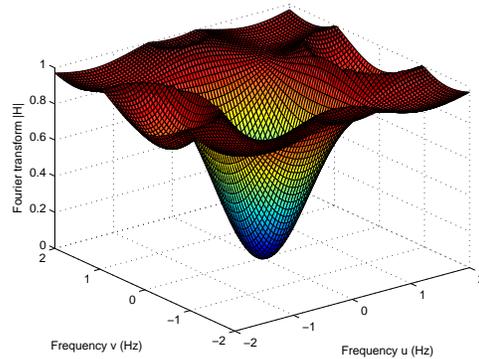

Figure 16: Illustration of the spectrum of the highpass 2D Hamming window function with Nyquist frequencies located at $(u, v) = (1, 1)$ Hz, $(u, v) = (-1, 1)$ Hz, $(u, v) = (-1, -1)$ Hz, $(u, v) = (1, -1)$ Hz

The spectrum of the 2D highpass Hamming window operator is shown on Figure

16. We can see that the Nyquist frequencies are located at $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$, $(u, v) = (1, -1)\ Hz$. The spectrum is centered at the origin but inverted compared to Figure 15. We emphasize that the spectrum is rotation invariant so the 2D Hamming window in the 2D wavenumber domain is an isotropic filter.

6.2.3   New filter based on the 1D Hamming window

As for the finite-difference, it is possible to define an operator based on the Hamming window which would only filter the x-direction of a seismic dataset and leaving the z-direction unchanged. We will have for a 3×3 mask with $x = (-1.5, 0, 1.5)$.

$$h_l(x, z) = \begin{pmatrix} 0 & 0 & 0 \\ 0.54 - 0.46\cos(\frac{2\pi}{N}(-1.5)) & 0.54 - 0.46\cos(\frac{2\pi}{N}(0)) & 0.54 - 0.46\cos(\frac{2\pi}{N}(1.5)) \\ 0 & 0 & 0 \end{pmatrix} \tag{57}$$

We notice as for Section 5.1.3 that all the elements on the z-direction have been set to zero. We remark that to filter only the vertical direction we would have used the following matrix

$$h_l(x, z) = \begin{pmatrix} 0 & 0.54 - 0.46\cos(\frac{2\pi}{N}(-1.5)) & 0 \\ 0 & 0.54 - 0.46\cos(\frac{2\pi}{N}(0)) & 0 \\ 0 & 0.54 - 0.46\cos(\frac{2\pi}{N}(1.5)) & 0 \end{pmatrix} \tag{58}$$
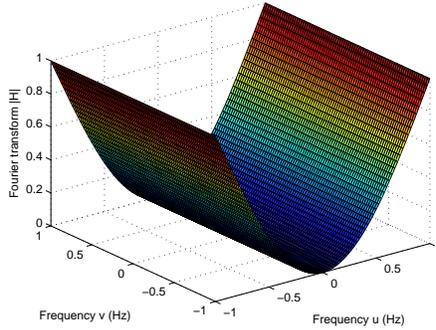

Figure 17: Spectrum of the modified operator based on the highpass Hamming window with the Nyquist frequencies $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$, $(u, v) = (1, -1)\ Hz$

We can see on Figure 17 that the amplitude spectrum of the modified operator based on the 1D Hamming window is very similar in shape compared to Figure 9. Its Nyquist frequencies are located at $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$, $(u, v) = (1, -1)\ Hz$. The spectrum is quadrantal. We see that the u frequency corresponding to the x-direction in the time domain vary as the v frequency corresponding

to the z-direction in the time domain remains constant. In fact we are just going to multiply the convolution product between the filter and the signal by $\cos\theta$ when we are filtering on the x-direction corresponding to the projection of the x-component of the velocity on the x-axis (see equation (22)).

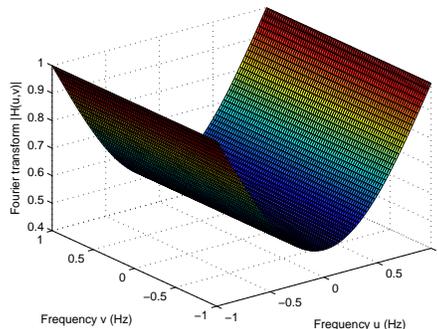
Figure 18: Spectrum of the modified operator based on the highpass Hamming window with the Nyquist frequencies $(u, v) = (1, 1)$ Hz, $(u, v) = (-1, 1)$ Hz, $(u, v) = (-1, -1)$ Hz, $(u, v) = (1, -1)$ Hz with respect to the direction of the velocity field (here with $\theta = \frac{\pi}{3}$)

On Figure 18, we can see that the magnitude of the Fourier transform vary as θ vary by comparing Figure 17 with $\theta = 0$ and Figure 18 with $\theta = \frac{\pi}{3}$. In other words, in the spatial or time domain, the operator $h_\theta(x, z)$ is dependent of the direction of the velocity.

7   Using rotations on the 1D operators

In this section we want to use rotations (not to be confused with the directionality of the velocity) on the 1D operator in the 2D domain studied previously in Section 6.1.3 and 6.2.3. Indeed we will operate rotations on filters by using linear interpolations. We are using linear interpolations to approximate the rotation process because it is impossible to strictly rotate the operator because the latter describe a square while the rotation process describe a circle. In Section 6.1.3 and 6.2.3, we saw that it was possible to make a 1D operator dependent of the direction of the velocity and get its projection on achosen direction. In this section, we want to rotate the whole operator along the direction of the local velocity. We will see in this section that a 1D filter can be generalized as a 2D filter by applying a rotation on the latter. We will then study and compare the spectrum of the differents filters obtained after the interpolation process.

7.1   Rotation on the operator based on the 1D finite-difference operator

We saw in Section 6.1.3, that the 1D finite-difference operator in the 2D wavenumber domain along the x-direction was defined by equation (43) and was filtering just the

x-direction and left unchanged the z direction. It is possible to rotate this operator and generalize its expression by using linear interpolations. In this section we want to show that instead of using a 2D Hamming window in the 2D wavenumber domain, we can use a 1D Hamming window and apply rotation on it to parameterize the filtering process. We will show that it is possible to use a 1D highpass filter in the 2D domain to mark down the low wavenumbers and accentuate high wavenumbers of a seismic dataset. This operator will filter the seismic dataset as a 2D filter but the main difference remains that this kind of filtering is anisotropic and we still in the 1D wavenumber domain. We will study the matrices resulting from the rotations and their spectrums.

In fact if we rotate the 1D finite-difference operator with an angle of $\frac{\pi}{4}$ anticlockwise, we get

$$h = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{59}$$

We see on equation (59) that the matrix is a diagonal matrix. The value of the coefficients are unchanged compared to equation (43) except that they occupy the diagonal cells instead of the horizontal cells. If applied to a seismic dataset we could use this filter if the angle corresponding to the direction of the velocity $\theta = \frac{\pi}{4}$.
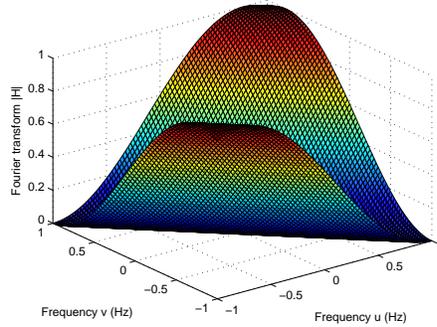

Figure 19: Plot of the spectrum of the operator based on 1D finite-difference after a rotation of $\frac{\pi}{4}$ with the Nyquist frequencies located at $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$, $(u, v) = (1, -1)\ Hz$

We see on Figure 19 that the spectrum of the operator based on finite-difference after a rotation of $\frac{\pi}{4}$ will both filter the x and z directions, since the spectrum vary both on the u and v frequencies. Its Nyquist frequencies are located at $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$, $(u, v) = (1, -1)\ Hz$. We can add that the spectrum on Figure 19 is the same as the spectrum on Figure 9 but spined with an angle of $\frac{\pi}{4}$. We see that the spectrum on Figure 19 is oriented diagonnaly and is centered on the diagonal. We notice that the symmetry is not quadrantal.

For an angle of $\frac{\pi}{8}$, we interpolate and get the following filter matrix

$$h = \begin{pmatrix} 0.5 & 0 & 0 \\ 0.5 & -2 & 0.5 \\ 0 & 0 & 0.5 \end{pmatrix} \tag{60}$$

We see on equation (60) that the interpolation process can be seen as if we weighted the x-direction and the diagonal direction by 0.5. Its spectrum is illustrated on Figure 20. We can see that the spectrum $|H|$ from Figure 20 is diagonal too but not centered on the diagonal because $h_{2,1} = 0.5$ and $h_{2,3} = 0.5$ (compare equation (60) with equation (61)). Its Nyquist frequencies are located at $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$, $(u, v) = (1, -1)\ Hz$. Here also the symmetry remains not quadrantal.
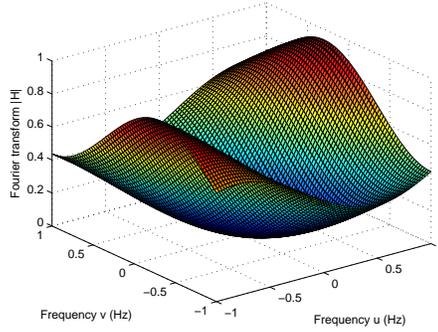

Figure 20: Plot of the spectrum of the operator based on the 1D finite differences after a rotation of $\frac{\pi}{8}$ with the Nyquist frequencies located at $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$, $(u, v) = (1, -1)\ Hz$

For a rotation of $\frac{\pi}{16}$, we will have the following matrix:

$$h = \begin{pmatrix} 0.25 & 0 & 0 \\ 0.75 & -2 & 0.75 \\ 0 & 0 & 0.25 \end{pmatrix} \tag{61}$$

Indeed, we can iterate this rotation process for every angles θ found in the seismic velocity dataset, and cumpute the matrix of the filter that will fit the best with the direction of the angle θ. For example if the angle $\theta = \frac{\pi}{4}$ then we will apply equation (59) to the seismic velocity dataset, if the angle $\theta = \frac{\pi}{8}$, we will apply equation (60), if the angle $\theta = \frac{\pi}{16}$, we will apply equation (61) and so on. In fact we are going to use the direction of the velocity or the angle θ then choose the most appropriate matrix to filter the dataset in that direction. We get the following algorithm

For x=0 to M

    For z=0 to N

        • Compute θ

        • Choose appropriate matrix (filter) corresponding to the value of θ

        • Apply the filter on the dataset

    End

End

This algorithm is important in order to impose constraints on the manner we wish to filter the dataset because, unlike the other type of filter, we have the possibility to choose which matrix should be the most appropriate to the direction of the velocity. We can then decide or not to filter the chosen direction, and we can decide which of the filters would be the most appropriate for the filtering if we wish to filter that chosen direction. Sometimes the angle θ is known beforehand from the geology and then it is not necessary to compute it. Other times, the angle must be estimated by using the seismic velocity data.

7.2   Rotation on the operator based on the 1D Hamming window operator

As for Section 7.1, we will use the 1D Hamming window expression in the 2D wavenumber (equation (57)) domain and approximate rotation on the latter by using linear interpolations. For an angle of $\frac{\pi}{4}$ anticlockwise, we have the following filter matrix

$$h_l = \begin{pmatrix} h_l(-1.5) & 0 & 0 \\ 0 & h_l(0) & 0 \\ 0 & 0 & h_l(1.5) \end{pmatrix} \tag{62}$$

We can see that the matrix if applied to seismic velocity dataset will diagonaly filter the latter. Equation (62) correspond to equation (59) for the finite differences. We see on Figure 21 that the spectrum of the operator based on the Hamming window after a rotation of $\frac{\pi}{4}$ will both filter the x and z-directions, since the spectrum vary both on the u and v frequencies. Its Nyquist frequencies are located at $(u, v) = (1, 1)\ Hz$, $(u, v) = (-1, 1)\ Hz$, $(u, v) = (-1, -1)\ Hz$, $(u, v) = (1, -1)\ Hz$. We emphasize that there is similarities on the geometry of the shape between Figure 19 and Figure 21. If
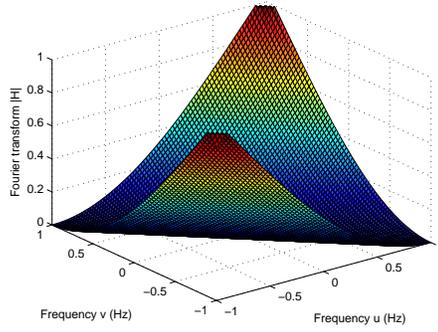
Figure 21: Plot of the spectrum of the operator based on the 1D highpass Hamming window after a rotation of $\frac{\pi}{4}$ with the Nyquist frequencies loacted at $(u,v) = (1,1)\ Hz$, $(u,v) = (-1,1)\ Hz$, $(u,v) = (-1,-1)\ Hz$, $(u,v) = (1,-1)\ Hz$

we apply a rotation angle $\theta = \frac{\pi}{8}$, we get the following matrix

$$h_l = \begin{pmatrix} \frac{h_l(-1.5)}{2} & 0 & 0 \\ \frac{h_l(-1.5)}{2} & h_l(0) & \frac{h_l(1.5)}{2} \\ 0 & 0 & \frac{h_l(1.5)}{2} \end{pmatrix} \tag{63}$$

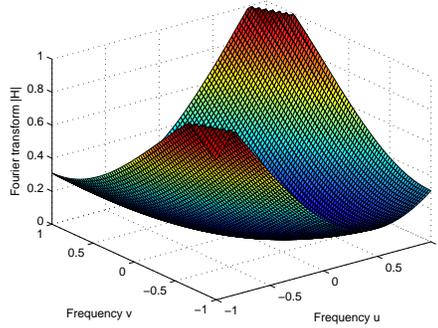Its spectrum is illustrated on Figure 22.   We can see on Figure 22 that the mag-

Figure 22: Plot of the spectrum of the operator based on the 1D highpass Hamming window after a rotation of $\frac{\pi}{8}$ with the Nyquist frequencies located at $(u,v) = (1,1)\ Hz$, $(u,v) = (-1,1)\ Hz$, $(u,v) = (-1,-1)\ Hz$, $(u,v) = (1,-1)\ Hz$

nitude of the diagonal elements correpsonding to the Nyquist frequencies (located at $(u,v) = (1,1)\ Hz$, $(u,v) = (-1,1)\ Hz$, $(u,v) = (-1,-1)\ Hz$, $(u,v) = (1,-1)\ Hz$) decrease compared to Figure 21, because of the weighted matrix expressed in equation (63). Indeed, there is similarities in the geometry of the spectrum between Figure 20 and Figure 22 due to the linear interpolation process which remains the same for both filters.

For a rotation of $\frac{\pi}{16}$, we will have the following matrix:

$$h_l = \begin{pmatrix} \frac{h_l(-1.5)}{4} & 0 & 0 \\ \frac{3h_l(-1.5)}{4} & h_l(0) & \frac{3h_l(1.5)}{4} \\ 0 & 0 & \frac{h_l(1.5)}{4} \end{pmatrix} \tag{64}$$

As for Section 7.1, we can iterate the rotation process for each angle found in the seismic velocity dataset and compute the matrix of the filter that will fit the best with the direction of the angle θ. We get the same algorithm, only the operator remains different.

8   Using different operators on a 2D synthetic seismic dataset

In this section we are going to use the different operator studied in the previous sections on a 50×50 synthetic seismic velocity field dataset by using the convolution product. We notice that the synthetic seismic velocity dataset is subject to a dip angle of $11.3°$ from the point (0,7) to the point (50,17). We are first going to apply both of the operators studied in Section 6.1.2 and 6.2.2 to the synthetic velocity seismic data, then will use the operators described in Section 7.

We remark that this type of filtering is nonrecursive because the output of the process is only dependent of the input function $f(x, z)$. We notice that for all the Figures below, the edges of the images have not been filtered. We denote the x-axis or x-direction the horizontal axis of the seismic dataset and the z-axis or z-direction, the vertical axis of the seismic dataset
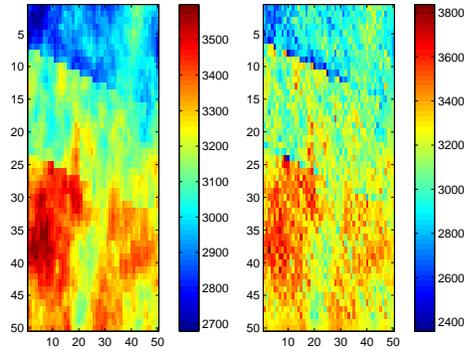

Figure 23: On the left: A synthetic seismic velocity field (50×50), on the right: 2D finite-difference operator (highpass) applied on the synthetic velocity field with a 3×3 mask

We can see on Figure 23 when we apply the Laplacian operator which is isotropic that the result here filter both the x and z-directions. We can see that the image looks clearer, it is easier to determine where the rapid variations of the velocity field are located

compared to the original seismic dataset. Indeed, this kind of filter is often used to detect discontinuities on the velocity fields and to accentuate the details on an image. In our case, since the filter is a highpass filter, it will mark down the low wavenumbers and accentuate the high wavenumbers corresponding respectively to the low frequencies and the high frequencies. On the Figure 23, we can see that the detail in the image is clearer and sharper than the original image. Nevertheless, the trend of the dip angle $11.3°$ from the point $(0, 7)$ to the point $(50, 17)$ does not seen clear. We notice that the problem with highpass filter is that if the seismic dataset contains noise, the finite-difference operator will also accentuate the noise on the image. To avoid this problem, we can use a smoothing algorithm and then apply the highpass filter to the seismic dataset.
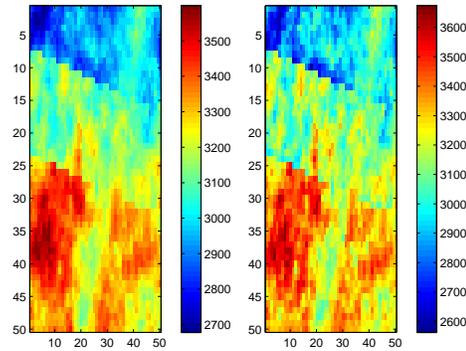

Figure 24: On the left: synthetic seismic velocity field (50×50), on the right: 2D highpass Hamming window operator applied on the synthetic velocity field with a 3×3 mask

We can see on Figure 24 that we get a quite good result, the image also looks clearer and sharper than the original seismic dataset and it is easy to see precisely where the rapid variations of the seismic velocity field are located. We can also use a larger mask, for example a 5×5 mask as shown on Figure 25. We have seen on Figure 23 and Figure 24 the result by applying a 2D filter in the 2D wavenumber domain and that will filter both x and z-direction. We can see that Figure 23 share many similarities with Figure 24, the rapid variations of the velocity field lies at the same locations. The main differences between Figure 23 and Figure 24 relies on the filtering process itself but the two filtering methods leads to the same goal: detect rapid variations on the velocity field, mark down low frequencies and accentuate high frequencies. The main problem with those kind of filters is that it is very difficult to impose constraints by incorporate directional penalty functions because the latters are directionaly independent and then will filter the seismic dataset without taking care of the direction of the velocity.
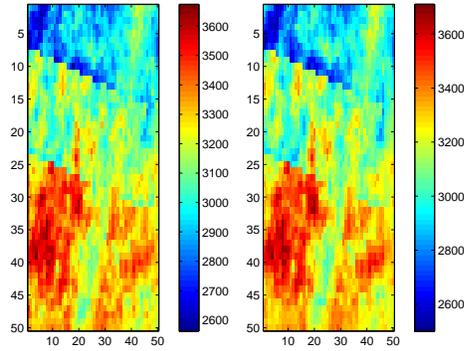
Figure 25: On the left: 2D highpass Hamming window operator applied on the synthetic seismic velocity field with a 3×3 mask, on the right: 2D highpass Hamming window operator applied on the synthetic velocity field with a 5×5 mask

We now want to plot the 1D finite-difference operator in the 2D wavenumber domain as discribed in Section 6.1.3. If we apply this filter on a seismic velocity dataset, it will filter the x-direction and would leave unchanged the z-direction as shown on Figure 26 (without taking care of the velocity direction for the moment). This filter will then penalty the filtering on the z-direction. We notice that theoretically, it is impossible to only filter the x-direction but the result we get on Figure 26 is very close to it.
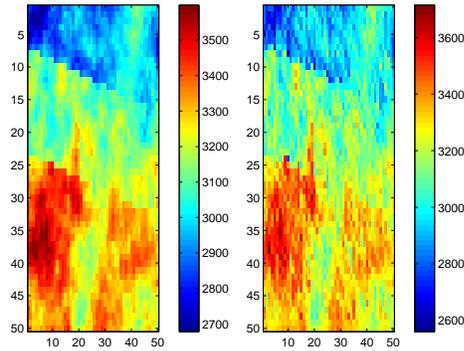

Figure 26: On the left: synthetic dataset, on the right, modified operator based on the 1D finite-difference applied to a dataset with a 3×3 mask. x-direction filtered, z-direction unchanged

We can see on Figure 26, that the x-direction has been filtered. Figure 26 is very similar to Figure 23 but the main difference between the two Figures is that the details that lie on the x-direction are accentuated but the details that lie on the z-direction remain the same. Nevertheless, the trend of the dip angle $11.3°$ from the point $(0, 7)$ to the point $(50, 17)$ does not seen clear. We notice that this filter operate the same way

as the Sobel filter (Gonzalez, 2001), the only difference between our filter and the Sobel filter is that we use the second derivative instead of the first derivative.
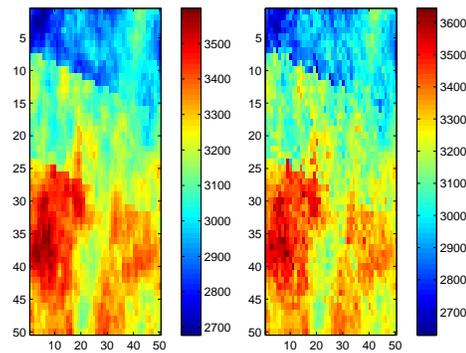

Figure 27: On the left: synthetic dataset, on the right, 1D finite-difference applied to a dataset with a 3×3 mask with respect to the direction of the velocity. x-direction filtered, z-direction unchanged.

On Figure 27, we used the 1D finite-difference operator and we took care of the velocity direction. The image looks clearer and sharper than the original seismic dataset and is different from the result found on Figure 26 because of the velocity direction.

On Figure 28, we have plotted the image of the angle θ which determine the direction of the velocity and give the norm of the projection of the filtered values on the x-direction. The angle vary between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$. We see that the angle θ vary a lot. This is due to the sensitiveness of the finite-difference operator itself (the finite-difference operator is very sensitive to the rapid variations of the data).
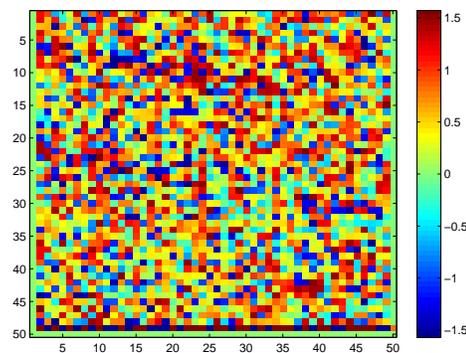

Figure 28: Plot of the angle θ corresponding to the direction of the velocity field at each meshpoint of the grid for the 1D finite-difference
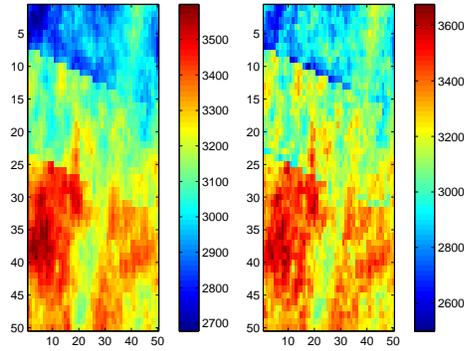
Figure 29: On the left: synthetic seismic dataset, on the right: operator based on the 1D Hamming window operator applied to the dataset with a 3×3 mask without taking care of the direction of the velocity. x-direction filtered, z-direction unchanged

We can see on Figure 29 that we get a good result. We see that the x-direction is filtered and the z-direction seems unchanged. As for the finite-difference , the result of this kind of filtering is to accentuate the high frequencies and mark down low frequencies on the x-direction. We see that the result of the filtering is different from the finite-difference because of the filter itself but leads to the same goal. If we take care of the direction of the velocity, we get Figure 31 and the angle θ is represented on Figure 30.
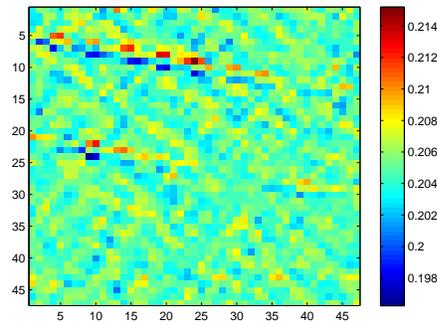

Figure 30: Plot of the angle θ correponding to the direction of the velocity field at each meshpoint of the grid for the 1D Hamming window operator

On Figure 30, we have plotted the angle θ, we see that the angle vary between 0 and 0.214 radian. We can easily recognize, as expected, the dip angle with a trend of approximately 0.199 radian which is $11°$ represented in blue.
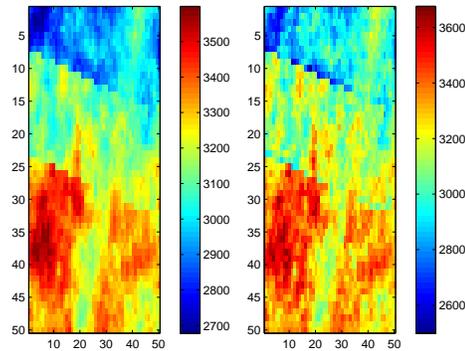
Figure 31: On the left: synthetic seismic dataset, on the right: operator based on the 1D Hamming window applied to a dataset with a 3×3 mask with respect to the direction of the velocity. x-direction filtered, z-direction unchanged

On Figure 31, we have plotted the result of the 1D highpass Hamming window applied to a seismic dataset, taking care of the direction of the velocity. We have filtered just the x-direction and we did not filtered the z-direction but unlike Figure 29, we took care of the direction of the velocity. We see that the image looks clearer compared to Figure 29.
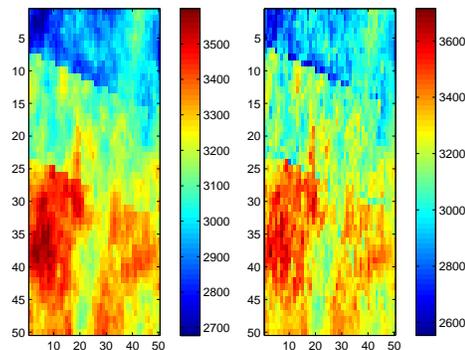

Figure 32: On the left: synthetic seismic dataset, on the right: modified operator based on the 1D finite-difference applied to a dataset with a 3×3 mask with rotation process applied

On Figure 32, we have plotted the result by using the 1D finite-difference with the rotation process described in Section 7.1. We used, to make this plot, five differents masks (equation (43), (45) (59), (60) and (61)). The condition parameter are if $0 < \theta < \frac{\pi}{16}$ we use the matrix (43), if $\frac{\pi}{16} < \theta < \frac{\pi}{8}$ we use the matrix (61), if $\frac{\pi}{8} < \theta \geq \frac{\pi}{4}$, we use the matrix defined in equation (60), if $\frac{\pi}{4} < \theta < \frac{\pi}{2}$, we use the matrix defined in equation (59), if $\theta \geq \frac{\pi}{2}$, we use the matrix defined in equation (45). We can see on Figure 32 that we got a quiet good result, since the image looks sharper. We also see that it is

easier to locate the dip angles because the latters look sharper. For example the dip angle on Figure 32 located at $(0, 7)$ following a trend of $11.3°$ to the point $(50, 17)$ is very accentuated compared to the original seismic velocity dataset.
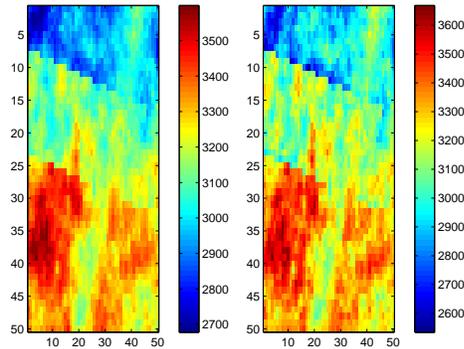

Figure 33: On the left: synthetic seismic dataset, on the right, modified operator based on the 1D Hamming window applied to a dataset with a 3×3 mask with rotation process applied

On Figure 33, we have plotted the result by using the 1D Hamming window with the rotation process described in Section 7.2. We used, to make this plot, four different masks (equation (57), (62) (63) and equation (58)). The condition parameter are if $0 < \theta < \frac{\pi}{8}$ we use the matrix (57), if $\frac{\pi}{8} < \theta < \frac{\pi}{4}$ we use the matrix (62) and if $\frac{\pi}{4} \le \theta < \frac{\pi}{2}$ we use the matrix expressed in equation (63) and if $\theta \ge \frac{\pi}{2}$ then we use the equation (58). We also see that it is easier to locate the dip angles because the latters look cleare and sharper.

If the dip angle is specified by the geology and is near 0, then the most appropriate mask would be an isotropic filter like the Laplacian or the 2D Hamming window in the 2D wavenumber domain. If the dip angle is constant or almost constant, but different from 0, then the most appropriate filter would be one filter obtained by linear interpolation oriented in the same direction than the dip angle. On the other hand, if the dip angle is unknown or varying a lot, the most appropriate filtering method would be an anisotropic filtering like the modified operator based on the 1D finite-difference or Hamming window with the rotation process applied on (as shown on Figure 32 and 33). We can also add that if we just want to filter one direction and not the other, as shown on Figure 27 and Figure 31, we would use the projection method described in equation (22).

9   Scaling penalty operators in the full object function

In Section 8, we saw that it was possible to apply all the operators on a synthetic seismic dataset that would have been generated after the inversion process. We emphasize

that since the inversion process is iterative, it is also necessary to filter the data at each iteration of the process to avoid errors. We then add that it is possible to process the data by incorporating the different operators within the inversion problem.

The main problem with this method is that the operators need to be chosen such that they can fit into the inversion problem. In our case we chose the Hamming window and the finite-difference operators because of that reason. We want to scale the penalty functions such that the result of the filtered data matches the original dataset. In effect, we encounter here a tradeoff between the problem of matching the data and the problem of enhancing the data.

As we saw earlier (see Section 4), since the accurate determination of the high frequencies both normally and laterally is the key step for improving seismic data, the quality of a solution of an inverse problem depends on the constraints imposed during the inversion process. We then have to decide the strength in which the penalty functions will filter on the dataset.

We will use in this section a Bayesian model in order to scale the Hamming window operator. In effect this method does not work for the finite-difference because the finite-difference operator is originally a highpass filter. In fact to make this method work for the finite-difference operator, we would have to find out the coefficients of its corresponding lowpass operator. We can write the likelihood as follow

$$f_{obs} = h_l f + \epsilon \qquad \epsilon \sim N(0, S) \tag{65}$$

where f_{obs} are the original seismic data or input signal that have to be processed, ϵ is the gaussian noise, meaning that f has prior mean μ and prior covariance matrix Σ. f_{obs} is a lowpass version, smearing the more rapidly the changing, underlying velocity f.

$$f \sim N(\mu, \Sigma) \tag{66}$$

We want to find the estimate $\hat{f}$ of f. We can also write the object function as a Gauss-posterior or a general object function.

$$O(f) = -\frac{1}{2}(f_{obs} - h_l f)' S^{-1}(f_{obs} - h_l f) - \frac{1}{2}(f - \mu)' \Sigma^{-1}(f - \mu) \tag{67}$$

We have then

$$\arg\max_f [O(f)] = [h_l' S^{-1} h_l + \Sigma^{-1}]^{-1}[\Sigma^{-1}\mu + h_l' S^{-1} f_{obs}] = \hat{f} \tag{68}$$

$\hat{f}$ is the posterior mean in the Bayesian model. It is also the maximum (minimum) solution in a 'combination solution' of 'matching the data' and keeping the a priori penalization. We emphasize that $\mu = 0$ if $f_{obs}^* = f_{obs} - \overline{f}_{obs}$ shifted around 0 ($\overline{f}_{obs}$ is the mean of all the observed data or input signal). The latter statement can act as a

condition for the scaling. We want f_{obs}^* to shift around 0. We can then write equation (68) as follow

$$\hat{f} = [h_l^{'} S^{-1} h_l + \Sigma^{-1}]^{-1}[h_l^{'} S^{-1} f_{obs}] \tag{69}$$

and then

$$\hat{f} = \overline{f}_{obs} + \hat{f}* \tag{70}$$

It is also necessary to specify the lowpass filter h_l, $S = \xi^2 I$ and $\Sigma = \sigma^2 I$ and $\lambda = \frac{\xi^2}{\sigma^2}$, we then have

$$\hat{f}* = [\frac{h_l^{'} h_l}{\xi^2} + \frac{I}{\sigma^2}]^{-1} \frac{h_l^{'}}{\xi^2} f_{obs}^* \tag{71}$$

$$= [h_l^{'} h_l + \lambda I]^{-1} h_l^{'} f_{obs}^* \tag{72}$$

We then need to specify the lowpass coefficients corresponding to the highpass, and this method will only work in one direction. We emphasize that the scale of the seismic dataset is dependent of the λ value. In effect, this parameter regularize the scaling procedure. We notice that, since we use matrices instead of a convolution product, if we are working with a 50×50 seismic dataset, then the h_l matrix is tridiagonal, have a 2500×2500 size and can only filter in one direction at a time. We can add that since the size of the filter matrix is very large, we will have problems with the inversion matrix involved in the computation of the $\hat{f}$ (see equation (72)). If we wanted to filter the seismic dataset in different directions, we would have used a diagonal block matrix with each block corresponding to a chosen filter matrix. Since the whole matrix is too large 2500×2500 to be inverted, we are going to scale only the first row of the seismic synthetic velocity dataset horizontally by using the 1D Hamming window operator without any rotation applied for $\lambda = 0.1$.
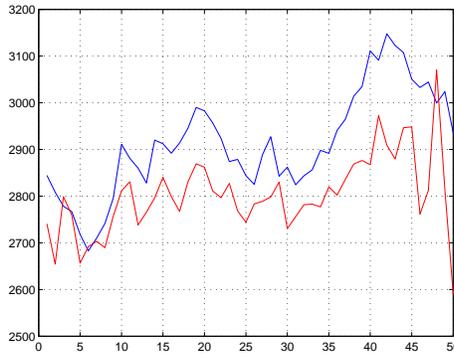
Figure 34: Bayesian scaling process applied on the first row of the synthetic seismic velocity dataset. The blue line represents the first row of the unscaled processed synthetic dataset. The red line represents the processed and scaled first row with $\lambda = 0.1$

We can see on Figure 34 that the red line corresponding to $\lambda = 0.1$ is situated below the blue line representing the unscaled and processed first row of the seismic dataset but

the two curves have approximately the same trend. We conclude then that the scaling procedure reduce the strength of the processing procedure. We notice that as λ increases, the processing procedure tends towards the mean value of the seismic dataset which is in this figure: 2918. As λ goes to 0, ξ^2 is much smaller than σ^2, indicating large precision in the data as compared to our prior knowledge. Hence we match the data in this case. We remark that λ act as a regularization parameter that reduce the strength of the processing procedure.

10 Discussion

All the programs and filters has been written in C (Kernighan and Ritchie, 2001) and the results have been plotted with Matlab. I had problems to optimize the implementation of the DFT, I finally used the separability property of the DFT to get a better running time (Gonzalez, 2001) described in Section 5. Indeed I could have used the Fast Fourier transform (FFT) instead of the DFT but the filter size was small enough (3×3) to avoid this optimalization problem. In effect if time had permitten, it could have been useful to work with larger masks, especially for the Hamming window operator because the latter needs to be discretize and then the more the operator discretization tend to the real function, the larger the filter is, the more precise results we have.

The dip angle direction of the synthetic seismic dataset in Section 8 corresponds to the filter matrix defined in equations (61) and (64). If the angle dip had been different, we would have extended the interpolation process to find out the correct filter matrix corresponding to the new direction.

We also notice that, the largest the filter size is, the more the linear interpolation approximation will be correct.

I used the convolution product to apply a filter on the synthetic seismic dataset in Section 8 because of implementation reasons and running time reasons instead of using the convolution theorem and operate transformations in the frequency domain.

We remark that the edges of the different images in Section 8 have not been filtered because we used the convolution product and there were not enough cells at the edges of the pictures to operate a convolution product. It is possible to avoid this problem by making the assumption that the dataset is a torus and then process the right edge with the left edge and the top edge with the bottom edge.

I also used usual highpass filters but in a larger project, I could have used high boost filtering or high frequency emphasis process to get a sharper image (Gonzalez, 2001).

The scaling method presented in Section 9 cannot be applied on large seismic dataset because of its running time which is too long and we must keep in mind that the scaling

process will also be incorporated in the full seismic tomography iterative process. We could have used others scaling procedures based on Bayesian models like the empirical Bayes approach to scale the operators. The scaling of the operators is often a subjective problem because the users decide either or not if the result of the scaling fit with the original seismic data. It is then important to have a regularization parameter as λ in Section 9 to adjust the scaling of the filtered data. We could have used other methods if time had permit like Thikonov regularization and a more sensitive regularization parameter(Mitsuhata, 2004).

11   Conclusion

We have implemented different filters and study their behaviour in the frequency domain. In effect, we were interested in the behaviour of the highpass operators because those kind of filters will help us to better find the location of the different mediums in the subsurface by marking down low frequencies.

In particular, we noticed that the Fourier transform corresponding to a given filter was rotation variant or not and a rotation applied to filter (in the spatial domain) involved a rotation on its corresponding filter in the frequency domain. Indeed we saw that the filters implemented in the 2D wavenumber domain were isotropic or rotation invariant and for that reason, we were not able to implement any constraint on the spatial variability of the velocity. We then suggested a method based on linear interpolations to construct a 2D filter based on an 1D finite-difference/Hamming window filter which revealed to make the resulting filters anisotropic. It was then possible to make constraints on the spatial variability of the velocity by choosing weither or not to rotate the operator by a certain angle of rotation corresponding to the direction of the velocity.

In Section 8, we have shown that the highpass filters was useful to accentuate the details and get a sharper image compared to the original seismic dataset. We tested both isotropic and anisotropic filter. We also used an algorithm to track the direction of the velocity and then update the direction of the filtering process by rotating the 1D operators. In practice, if the dip angle is near 0, then it would not be necessary to apply an anisotropic filter on the seismic velocity data and then in this case we could use the Laplacian operator or the 2D Hamming window operator in the 2D wavenumber domain. On the other hand, if the dip angle is different from 0, varying a lot or unknown, then it would be more appropriate to use an anisotropic filter. If the dip angle is constant but different from 0, then we will filter the dataset with one mask directed on the same direction as the dip angle.

In Section 9 we suggested a method based on a Bayesian model to scale the operators that revealed to be useless for large seismic dataset. We saw that the strength of the filtering procedure could be constrained by the scaling procedure by making vary the regularization parameter λ. Nevertheless other scaling methods could have been used to

regularize the filtering procedure on the velocity field. The combination of the filtering process and the scaling process in each iteration of the full depth seismic tomography problem would have given a new velocity field in which it would have been easier to locate and afterwards study, the different mediums in the subsurface.

In a larger project, we could have incorporated the different penalty operators in the iterative solution for the 2D depth tomography and study the different results by making vary the regularization parameter λ. It would had also been interesting to use the different filters on a real seismic dataset and use other filtering methods like anisotropic diffusion instead and compare the results with the results obtained with the 1D filters based on linear interpolation.

References

Alvarez, L., Lions, P., and Morel, J. *Image selective smoothing and edge detection by nonlinear diffusion*. 854-866. Siam Journal on Numerical Analysis, Vol 29,No.3 (1992).

Barner, K. and Arce, G. *Nonlinear Signal and Image Processing*. CRC Press (2004).

Burden, R. and Faires, J. *Numerical Analysis*. Brooks/Cole (2001).

Dines, K. and Lytle, R. *Computized Geophysical tomography*. 1065-1073. Proceedings of the ieee (1979).

Gonzalez, R. *Digital image processing*. Prentice Hall (2001).

Kernighan, B. and Ritchie, D. *The C programming language*. Prentice Hall (2001).

Lu, W.-S. and Antoniou, A. *Two-Dimensional Digital Filters*. Marcel Dekker, inc (1992).

Mitsuhata, Y. *Adjustement of regularization in ill-posed linear inverse problems by the empirical approach*. 213-239. Geophysical Prospecting (2004).

Sheriff, R. and Geldart, L. *Exploration Seismology*. Cambridge (1995).

Stork, C. and Clayton, R. *Linear aspects of tomographic velocity analysis*. 483-495. Society of Exploration Geophysicists (1991).

—. *Using Constraints to address the instabilities of automated prestack velocity analysis*. 404-419. Society of Exploration Geophysicists (1992).