*Physics is mathematical not because we know so much about the physical world, but because we know so little; it is only its mathematical properties that we can discover.*

Bertrand Russell

**University of Alberta**

# Multidimensional seismic data reconstruction using tensor analysis

by

Nadia Kreimer

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

Doctor of Philosophy
in
Geophysics

Department of Physics


Fall 2013
Edmonton, Alberta

# Abstract

Exploration seismology utilizes the seismic wavefield for prospecting oil and gas. The seismic reflection experiment consists on deploying sources and receivers in the surface of an area of interest. When the sources are activated, the receivers measure the wavefield that is reflected from different subsurface interfaces and store the information as time-series called traces or seismograms. The seismic data depend on two source coordinates, two receiver coordinates and time (a 5D volume).

Obstacles in the field, logistical and economical factors constrain seismic data acquisition. Therefore, the wavefield sampling is incomplete in the four spatial dimensions. Seismic data undergoes different processes. In particular, the reconstruction process is responsible for correcting sampling irregularities of the seismic wavefield.

This thesis focuses on the development of new methodologies for the reconstruction of multidimensional seismic data. This thesis examines techniques based on tensor algebra and proposes three methods that exploit the tensor nature of the seismic data. The fully sampled volume is low-rank in the frequency-space domain. The rank increases when we have missing traces and/or noise. The methods proposed perform rank reduction on frequency slices of the 4D spatial volume. The first method employs the Higher-Order Singular Value Decomposition (HOSVD) immersed in an iterative algorithm that reinserts weighted observations. The second method uses a sequential truncated SVD on the unfoldings of the tensor slices (SEQ-SVD). The third method formulates the rank reduction problem as a convex optimization problem. The measure of the rank is replaced by the nuclear norm of the tensor and the alternating direction method of multipliers (ADMM) minimizes the cost function.

All three methods have the interesting property that they are robust to curvature of the reflections, unlike many reconstruction methods. Finally, we present a comparison between the methods proposed in this thesis and other two reconstruction methods. This thesis demonstrates the suitability of tensor completion techniques for solving the simultaneous denoising and reconstruction problem.

# Acknowledgements

First I would like to thank my supervisor M.D. Sacchi for his support and encouragement during the duration of my PhD. His ability to grasp several different subjects and find connections and similarities among them have impacted my work in many ways. His passion for geophysics continue to be an inspiration to all of his students and colleagues. I am also grateful to all geophysics professors in the Department of Physics at the University of Alberta for sharing their work and knowledge. It has been an honor to be part of such a prestigious research group as is SAIG. Every member of this group contributed to my research in different ways through our regular meetings and occasional discussions. I am grateful to the sponsors of SAIG for providing us with financial support and their invaluable feedback in our annual meetings. I would also like to thank all my student colleagues in the Department of Physics for the entertaining coffee socials and conversations in the halls. Furthermore, I appreciate the support and guidance from Johannes Grimm during my internship in Shell Canada. In addition, I would like to thank the members of my committee for taking their time to carefully review this thesis. Finally, my deepest gratitude goes to Karol and my family, who have supported me in every step of the way.

# Contents

# List of Tables

# List of Figures

# List of symbols and abbreviations

## Variables

| | |
|---|---|
| $a, \alpha$ | Scalars |
| $\mathbf{a}$ | Vector |
| $\mathbf{A}$ | Matrix |
| $A_{ij}$ | Element of a matrix |
| $\boldsymbol{\mathcal{G}}$ | Core-tensor in the HOSVD |
| $\boldsymbol{\mathcal{X}}$ | Tensor |
| $X_{ijkl}$ | Element of a fourth-order tensor |
| $\boldsymbol{\mathcal{X}}(:, j, k, l)$ | Tensor fiber |
| $\boldsymbol{\mathcal{X}}(i, j, :, :)$ | Tensor slice |
| $\mathbf{X}^{(n)}$ | $n$-mode unfolding of tensor $\boldsymbol{\mathcal{X}}$ |

## Abbreviations

| | |
|---|---|
| ADMM | Alternating direction method of multipliers |
| ALS | Alternating least squares |
| AVAz | Amplitude-versus-azimuth |
| AVO | Amplitude-versus-offset |
| CANDECOMP | Canonical decomposition |
| CMP | Common midpoint |
| CP | (C)ANDECOMP/(P)ARAFAC |
| DFT | Discrete Fourier transform |
| FFT | Fast Fourier transform |

| | |
|---|---|
| *F-X* | Frequency-space |
| HOSVD | Higher-order singular value decomposition |
| MSSA | Multichannel singular spectrum analysis |
| MWNI | Minimum weighted norm interpolation |
| NDTM | Nondeterministic Turing machine |
| NFFT | Nonuniform fast Fourier transform |
| NMO | Normal moveout |
| NP | Nondeterministic polynomial |
| P | Polynomial |
| PARAFAC | Parallel factors |
| *Q* | Quality of the reconstruction |
| SEQ-SVD | Sequential singular value decomposition |
| *SNR* | Signal-to-noise ratio |
| SVD | Singular value decomposition |

# CHAPTER 1

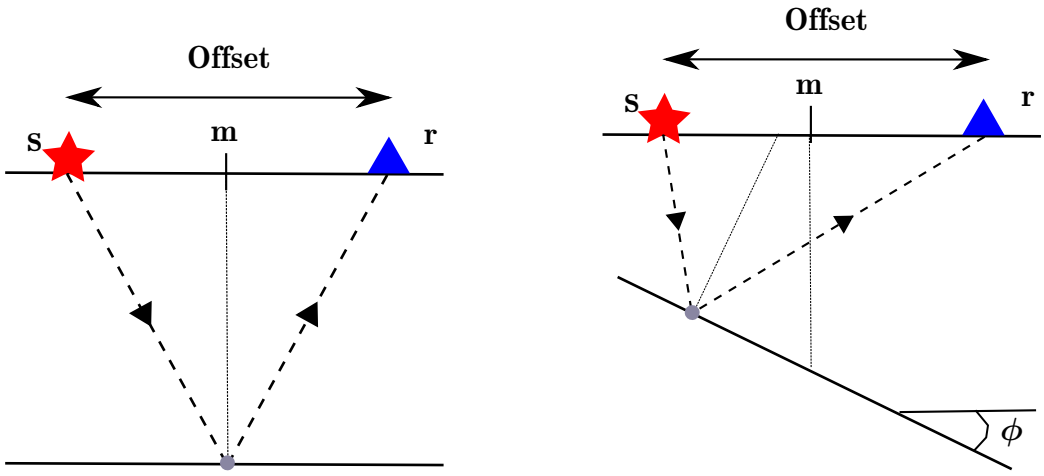
# Introduction

The seismic experiment consists of measuring the signal associated to the wavefields propagating in the subsurface. This signal is generated by a source and measured by detectors deployed on the ground. The sources are usually vibroseis, dynamite, air-gun and the detectors are often named receivers (geophones in land and hydrophones in water) (Yilmaz, 2001; Sheriff, 2002). The sources cause seismic waves to propagate in the subsurface. Energy partitioning occurs when the waves encounter a change in the elastic properties of the media, commonly denoted as interface. In an interface, some energy is transmitted and some energy is reflected. The steering vector orthogonal to the wavefront is the ray path and the laws of optics, such as Snell's Law, are used to simplify the analysis of wavefronts. The signal captured by a receiver, a time series, is called a trace or seismogram. In an ideal case of no background noise, a trace shows a wavelet or a wiggle shape at a time equal to the travel-time from the source to the interface and back to the surface to the receiver. The field data or "raw" data undergoes several processes both in the field and in a processing facility to achieve an interpretable image of the subsurface. In particular, this thesis concerns with one step of the seismic processing flow, called seismic data reconstruction. The final image obtained can be interpreted with the purpose of delineating oil and gas reservoirs, for crustal modeling and for near-surface applications like mining and hydrology. In hydrocarbon exploration, seismic data can be combined with well information to fill in the gaps where no wells are available. Naturally, seismic exploration is more economical than drilling wells and serves to minimize the possibility of drilling unproductive wells.

## 1.1 2D Seismic Experiment

The 2D seismic experiment contains sources and receivers deployed over a line. Therefore, this experiment will give a 2D image of the subsurface. Figure 1.1(a) shows the seismic experiment in 2D for a horizontal reflector, while Figure 1.1(b) does it for a dipping reflector. In these figures, the source is symbolized by a red star and the letter $s$, while the receiver is symbolized by a blue triangle and the letter $r$. If we consider all the traces obtained in a 2D seismic experiment, we obtain a 3D volume $D(t, s, r)$ with one coordinate for the source, one coordinate for the receiver and time. The point that lies midway between the source and the receiver is called "midpoint", and its coordinate is $m = \frac{s+r}{2}$. The distance between the source and the receiver is called "offset". The midpoint and offset coordinates are often used as an additional domain to sort the traces (Yilmaz, 2001). Seismic processing operates in one or either domains and often seismic data are transformed back and forth among these domains. It is a common practice to group the traces that contain a common coordinate, such as common source, common receiver, common midpoint (CMP) or common offset. In the midpoint domain, the travel time for a reflected wave can be described by a hyperbolic function (Lay and Wallace, 1995). In a real experiment we also find other arrivals like the direct wave and ground roll, but we are not concerned about these in this thesis.

When the interface is horizontal, the surface midpoint $m$ coincides with the depth midpoint; refer to Figure 1.1(a). However, when the interface contains dip, the surface midpoint is not a projection of the subsurface point of reflection and these two are different; refer to Figure 1.1(b), where the projection of the grey point to the surface is not $m$. The depth point is the point of reflection on the subsurface interface, while the midpoint is its projection on the surface. Therefore, grouping traces under CMP is not the same as grouping traces under common-depth point (CDP) when the interfaces contain dip. In other words, in the presence of dip CMP traces do not sample the same subsurface area. This discrepancy is alleviated by processes like dip moveout (DMO) and migration (Deregowski, 1991; Gray et al., 2001). These processes take care of making this error as small as possible. In both 2D and 3D acquisition, sources and receivers are relocated in the field such that we can obtain more than one trace per midpoint (in an area around the midpoint, a midpoint bin more precisely). The number of traces per midpoint is the fold number for that midpoint. In the data processing stage, traces are grouped per common midpoint and averaged together (stacked) to attenuate the noise and obtain a rough image of the subsurface (Mayne, 1962).




(a) Horizontal reflector

(b) Dipping reflector with dip equal to $\phi$.

Figure 1.1: 2D Seismic Experiment. The red star is the source, the blue triangle the receiver and the grey dot is the midpoint in the subsurface.

## 1.2 3D Seismic Experiment

In the 3D seismic experiment, the sources and receivers are not confined to a line. The seismic volume obtained through this type of acquisition depends on four spatial dimensions and time: two coordinates for the source and two coordinates for the receiver $D(t, s_x, s_y, r_x, r_y)$. Figure 1.2 clarifies this idea for the orthogonal acquisition case, where the receiver lines are orthogonal to the source lines (Sheriff, 2002). Only one ray path is shown in this figure. The plane view for one source and one receiver is displayed in Figure 1.3. Figure 1.4 gives an example of marine acquisition.

In this thesis, the real data we used for testing algorithms were recorded using an orthogonal geometry for land surveys (Figure 1.5). In this setting, the receiver lines are orthogonal to the source lines. This is the most usual geometry for onshore surveys because it is simple to deploy in the field (Cordsen et al., 2000). In general, all sources in an area are activated and some receiver lines are "rolled along" such that the receiver patch covers a new area. This procedure is repeated as many times as necessary until the whole area of interest has been surveyed. The direction along the receiver lines is called "inline" direction while the one along the source lines is called "crossline" direction. Other acquisition patterns used in land surveys are the brick geometry (Figure 1.6), non-orthogonal geometry (Figure 1.7), and zig-zag geometry (Figure 1.8) (Cordsen et al., 2000).

When the data sorting is implemented according to the source-receiver coordinates $s_x$, $s_y$, $r_x$, $r_y$, the data is said to be sorted in the source-receiver domain. However, other domains

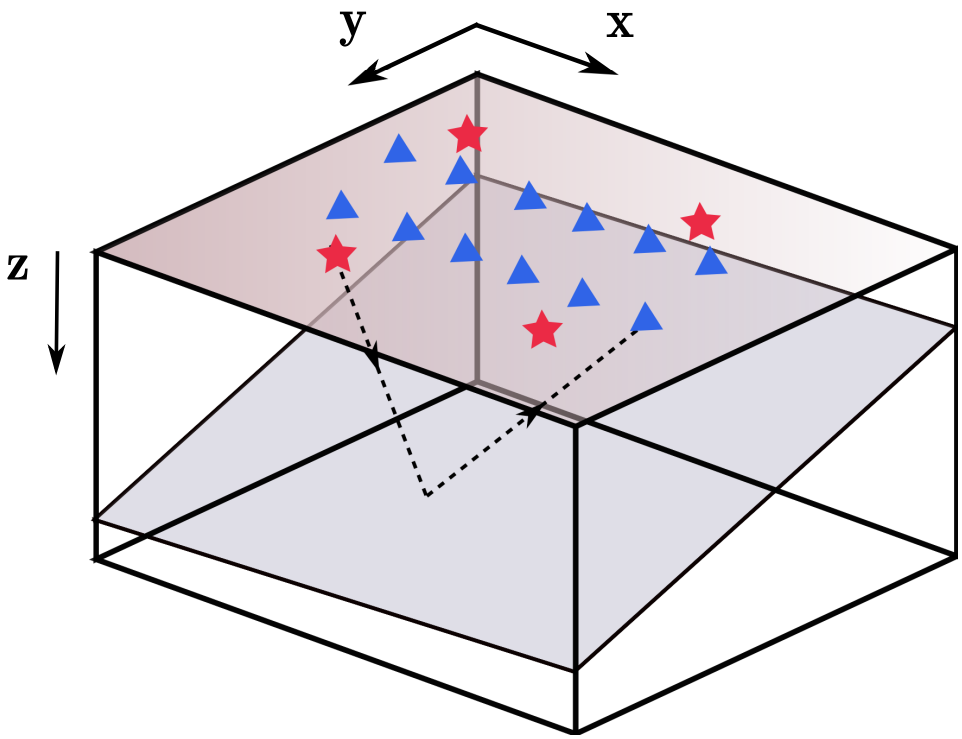



Figure 1.2: 3D seismic experiment, 3D view. Sources are represented by red stars and receivers by blue triangles. Notice that different source-receiver combinations do not lie over a line. The coordinates $x, y$ can represent East and North or any other arbitrary direction, while $z$ represents depth (or time in a seismic gather).

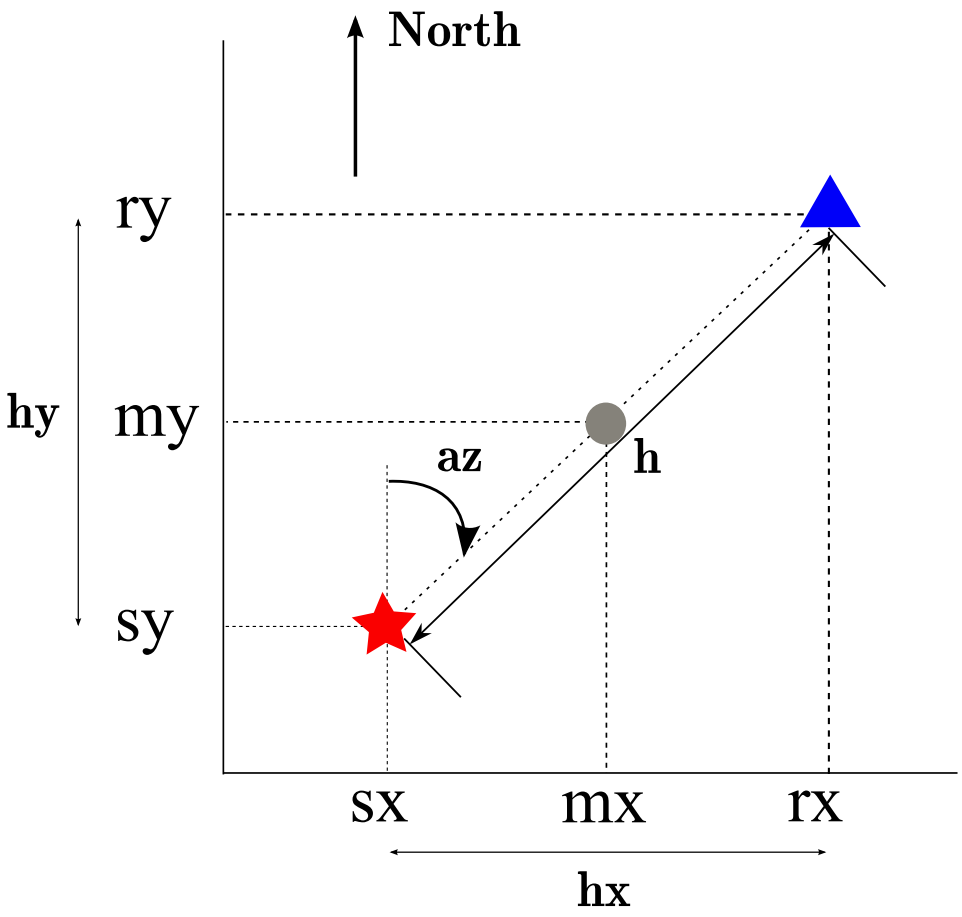



Figure 1.3: 3D seismic experiment, plane view. The red star is the source and the blue triangle is the receiver. The grey dot is the midpoint.

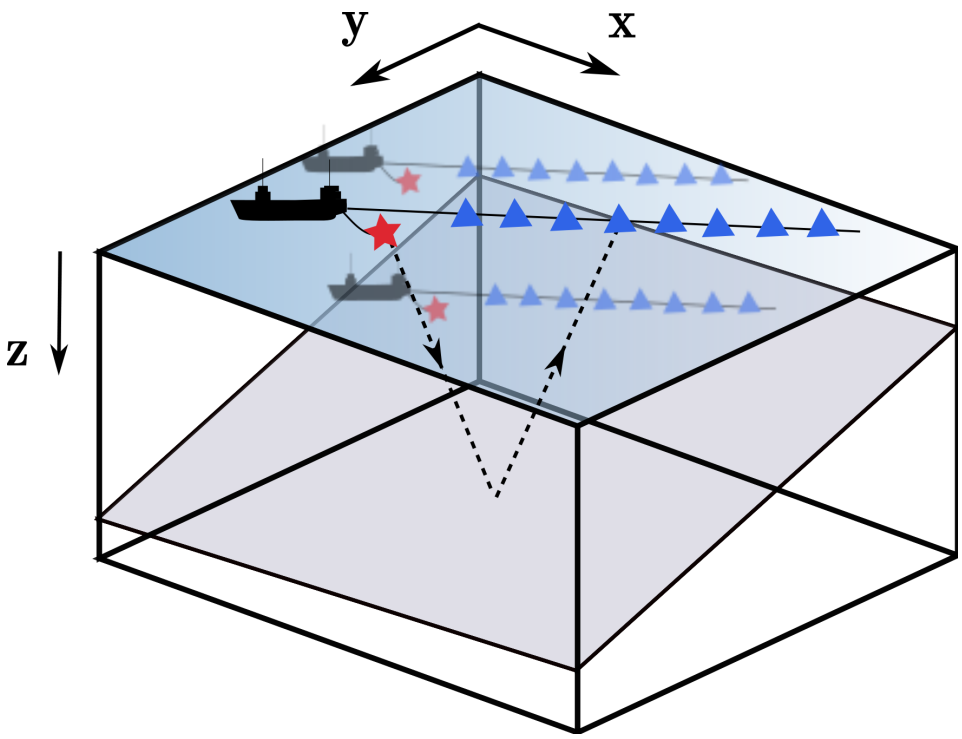



Figure 1.4: 3D seismic experiment, 3D view, marine survey. Sources are represented by red stars and receivers by blue triangles. The slightly transparent boats, sources and receivers show the boat at previous and later times. The dotted line is the ray-path for one source-receiver configuration.

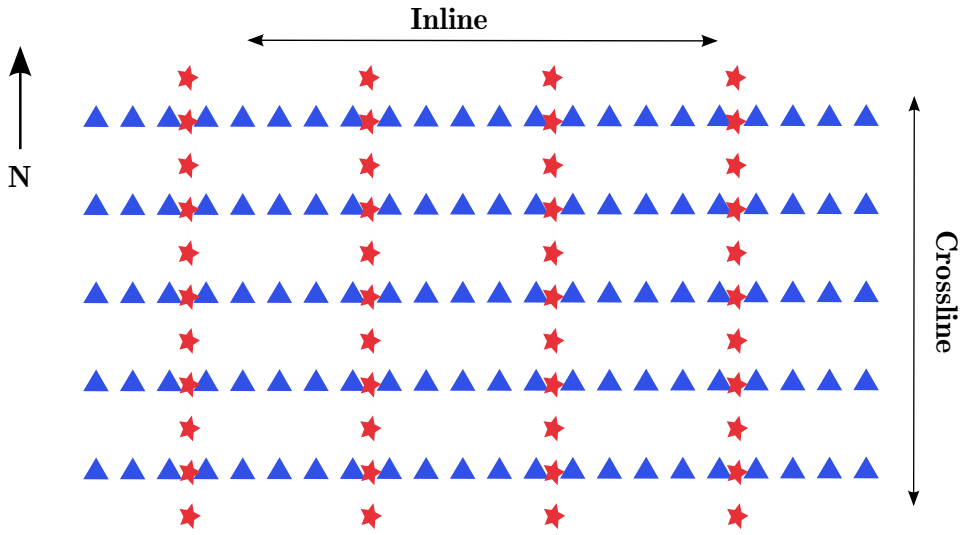



Figure 1.5: Land orthogonal geometry. The red stars represent the sources and the blue triangles the receivers.

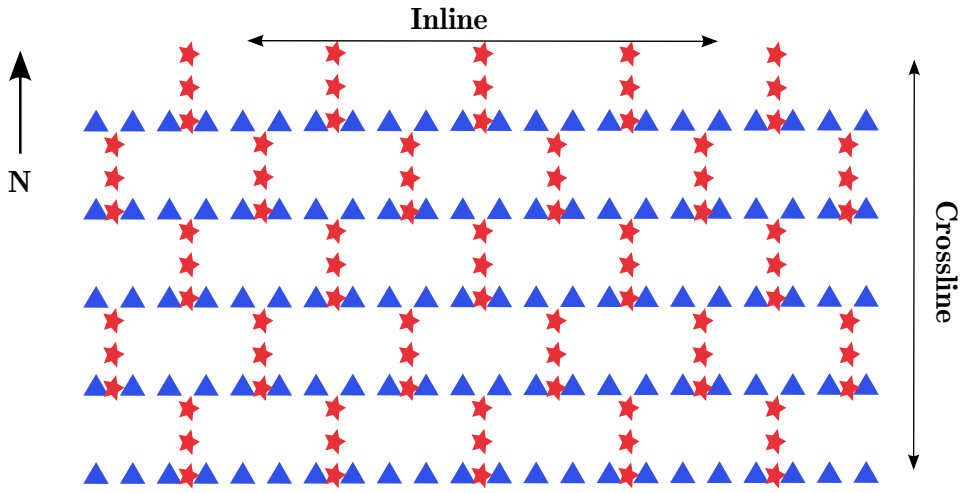



Figure 1.6: Brick geometry. The red stars represent the sources and the blue triangles the receivers.

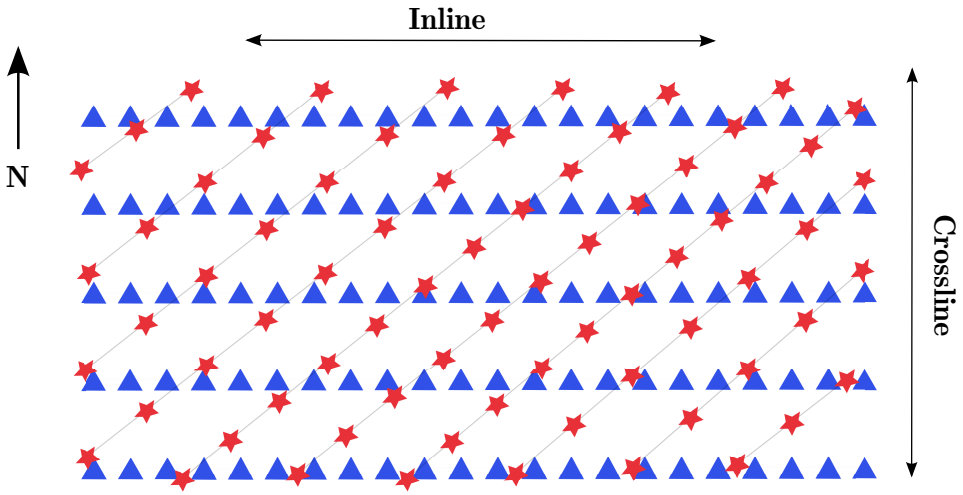



Figure 1.7: Non-orthogonal geometry. The red stars represent the sources and the blue triangles the receivers.

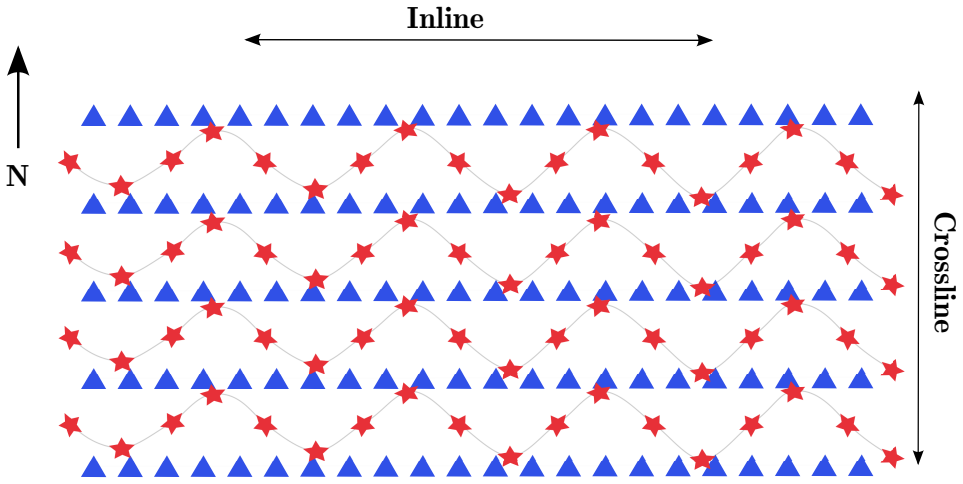



Figure 1.8: Zig-zag geometry. The red stars represent the sources and the blue triangles the receivers.

appear frequently in seismic processing and we will utilize them in the following chapters. For example, the midpoint-offset domain is an alternative domain where the new coordinates are obtained through the following transformation

$$m_x = \frac{s_x + r_x}{2}, \quad m_y = \frac{s_y + r_y}{2}, \quad h_x = \frac{s_x - r_x}{2}, \quad h_y = \frac{s_y - r_y}{2}, \tag{1.1}$$

where $m_x, m_y$ are often referred to as inline-crossline and $h_x, h_y$ as inline offset and crossline offset, respectively. In matrix notation, the transform between coordinates is

$$\frac{1}{2}\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix}\begin{pmatrix} s_x \\ s_y \\ r_x \\ r_y \end{pmatrix} = \begin{pmatrix} m_x \\ m_y \\ h_x \\ h_y \end{pmatrix} \tag{1.2}$$

The midpoint-offset-azimuth domain contains the absolute value of the offset and the azimuth

$$h = \sqrt{{h_x}^2 + {h_y}^2}, \quad az = \tan^{-1}\left(\frac{h_x}{h_y}\right). \tag{1.3}$$

We often assume that the azimuth is measured from the North, although other conventions could be adopted as well.

"Prestack" seismic data are the data before averaging all traces falling on the same midpoint bin, or before averaging along the offset dimension. This data volume depends on four spatial dimensions and time. The time sampling rate in seismic exploration is typically 2 or 4 ms and the samples are rarely discontinuous in time. The same cannot be said about the spatial coordinates. These are often poorly sampled in most or part of the survey because of physical obstructions, legal boundaries and limitations arising from logistic factors (Trad, 2009). This thesis will focus on designing algorithms for reconstruction of prestack volumes while simultaneously attenuating incoherent noise. In particular, we investigate the potential of tensor analysis for the solution of multidimensional reconstruction and denoising problems.

Reconstruction of seismic data is a relevant research topic because many processes are affected by missing traces. Processes that are sensitive to inproper spatial sampling are:

**Surface-related multiple removal (SRME)** (Verschuur et al., 1992): the 3D implementation of SRME requires a dense distribution of sources and receivers on a grid; i.e. complete source-receiver reciprocity. Conventional marine and land acquisition do not lead to an adequate sampling. Both reconstruction and extrapolation become a necessity for this method. In fact, Van Dedem and Verschuur (2005) designed a different method to tackle the 3D surface-related multiple prediction due to the sampling problem in the crossline direction. They use sparse inversion along the crosslines to accomodate for the sparse sampling in

that direction. Van Dedem and Verschuur (1998) compare two interpolation/extrapolation techniques prior to applying SRME on marine data. Van Borselen et al. (2004) give recommendations for the acquisition design and the processing flow to remediate the sampling irregularities in SRME. They propose to create a shooting overlap, decreasing the crossline spacing (given by the sail line) and making the crossline spacing a multiple of the streamer spacing. On the processing side, they recommend to recover missing traces, recover the near offsets via extrapolation and potentially upsample the shots. Baumstein and Hadidi (2004) outline a DMO-based reconstruction scheme that aims at missing shots recovery and regularization. Hanson et al. (2006) propose wavefield regularization first in the inline direction, followed by a 3D interpolation in the crossline direction. The final volume is the input to 3D SRME. Cai et al. (2009a,b) focus on the reconstruction problem in wide-azimuth surveys prior to 3D SRME. They show that interpolation and regularization with a Fourier-based method improve the results of SRME, wavefield extrapolation and reverse time migration (RTM) at the same time.

**Shear-wave splitting analysis**: the reconstruction of multicomponent data is usually implemented separately for each component. Stanton and Sacchi (2013a) present a technique for simultaneous reconstruction of all components using quaternions and projection onto convex sets (Abma and Kabir, 2006), denoted vector reconstruction. They claim that good quality 5D vector-reconstructed multicomponent datasets can be better than using the super-binning technique for the analysis of shear wave splitting. Stanton and Sacchi (2013b) show how 5D vector reconstruction can be used to populate the azimuths prior to shear wave splitting compensation. These effects are easier to identify after reconstruction and the reconstructed data are better prepared for the estimation of shear wave splitting parameters.

**Amplitude-versus-offset (AVO) and amplitude-versus-azimuth (AVAz) analysis:** The AVO/AVAz effect is a variation of the amplitude with the source-receiver distance or the angle of incidence. This variation occurs because of velocity, density and/or Poisson's ratio changes. The AVO/AVAz response is an indicator usually used for reservoir characterization and to distinguish between oil and gas. There are different classes of AVO/AVAz effects that depend on the impedance contrast of the surrounding rocks. AVO/AVAz is measured in prestack data and thus the presence of noise and incomplete sampling can affect its estimation (Sheriff, 2002). Sacchi and Liu (2005) analyze the effect of reconstruction with minimum weighted norm interpolation (Liu and Sacchi, 2004) before wave-equation AVA imaging. They show that the AVA curves extracted from common-image gathers follow the theoretical AVA trends, unlike the curves extracted from the incomplete datasets. Gray and Wang (2009) compare several workflows for extracting AVO attributes, including the role of interpolation via 5D minimum weighted norm interpolation. They prove that the best workflow includes 5D interpolation before imaging. Hunt et al. (2010) present a case study

where interpolation of prestack data prior to prestack migration and AVO analysis produces improved results on the AVO attributes, compared to other techniques like superbinning. They focus on increasing the offset sampling, necessary for AVO analysis, and using an interpolation technique that preserves the amplitudes. AVO analysis is also sensitive to the presence of noise. Therefore, an interpolation technique that can also provide some level of noise attenuation provides a better AVO estimation.

**Migration:** Abma et al. (2007) demonstrates with an example that, when using Kirchhoff migration, the irregularity of the azimuth distribution generates artifacts. In this type of migration, the sampling irregularities in the traces that are involved in the sum for a particular trace can introduce artifacts in the final migrated trace. Spatial aliasing (a sampling problem) is an important source of artifacts in migrated images. The spatial aliasing effect on the imaging operator in Kirchhoff is an issue addressed in the article by Biondi (2001). Instead of adapting the imaging operator to handle alias, we could implement a dealising interpolation method like the one proposed by Naghizadeh and Sacchi (2007) prior to conventional Kirchhoff migration. However, operator aliasing can still arise despite of correct data sampling. In this case, we need to also add constraints in the operator frequencies. The work from Sacchi and Liu (2005) is also a good example of reconstruction improving imaging. Trad (2008) shows the improvement of the image with interpolation before prestack migration on real data examples. The 3D implementation of wave-equation migration from Claerbout (1971) is also sensitive to the source-receiver coverage. Other imaging technique which requires correctly sampled midpoint and offset coordinates is common-azimuth migration (Biondi and Palacharla, 1996).

**Survey merging:** Merging is necessary when datasets acquired in different periods of time on adjacent or overlapping areas need to be processed as one survey. Often we desire to merge surveys because it is more economic than acquiring a new survey over the whole area (Gierse et al., 2010). Furthermore, time-lapse seismic monitoring involves acquiring different datasets over time to monitor reservoir changes. The repeatability of the seismic survey is key for the comparison between datasets (Leggott et al., 1999; Morice et al., 2000; Eiken et al., 2003) and regularization is able to improve it (Schonewille, 2003). In both of these cases we need to make several surveys homogeneous under the same grid. Although the repeatability problem in time-lapse monitoring has recently been minimized with the use of ocean bottom seismic (OBS) surveys, regularization can still be useful for merging new and vintage surveys.

## 1.3 Review of previous work

For more than a decade, with reservoirs becoming mature and approaching the end of its productivity, interest has arisen to lower seismic data acquisition costs. In this situation, reconstruction methods can help alleviate the costs of acquisition. However, reconstruction cannot fully replace field measurements. Many methods exist to alleviate the problem of spatial sampling irregularities using different numbers of spatial dimensions.

Reconstruction methods for seismic data can be divided into four groups, listed in the subsections below.

### 1.3.1 Reconstruction methods based on signal processing principles

These methods are the type of methods most commonly used by industry for production processing. This group of methods exploit the property of the signal in a supplementary space such as Fourier, curvelet and Radon domain. The reconstruction problem is underdetermined and becomes an inverse problem where the coefficients of a transform are the by-product of an inversion. In general, constraints are imposed on the signal or on the coefficients to better condition the inverse problem. The most commonly exploited property is sparsity in the frequency-space ($F - X$) domain. Furthermore, most of these methods have a linear event assumption; i.e. the signal in the frequency domain is a superposition of plane waves. Methods in this category may be divided into two, according to their ability to preserve the original trace locations:

a) Methods which are able to preserve the exact location of the traces: these methods usually require the use of the discrete Fourier transform (DFT) and are unable to take advantage of its faster counterpart, the fast Fourier transform (FFT). Naturally, the DFT implementation is more computationally intensive than the FFT. In the presence of an irregular grid, the DFT is often referred to as the nonuniform DFT (NDFT). Duijndam and Schonewille (1999) propose the nonuniform fast Fourier transform (NFFT) and show an improvement on the computation time of the NDFT, although the cost is still higher than that of the FFT. Zwartjes and Gisolf (2007) use the NFFT in an inversion scheme that includes non-quadratic weights to stabilize the inversion. Furthermore, Zwartjes and Sacchi (2007) also utilize the NFFT in an inversion but only in the lower unaliased frequencies, based on the ideas from Gulunay (2003), followed by an estimation of a weight matrix to perform the inversion of the higher frequencies. Their method is called Fourier reconstruction via sparse inversion (FRSI). They construct the weight matrix applying FRSI in two passes: the first pass estimates coefficients that in the second pass allow them to distinguish between aliased and unaliased coefficients in the spectrum. The antileakage Fourier transform from Xu et al.

(2005) is the most popular method from this group. This method proposes to estimate the largest Fourier coefficient on an irregular (multidimensional) grid and subtract it from the spectrum, with the assumption that the largest coefficient causes most of the spectral leakage. The method continues iteratively with a new input until it estimates all Fourier coefficients. The non-equispaced curvelet reconstruction with sparsity-promoting inversion (NCRSI) from Hennenfent et al. (2010) is a method that exploits sparsity in the curvelet domain and it can handle irregular sampling. This method is only able to handle up two spatial dimensions, because no practical implementation for higher dimensions exists today.

b) Methods which require binning to a regular grid: these methods require binning the data before the reconstruction. The process of binning implies placing the sampled traces on a regular grid. For each bin, we assign the trace that is closest to that bin in the grid and discard the rest of the traces within that area. In a few words, we perform a nearest neighbor interpolation of the traces to the grid. After binning, the grid contains bins with traces and empty bins. This grid is the input to a reconstruction method. Binning is not an ideal process and it can produce artifacts on the final image. Especially when the data contains curvature, binning can introduce jitter in the reflections travel-time curves. Therefore, binning usually follows a normal-moveout (NMO) correction to mitigate the binning errors (Trad, 2009). Despite the disadvantages of the binning process, literature abounds in reconstruction methods which require binning. The methods in this category operate one frequency slice at a time. The best known method in this area is minimum weighted norm interpolation (MWNI) from Liu and Sacchi (2004). This method is based on the inversion of a cost function that contains a misfit term and a term that constrains the solution to be bandlimited with a prior spectral shape. This algorithm uses conjugate gradients and the FFT for fast matrix-vector multiplications. MWNI is currently a widely used method in industrial applications. Projection onto convex sets (POCS) from Abma and Kabir (2006) is an iterative method that after performing FFT on a frequency slice, it thresholds the amplitudes, performs inverse FFT and reinserts the original traces in the final volume. The threshold value changes with the number of iterations from a large to a small value. The methods in this category frequently struggle when the missing traces pattern in the grid is regular. For instance, a grid in the source-receiver domain has regularly missing traces if we want to up-sample the source and/or receiver lines. In other words, if we want to create new lines of sources and/or receivers lines among existing ones, the sampling pattern will be regular in one or two directions in the survey. This type of sampling introduces spatial alias and constraints are needed to remove the alias. The generalized $F-K$ trace interpolation (GFKI) from (Gulunay, 2003) uses a replication of the original transform and a multiplication between this replica and an operator designed to reinforce the non-aliased coefficients. GFKI has been designed explicitly to deal with the up-sampling problem. MWNI in its original form by Liu and Sacchi (2004) does not lead to satisfactory results

in the presence of spatial alias and its performance is best when the sampling is random in all spatial dimensions (Naghizadeh and Sacchi, 2010b; Cary, 2011). However, strategies including spectral weights and bootstrapping (Naghizadeh and Sacchi, 2010b; Naghizadeh, 2012) aid to some extent in the presence of alias, especially when the aliased spectrum does not overlap with the original spectrum.

The Radon transform is an alternative to the Fourier transform and is exploited in the methods from Kabir and Verschuur (1995) and Trad et al. (2002). These methods use the parabolic Radon transform and hyperbolic and elliptical Radon transforms, respectively. They are mostly used for regularization of CMP gathers.

### 1.3.2 Reconstruction methods based on prediction filters

These methods use the property of linear prediction of the signal in the $F - X$ (Spitz, 1991; Porsani, 1999) or the $F - K$ (Naghizadeh and Sacchi, 2007, 2010c) domain. In these methods, the data from the low frequencies are used to estimate prediction error filters that can recover data from the high frequencies. Spitz (1991) introduced the original idea. Spitz's method constructs prediction error filters in the $F - X$ domain that are the solution to a linear system of equations that originates in the spatial predictability of the signal when assumed to be a superposition of plane waves. This method is able to upsample the data in a grid and it is not designed for randomly missing patterns. Porsani (1999) improved on the work of Spitz by simplifying the system of equations to solve for the filters. Naghizadeh and Sacchi (2007) use MWNI to reconstruct the unaliased low frequencies followed by a calculation of the prediction error filters to dealias the higher frequencies. Naghizadeh and Sacchi (2009) derive adaptive prediction error filters by exponentially weighted recursive least squares to avoid windowing and the linear events assumption. In other words, this method is able to reconstruct not only spatially aliased events, but reflections containing curvature.

### 1.3.3 Wave-equation based methods

Wave-equation based methods reconstruct seismic data through wave propagation principles. In general, the reconstruction is achieved by inverting an operator that links the subsurface model to the data. The operator is built using linearized wave-equation operators and its inversion maps the data to the desired geometry. These methods require knowledge of the velocity field, contrary to methods based on signal processing. Although computationally more expensive than methods in a) and b), wave-equation based methods are able to preserve the original position of the sampled traces and have the advantage

that they use the physics of wave-propagation when generating new traces. Ronen (1987) proposed DMO to reconstruct aliased data. He creates a system of equations where the inverse DMO operator maps the zero-offset section to common offset sections, while missing traces are considered zeros. He proposes using a conjugate gradients scheme where each iteration involves DMO stacking and transpose DMO stacking, combined with a spectral balancing procedure. Nemeth et al. (1999) propose using least-squares migration for the reconstruction of seismic data. They modify the normal equations and add regularization to the preconditioned conjugate gradients scheme. The migrated sections are then used to predict missing data. The intention of this method is to improve Kirchhoff migration in the case of incomplete data. Stolt (2002) creates data mapping operators suitable for infilling missing data and regularization. The operators are based on a straight ray assumption and contain a time-dependent weight factor for each trace which depends on the original gather. Fomel (2003) designs a finite-difference offset-continuation filter in the log-stretch frequency domain that can interpolate seismic data. Malcolm et al. (2005) deal with the reconstruction problem in the presence of caustics. They concatenate an imaging operator, a modeling operator and a restriction operator that map the input data to the desired data. Kaplan et al. (2010) use shot-profile migration and demigration operators to obtain reconstructed gathers assuming a constant velocity. They incorporate these operators in a least-squares migration scheme, allowing them to penalize the aliased components. Their model space is parametrized by pseudodepth and lateral position. They are able to separate between signal and alias in the model space exploiting the band limitation of the adjoint operator according to the maximum dip of the reflectors.

### 1.3.4 Rank reduction methods

Rank reduction methods perform rank reduction on matrices or tensors in the $F-X$ domain. These methods make the assumption that seismic data in $F-X$ is low-rank, where the data is described by a Hankel or Toeplitz matrix or by a tensor. The low-rank assumption is analogous to the sparsity assumption from the methods in the first category listed before. Methods in this category require binning prior to reconstruction and are not able to preserve the exact location of the traces. Interpolation with rank reduction methods also requires to develop an antialias strategy. The latter has been recently shown by Naghizadeh and Sacchi (2013). We divide this group of methods into two:

a) Methods which perform rank reduction on matrices: the frequency slices are matrices that contain a Hankel structure, as many Hankel matrix levels as dimensions, so that prestack data reconstruction requires block Hankel matrices. Trickett et al. (2010) first proposed Cadzow reconstruction for both reconstruction and random noise attenuation of seismic data. For prestack seismic data, the matrix used is a four level Hankel matrix. The rank

reduction operation by itself is not enough to recover the missing observations. The authors claim that they resort to matrix completion to recover the missing elements. The final reconstructed trace is obtained by averaging along the antidiagonals of the Hankel (or block Hankel) matrix. The method from Oropeza and Sacchi (2011), called multichannel singular spectrum analysis (MSSA), is similar to Cadzow's method and uses the truncated SVD for the rank reduction. The authors present a POCS-like reinsertion algorithm for the recovery of the missing traces. Furthermore, they propose to use the randomized SVD to avoid calculating the SVD on the large Hankel matrices. An important fact regarding Cadzow/MSSA is that if the data is a superposition of $k$ plane waves in the $F-X$ domain, the rank of the Hankel matrix is also $k$. Therefore, adopting small spatio-temporal windows where the linear event assumption is valid guarantees that the rank reduction operation will preserve most of the signal. Gao et al. (2013) introduce a fast version of Cadzow/MSSA where they embed the data in block Toeplitz matrices instead of Hankel matrices. The rank reduction operation uses the Lanczos bidiagonalization algorithm instead of the truncated SVD. This algorithm is faster than doing SVD on large Toeplitz matrices because it uses the FFT to multiply matrices and vectors (it avoids storing in memory large Toeplitz matrices). Naghizadeh and Sacchi (2013) adapts Cadzow/MSSA to the reconstruction of aliased data. They use the rank-reduced Hankel matrix from the low frequencies for the rank reduction of the Hankel matrix in the high frequencies, with the aid of an upsampling matrix.

b) Methods which perform rank reduction on tensors: Rank reduction on tensors does not require a Hankel structure and it is the topic of this thesis. Frequency slices are multidimensional arrays or tensors that are subject to the low-rank assumption. However, tensor rank is not equivalent to matrix rank and there are several ways to define it. In this thesis we use the concept of $n$-rank of a tensor (refer to Chapter 2). We assume that the $n$-ranks of the fully sampled volume are small, similarly to matrix rank reduction techniques. Appendix A explores the validity of this assumption. In this thesis, we present three methods based on tensor completion; namely, the HOSVD reconstruction method (Kreimer and Sacchi, 2012d) in Chapter 3, the SEQ-SVD method (Kreimer and Sacchi, 2012a) in Chapter 4 and nuclear norm based reconstruction method (Kreimer and Sacchi, 2012c) in Chapter 5. The main difference between the tensor methods and the matrix methods is that the former are robust to curvature in the reflections. In other words, they are not subject to the plane wave assumption as much as the latter methods are.

Recent work by Trickett and Burroughs (2013) combines groups a) and b) and is called Hankel tensor completion. They form Hankel tensors by creating two tensor orders in each dimension. Therefore, with four spatial coordinates they create an eighth-order tensor. They use the CP decomposition immersed in a ALS algorithm (refer to Chapter 2) for the rank reduction step. They compare the performance of this method with Cadzow and notice that the former method is more robust to large gaps and large levels of sparsity than

Cadzow. No comparison exists between the Hankel tensor completion method and any of the methods proposed in this thesis.

Throughout the thesis, we use the term "reconstruction" to imply filling empty bins with new traces and/or denoise existing traces. The term "interpolation" should be reserved for methods that attempt to infill regularly sampled data. In other words, this thesis deals with the reconstruction problem and no attempt has been made to utilize the proposed methods for seismic trace interpolation. Other techniques can be effectively used to upsample regular data (Spitz, 1991).

## 1.4 Convex optimization methods

The alternating direction method of multipliers will be the basis of the algorithm to reconstruct seismic data elaborated in Chapter 5. To understand this method, we first need to review the classical method of multipliers, the quadratic penalty method and the dual ascent method. These methods have in common that they solve a series of subproblems instead of the original problem. They do so by adding the constraints as additional terms in the objective function (Nocedal and Wright, 2006).

As a reminder for the reader, a convex set $X$ is a set where any two points inside the set form a line segment that is contained in the same set. A function $f$ on a convex set $X$ is convex if $f(ax_1 + (1-a)x_2) \leq af(x_1) + (1-a)f(x_2)$, for any $a \in [0,1]$ and any points $x_1, x_2 \in X$. The function $f$ is said to be strictly convex if the inequality in the previous expression is strict (no equal sign). For example, any vector norm $|x|^p$ with $p \geq 1$ is convex, the exponential function is strictly convex, etc.

The classical equality-constrained convex optimization problem consists of

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & h(\mathbf{x}) = 0, \end{aligned} \tag{1.4}$$

where the function $f : \mathbb{C}^n \to \mathbb{C}$ is convex and $h : \mathbb{C}^n \to \mathbb{C}^m$. For simplicity, we will consider only equality constraints. However, constrained convex optimization problems often have a mix of constraints which are equalities and inequalities. Notice that we can rewrite any equality constraint as an expression $h(\mathbf{x}) = 0$. For example, we can reformulate the constraint $\mathbf{Ax} = \mathbf{b}$ for $\mathbf{x} \in X$, where $X$ is a subset of $\mathbb{C}^n$, as $h(\mathbf{x}) = \mathbf{Ax} - \mathbf{b} = 0$.

### 1.4.1 Penalty function method

The basic concept of the penalty method involves creating a cost function where we add a sum of penalty terms consisting of the constraint violations to the function we want to minimize, multiplied by a positive coefficient called the penalty parameter. As this coefficient grows, the constraints are given more weight, such that the minimum to the unconstrained problem is closer to the solution space of the constrained problem. We will consider quadratic penalties. The objective function for this method is

$$\mathcal{L}_{qp}(\mathbf{x}) = f(\mathbf{x}) + \frac{\beta}{2}\,|h(\mathbf{x})|^2, \tag{1.5}$$

where $\beta > 0$ is the penalty parameter. As $\beta \to \infty$ the violations to the constraints (solutions which do not satisfy $h(\mathbf{x}) = 0$) are penalized harder. The quadratic penalty method solves the series of problems

$$\begin{aligned} \text{minimize} \quad & \mathcal{L}_{qp,\beta_k} && (1.6)\\ \text{subject to} \quad & \mathbf{x} \in X, && (1.7) \end{aligned}$$

where $\beta_k$ is a series of penalty parameters that satisfy $0 < \beta_k < \beta_{k+1}\ \forall k$, $\beta_k \to \infty$. This method requires increasing $\beta$ sequentially to converge. The minimization of $\mathcal{L}_{qp,\beta_k}$ is an unconstrained minimization such that any method suitable for unconstrained minimization serves the purpose (e.g quasi Newton, conjugate gradients, Newton, steepest descent, etc).

If there is a good initial guess $\beta_0$ and well chosen stepsizes for $\beta$, this method can converge in a few iterations for a certain $\beta_k$. The gradient of the objective function $\mathcal{L}_{qp,\beta_k}$ is often taken as a proof that a reasonable solution $\mathbf{x}_k$ has been reached. For more details on the convergence of this algorithm refer to Bertsekas (1996). Numerical instabilities can arise when $\beta_k$ becomes very large, mainly because of the ill conditioning of the Hessian $\nabla\mathcal{L}_{qp,\beta_k}$ near the minimum. Newton's method seems the best one to use for large $\beta_k$ although it can fail if the choice of the starting point $\mathbf{x}_k$ is not close enough to the solution. In synthesis, the method of quadratic penalty functions implies choosing a good compromise between the speed of the convergence and the possibility of ill-conditioning.

### 1.4.2 Dual ascent

The dual ascent method solves problem 1.4 by maximizing the dual of the cost function (Boyd et al., 2011). The cost function for this problem is defined as the conventional

Lagrangian

$$\mathcal{L}_{da}(\mathbf{x}, \mathbf{w}) = f(\mathbf{x}) - \mathbf{w}^T h(\mathbf{x}), \tag{1.8}$$

where $\mathbf{w}$ is the vector of Lagrange multipliers, also called dual variable. Minimizing the objective function in equation 1.8 is what we call the primal problem. The Lagrange dual function is defined as

$$g(\mathbf{w}) \;=\; \inf_x \mathcal{L}_{da}(\mathbf{x}, \mathbf{w}) \;=\; \inf_x \big( f(\mathbf{x}) - \mathbf{w}^T h(\mathbf{x}) \big). \tag{1.9}$$

The dual function can take values of $-\infty$ when $\mathbf{x}$ does not have a lower bound. The dual problem consists on maximizing the dual function in equation 1.9 over $\mathbf{w}$. When strong duality applies, we can assume that the best values for the Lagrangian (equation 1.8) and the Lagrange dual (equation 1.9) coincide. Therefore we can use an optimal value $\widetilde{\mathbf{w}}$ found by maximizing the dual to obtain the optimal value $\widetilde{\mathbf{x}}$ of the primal problem solving $\arg\min_x \mathcal{L}_{da}(\mathbf{x}, \widetilde{\mathbf{w}})$.

The dual ascent method utilizes gradient ascent to solve the dual problem. We assume that $\nabla g(\mathbf{w})$ exists, which in our formulation is equal to the constraint function (Boyd et al., 2011). First, we obtain a minimum for $\mathbf{x}$ with $\mathbf{x}^\dagger = \arg\min_x \mathcal{L}_{da}(\mathbf{x}, \mathbf{w})$ and we evaluate $\nabla g(\mathbf{w}) = h(\mathbf{x}^\dagger)$. The dual ascent method works iteratively by solving

$$\begin{aligned} \mathbf{x}^{k+1} &= \arg\min_x \mathcal{L}_{da}(\mathbf{x}, \mathbf{w}^k) \\ \mathbf{w}^{k+1} &= \mathbf{w}^k - \mu^k \, h(\mathbf{x}^\mathbf{k}), \end{aligned} \tag{1.10}$$

where $\mu^k > 0$ is the step-length. If $\mu^k$ is chosen conveniently, the dual increases with each iteration $g(\mathbf{w}^{k+1}) > g(\mathbf{w}^k)$ and therefore the name of the method refers to the ascent of the dual. Under certain conditions and if $\mu^k$ is chosen correctly, $\mathbf{x}^k$ and $\mathbf{w}^k$ should converge to the primal and dual optimal points, respectively (Boyd et al., 2011). However, the conditions of this method are often not fulfilled in practice and this method cannot be used. A useful property of the method of dual ascent is that if the function $f(\mathbf{x})$ is separable (can be expressed as a sum of functions for each of its variables), the minimization step in the algorithm 1.10 can be split into separate minimization problems and implemented in parallel computing (Boyd et al., 2011).

### 1.4.3 Method of multipliers

The method of multipliers (Hestenes and Stiefel, 1952; Powell, 1969) uses the same cost function as the penalty method, equation 1.5, with the addition of a term with Lagrange

multipliers. This new cost function is called the *augmented Lagrangian* and has different properties than the cost function in equation 1.5. The method of multipliers is a blend of the penalty function and dual ascent methods. With a particular choice of $\mu^k$, it can be interpreted to be the method of dual ascent with $f(\mathbf{x})$ replaced by $f(\mathbf{x})$ plus the addition of a quadratic term containing the penalty constraint.

The augmented Lagrangian for problem 1.4 is

$$\mathcal{L}_m(\mathbf{x}) = f(\mathbf{x}) - \mathbf{w}^T h(\mathbf{x}) + \frac{\beta}{2} |h(\mathbf{x})|^2, \tag{1.11}$$

where $\beta > 0$ is the penalty parameter, as before, and $\mathbf{w}$ is the vector of Lagrange multipliers. Hestenes and Stiefel (1952) proved in a theorem that

> "if $\widetilde{\mathbf{x}}$ is a nonsingular minimum of problem 1.4, then it exists a multiplier $\mathbf{w}$ and a constant $\beta$ such that $\widetilde{\mathbf{x}}$ is an *unconstrained* local minimum of the cost function 1.11".

According to this theorem, it is also valid that

> "if $\widetilde{\mathbf{x}}$ is a minimum of function 1.11 and also $h(\widetilde{\mathbf{x}}) = 0$ then $\widetilde{\mathbf{x}}$ is a constrained minimum of $f$ subject to $h = 0$".

Through this theorem we guarantee the equivalence of the constrained and unconstrained problems.

According to Hestenes and Stiefel (1952), the method of multipliers operates by minimizing the cost function 1.11 successively for a certain vector of Lagrange multipliers $\mathbf{w}^k$ and a penalty parameter $\beta^k$ (starting from certain values $\mathbf{w}^0$ and $\beta^0$). If the minimum found is denoted $\mathbf{x}^k$ at iteration $k$, the Lagrange multiplier is updated by the following rule

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \beta^k \, h(\mathbf{x^k}). \tag{1.12}$$

Subsequently, the penalty parameter is increased such that $\beta_{k+1} \geq \beta_k$ and the process continues. One of the important advantages of this method over the quadratic penalty method is that $\beta$ does not need to be increased rapidly to achieve convergence. This method does not suffer from the ill-conditioning of the quadratic penalty method and the choice of the starting value $\mathbf{x}^0$ is not that critical. Furthermore, the convergence rate of this method is improved with respect to the method of penalty functions and it has a linear or superlinear convergence rate. A drawback from the method of multipliers is that if the function $f$ is separable, the minimization of $\mathcal{L}_m(\mathbf{x})$ cannot be separated in the minimization step because of the presence of the quadratic term, which is a problem in some applications (as we will

see later). For convergence properties and theorems, refer to Bertsekas (1996) and Nocedal and Wright (2006).

The reason for the multiplier update in equation 1.12 can be explained by the Karush-Kuhn-Tucker (KKT) conditions (Kuhn and Tucker, 1951; Karush, 1939). These are first-order necessary conditions that guarantee that if a local minimum $\widetilde{\mathbf{x}}$ exists for problem 1.4, there is a Lagrange multiplier $\widetilde{\mathbf{w}}$ satisfying the properties

$$\nabla \mathcal{L}_m(\widetilde{\mathbf{x}}, \widetilde{\mathbf{w}}) \;=\; 0 \tag{1.13}$$

$$h(\widetilde{\mathbf{x}}) \;=\; 0 \tag{1.14}$$

$$\widetilde{\mathbf{w}}\, h(\widetilde{\mathbf{x}}) \;=\; 0. \tag{1.15}$$

These properties utilize the gradient of the cost function 1.11, which is the reason they are called "first-order" conditions. The property 1.15 means that either the constraint is active or the Lagrange multiplier is zero, or both (Nocedal and Wright, 2006). In our case, the KKT conditions are summarized in the expression

$$\nabla \mathcal{L}_m(\widetilde{\mathbf{x}}, \widetilde{\mathbf{w}}) = \nabla f(\widetilde{\mathbf{x}}) - (\, \widetilde{\mathbf{w}} - \beta\, h(\widetilde{\mathbf{x}})\,)\, \nabla h(\widetilde{\mathbf{x}}) = 0. \tag{1.16}$$

By theorem 17.2 from Nocedal and Wright (2006), $\widetilde{\mathbf{w}}$ satisfies one of the KKT conditions if

$$\nabla f(\widetilde{\mathbf{x}}) - \widetilde{\mathbf{w}}\; \nabla h(\widetilde{\mathbf{x}}) = 0, \tag{1.17}$$

and because of equation 1.17, $\widetilde{\mathbf{x}}$ becomes a KKT point of problem 1.4 with the unique Lagrange multiplier $\widetilde{\mathbf{w}}$. Comparing expressions 1.16 and 1.17 we can extract the useful formula for updating the Lagrange multipliers given in equation 1.12. Compare the expression between brackets in equation 1.16 with $\widetilde{\mathbf{w}}$ in equation 1.17.

### 1.4.4 Alternating Direction Method of Multipliers

The Alternating Direction Method of Multipliers (ADMM) (Arrow et al., 1968; Glowinski and Marrocco, 1975; Gabay and Mercier, 1976) is a combination of the method of dual ascent and the method of multipliers. ADMM solves problems of the type

$$\begin{aligned} &\min\; f(\mathbf{x}) + g(\mathbf{A}\mathbf{x}) \\ &\text{subject to}\; \mathbf{x} \in C_1,\; \mathbf{A}\mathbf{x} \in C_2, \end{aligned} \tag{1.18}$$

where the functions $f : \mathbb{C}^n \to \mathbb{C}$ and $g : \mathbb{C}^m \to \mathbb{C}$ are convex, $\mathbf{A} \in \mathbb{C}^{m \times n}$ and $C_1 \subset \mathbb{C}^n, C_2 \subset \mathbb{C}^m$ are nonempty polyhedral sets (convex sets formed by a finite collection of linear

inequalities) (Bertsekas and Tsitsiklis, 1989). Performing a change of variables $\mathbf{y} = \mathbf{A}\mathbf{x}$, $\mathbf{y} \in \mathbb{C}^m$, the problem 1.18 becomes

$$\begin{aligned} &\min\ f(\mathbf{x}) + g(\mathbf{y}) \\ &\text{subject to}\ \mathbf{x} \in C_1,\ \mathbf{y} \in C_2,\ \mathbf{A}\mathbf{x} = \mathbf{y}. \end{aligned} \tag{1.19}$$

The augmented Lagrangian corresponding to problem 1.19 is

$$\mathcal{L}_{admm}(\mathbf{x}, \mathbf{y}, \mathbf{w}) = f(\mathbf{x}) + g(\mathbf{y}) - \mathbf{w}^T(\mathbf{A}\mathbf{x} - \mathbf{y}) + \frac{\beta}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2. \tag{1.20}$$

ADMM is an iterative method that minimizes one variable at a time (alternating sequentially, like the method's name), followed by an update of the Lagrange multipliers, in the following way

$$\begin{aligned} \mathbf{x}^{k+1} &= \min_{\mathbf{x}} \mathcal{L}_{admm}(\mathbf{x}, \mathbf{y}^k, \mathbf{w}^k) \\ \mathbf{y}^{k+1} &= \min_{\mathbf{y}} \mathcal{L}_{admm}(\mathbf{x}^{k+1}, \mathbf{y}, \mathbf{w}^k) \\ \mathbf{w}^{k+1} &= \mathbf{w}^k - \beta\,(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{y}^{k+1}). \end{aligned} \tag{1.21}$$

Notice that the Lagrangian now is minimized not jointly in the two variables $\mathbf{x}, \mathbf{y}$ but separately and consecutively, alleviating the separability problems of the classical method of multipliers and even making the problem easier to parallelize. Notice that in contrast to the method of multipliers, the scalar $\beta$ is not changed with the iterations. However, He et al. (2000) proposed a self-adapting technique to update $\beta$ every certain amount of iterations for problems of variational inequalities. For convergence details refer to Bertsekas and Tsitsiklis (1989).

It is interesting to notice that when either $f$ or $g$ are strictly convex, another method similar to ADMM exists denoted alternating minimization algorithm (AMA) (Tseng, 1991). The AMA algorithm follows the steps in algorithm 1.21 with the difference that the cost function to problem 1.19 is the classical Lagrangian instead of the augmented Lagrangian. The advantage of using the classical over the augmented Lagrangian is that the cost function in equation 1.20 can be sometimes difficult to minimize in the presence of the quadratic term. For the interested reader, please refer to the articles by Hestenes (1969); Powell (1969); Bertsekas and Tsitsiklis (1989); Bertsekas (1996); Nocedal and Wright (2006); Boyd et al. (2011) for more details on ADMM.

## 1.5 Classes P, NP, NP-complete and NP-hard

A (computational) *problem* is a question that needs to be answered and usually contains *parameters* with an undefined value (Garey and Johnson, 1990). The problem generally specifies the meaning of the parameters and the requirements that the answer, or *solution*, needs to fulfill for this particular problem. An alternative definition is that a problem describes the relationship between the input (data) and output (solution to the problem) (Reiter and Johnson, 2012). A problem's *instance* is a specific version of the problem for a certain choice of parameters. A *decision problem* is a problem with only two possible solutions "yes" or "no". All computational problems can be transformed into decision problems and they are analogous. The theory of P, NP and NP-complete problems generally refers to decision problems.

An *algorithm* consists of a sequence of steps for solving computational problems and it solves the problem if it always produces a solution, no matter the instance of the problem being solved (Garey and Johnson, 1990). The efficiency of an algorithm is usually determined by the time it takes to the algorithm to find a solution (which we have referred to as computational running time and also called time complexity). An algorithm is considered efficient when it requires a polynomial number of steps proportional to the input size of the problem. In other words, if $n$ is the input length, an algorithm is efficient when its running time is $\mathcal{O}(p(n))$, for some polynomial $p$. Notice that the running time does not need to be polynomial exactly, but has to be bounded by a polynomial function. But why is polynomial time considered efficient? The reason is that algorithms that are not bounded by a polynomial generally increase too fast to be considered efficient (exponential or superpolynomial algorithms). Parallelizing exponential time algorithms does not help much either in making the problem more efficient. Exponential time algorithms generally resemble exhaustive search, while polynomial time algorithms require a deeper understanding of the problem (Garey and Johnson, 1990). A problem is *tractable* (or *feasible*) if a polynomial time algorithm exists that solves the problem, and it is *intractable* otherwise. The set of problems that have polynomial time algorithms form the class P. Notice that problems considered to be intractable might become tractable when someone finds a suitable polynomial time algorithm to solve it.

The Turing machine model is a model of a computer that helps in definitions and proofs of concepts to avoid referring to a particular computer. The (deterministic) Turing machine has a large memory (infinitely large such that no algorithm can waste all its memory) and read/write capabilities to the memory tape (the tape has discrete squares). The function of the Turing machine is to find a "yes" or "no" answer for a certain input string (Reiter and Johnson, 2012). The machine operates by reading the memory one square at a time, possibly

making changes and moving left or right to continue to the next step. The Turing machine contains a set of instructions or orders that indicate what to do when reading a specific portion of the string. Schematically, the Turing machine consists of an infinitely long tape with symbols and a read/write head pointing to one of the symbols in the tape. The head is connected to a control that contains the instructions for the head (Reiter and Johnson, 2012). The Turing machine stops working (halts) when it reaches either an "accept" (yes) state or a "reject" (no) state. The formal definition for the Turing machine can be found in the books from Garey and Johnson (1990) and Reiter and Johnson (2012) and go beyond the scope of this thesis.

Nondeterministic instructions (in a Turing machine) are those that when acting on a certain symbol can result in different options. With these type of instructions, the action on a string has different possible effects and can sometimes reach the accept state, or the reject state or loop indefinitely, while acting on the same string. This is an unwanted effect and motivates the following definition. A nondeterministic Turing machine (NDTM) accepts a string only if some (or any) of its instructions lead to an "accept" state (Reiter and Johnson, 2012). This means that the NDTM will operate on a string only if it has (even one) chance of halting with the "accept" state. A deterministic Turing machine is a particular case of an NDTM where only one outcome is possible when acting on a certain symbol. It is interesting to notice that every nondeterministic Turing machine has an equivalent deterministic Turing machine, although running times for the latter are increased as a power of 2 (Theorem 5.9 in Reiter and Johnson, 2012). An NDTM can be summarized in two steps: guess and check (Tourlakis, 2012). Guess is the step where it chooses a string (it chooses correctly a string which will lead to an "accept" state) and check is the step in which it verifies (in a deterministic manner) whether that string leads to the "accept" state.

The class of nondeterministic polynomial type problems (NP) are not easy to solve as the P-problems but its solutions are easy to verify. For any "yes" instance of a problem in NP, an NDTM reaches the "accept" state in an amount of time bounded by a polynomial proportional to the size of the input (Reiter and Johnson, 2012). If otherwise the instance is a "no" instance, then the NDTM will never reach the "accept" state and it can reach the "reject" state or compute forever.

The class of NP problems is such that P is included in NP (P $\subseteq$ NP). Therefore, at least some problems in NP have polynomial time algorithms and, so far, no proof exists that all problems in NP have polynomial time algorithms (Reiter and Johnson, 2012). Also, some problems in NP do not have a polynomial solver. The question "$P = NP$?" is one of the biggest mysteries in computer science and mathematics.

A problem $P_1$ can be reduced to another problem $P_2$ through a polynomial transformation that maps all instances of $P_1$ to all instances of $P_2$. The transformation is possible only if

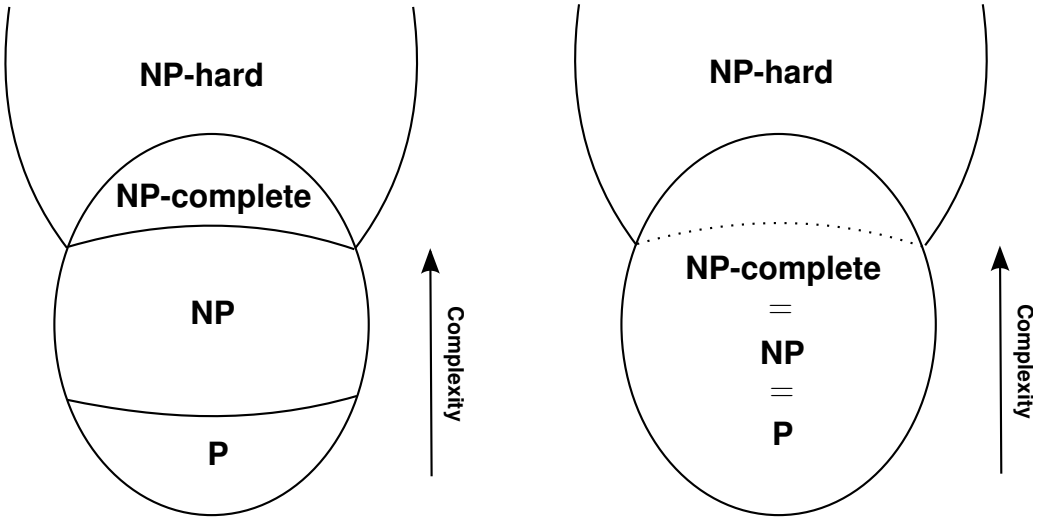
calculated in polynomial time (proportional to the size of the input). Moreover, all mapped "yes" instances of $P_1$ are "yes" instances of $P_2$, and the same with the "no" instances. Theorems prove that if $P_1$ is in P, then $P_2$ is in P. Also, if $P_1$ is in NP, then $P_2$ is in NP.

Finally, we are able to define the last two important groups classes. A problem $Z$ is NP-hard if any other problem $Y$ in NP can be polynomially reduced to $Z$. A problem $Z$ is NP-complete if it is NP-hard and it is also in NP (Reiter and Johnson, 2012; Tourlakis, 2012). While the class P is considered the collection of problems which are "easiest" to solve, the NP-complete class encompasses the "hardest" problems in NP. If we could find only one NP-complete problem that can be solved in polynomial time, then all problems that are NP-complete would be solved similarly. An interesting aspect about NP-hard problems is that they do not need to be in NP necessarily, as long as we can polynomially transform any problem in NP to them. Therefore, NP-hard problems are said to be "at least as hard" as the NP-complete problems, or "at least as hard as the hardest problem in NP" (Garey and Johnson, 1990). We also call problems that are NP-hard to be "intractable", whereas those problems in P are referred to as being "tractable". The reason why NP-hard problems are intractable is that nobody has presented yet a polynomial time algorithm to solve an NP-hard problem (although we can not discard that someday someone will) and no proof exists that NP-hard problems cannot have a polynomial time solver. Figure 1.9 contains diagrams with the P, NP, NP-complete and NP-hard classes for both the cases of P=NP and P≠NP. In these diagrams, the arrows at the right hand side point towards the groups of increasing complexity. Notice that the NP-hard group is not completely included inside NP. Additionally, the dashed line in Figure 1.9(b) indicates that P=NP does not necessarily imply that NP-hard problems can be solved in polynomial time.

## 1.6 Motivations

The motivation for this thesis is to establish tools from the field of tensor algebra in the particular application of reconstruction and random noise attenuation of seismic wavefields. The objective is to design reconstruction methods with the following properties:

- ⋆ Exploit the multidimensional characteristics of prestack seismic data.
- ⋆ Robustness to curvature of the reflections.
- ⋆ Allow large levels of sparsity and low signal-to-noise ratio.
- ⋆ Simple parameterization.




(a) P≠NP (b) P=NP

Figure 1.9: Problem classes for the two potential scenarios.

## 1.7 Contribution

The contributions from this thesis are the following:

- ⋆ We propose the use of tensor analysis in seismic processing. In particular, we focused on the problem of reconstruction of prestack seismic data and noise attenuation of incoherent noise.
- ⋆ We introduce the Higher-Order Singular Value Decomposition (HOSVD), an extension to the singular value decomposition for tensors, to perform rank reduction on the tensor representation of the seismic data for trace recovery and denoising.
- ⋆ We present a rank reduction strategy that is different from the HOSVD, based on truncating the SVD of unfolded tensors (SEQ-SVD).
- ⋆ We test novel ideas in the field of compressive sensing encompassing tensor completion as a convex optimization problem.
- ⋆ We compare all three methods proposed in this thesis with synthetic examples and a real data example.

## 1.8 Overview

**Chapter 2** gives an overview of the notation that will be used throughout this thesis, as well as definitions of vector spaces, tensor product of vector spaces and tensors. Furthermore, it covers other definitions pertaining to tensor unfoldings and tensor products. An important part of this chapter concerns with tensor decompositions and definitions of rank and $n$-rank, including the algorithms to calculate truncated decompositions.

**Chapter 3** introduces the higher-order singular value decomposition (HOSVD) reconstruction method. This method utilizes the truncated HOSVD as the rank reduction strategy and also includes a modified imputation algorithm to aid in the reconstruction of missing traces. Synthetic data examples show that this method is robust to curvature in the reflections, low signal-to-noise ratios and high levels of sparsity. The real data example from the Western Canadian Sedimentary Basin serves to assess the algorithm's behavior in a more realistic scenario.

In **Chapter 4**, we present a variation of the rank reduction strategy presented in the previous chapter. Instead of using the truncated HOSVD, we implement rank reduction through iterative hard thresholding of the singular values of the unfoldings of the tensor (an $F-X$ slice of the data to reconstruct). For seismic data with a high $SNR$, this procedure is equivalent to solving the non-convex problem of rank minimization one unfolding at a time. Despite having larger running times than HOSVD, the results obtained with this technique are very similar to the results with HOSVD and it is conceptually simpler than HOSVD. Because the method in this chapter exploits the tensor nature of the data, it has similar properties than the method in the previous chapter: it is robust to curvature and operates in scenarios of low $SNR$ and sparsely sampled data.

**Chapter 5** deals with an approach that is quite different from Chapters 3 and 4. For each frequency, the goal is to find a fully sampled low-rank tensor that fits the observations up to the level of the noise. We achieve the latter by formulating our goal as an inverse problem: find the low-rank tensor subject to the observations. This formulation has a great disadvantage, because measuring the rank of a tensor is an NP-hard and non-convex problem. Therefore, we replace the measure of the rank by the sum of the nuclear norm of the unfoldings of the tensor, turning the problem into a convex one. The alternating direction method of multipliers is the method used for the minimization of the objective function and it leads to a simple algorithm. Our tests with synthetic data indicate that the method in this chapter benefit from the same features as the methods from the previous two chapters. Our real data example is from a heavy oil field in Alberta and implied performing reconstruction on several spatial windows that cover a crossline swath.

**Chapter 6** includes the comparison of the three methods presented in this thesis (Chapters

3, 4 and 5) with two other widely known reconstruction methods: Cadzow/MSSA (Trickett et al., 2010; Oropeza and Sacchi, 2011) and minimum weighted norm interpolation (Liu, 2004). First we present the comparison on different synthetic tests with linear events (low and high $SNR$) and with curved events (low and high $SNR$), both cases with 70% of the samples randomly removed. For these examples, we evaluate both the quality of the reconstruction and the running time for all methods. The second part of this chapter concerns with a simulation of the binning parameters over field data from the Western Canadian Basin. We establish the concepts of grid population and redundancy of the grid to be those defining the suitability of a grid to perform reconstruction. In short, we consider that the reconstruction results are more successful and observed traces are better utilized when the grid contains only a few zeros and the redundancy is small. At the end of this chapter, we show the results for the reconstruction with the five methods on one of the grids that we consider suitable, according to our predefined measures.

**Chapter 7** contains the conclusions of this thesis given by our contributions, some of the limitations of the proposed methods and future work.

# CHAPTER 2

# Review on tensor algebra [1]

## 2.1 Introduction

This thesis concerns with applications of tensor algebra in the field of reconstruction and denoising of prestack seismic data. As such, this chapter focuses on establishing a foundation to understand the subsequent chapters. It is important to stress that the tensors used in this thesis are not the tensors used in quantum mechanics or in some fields of engineering (e.g. the stress tensor or the inertia tensor), often referred to as "tensor fields". In physics and engineering, a tensor is a multidimensional array that obeys certain rules under a change of basis. However, in mathematics tensors are considered to be just multidimensional arrays, an element of a space formed by the tensor product of vector spaces, while the change of basis properties arise from using the tensor product and the rule of change of basis for vectors (De Silva and Lim, 2008).

## 2.2 Notation

We will utilize the notation from the work of Kolda and Bader (2009). Scalars are denoted with italics ($a$ or $\alpha$), vectors with boldface lower-case letters ($\mathbf{a}$) and matrices with boldface upper-case letters ($\mathbf{A}$). Higher-order arrays, or tensors, are denoted with boldface Euler script letters ($\boldsymbol{\mathcal{X}}$). The $(i, j)$ element of a matrix $\mathbf{A}$ is denoted by the scalar $A_{ij}$, the element $(i, j, k)$ of a third-order tensor ($N = 3$) $\boldsymbol{\mathcal{X}}$ is denoted by $X_{ijk}$ and similarly, the element $(i, j, k, l)$ of a fourth-order tensor ($N = 4$) $\boldsymbol{\mathcal{X}}$ is denoted by $X_{ijkl}$. For practicality,

[1] Parts of this chapter have been published in Kreimer and Sacchi (2012a), Geophysics, http://dx.doi.org/10.1190/geo2011-0399.1

the indices will vary from 1 to their capital version, for instance $i = 1, \cdots, I$, unless otherwise stated. A vector space will be written with a capital letter $V$.

## 2.3 Definitions

A vector space $V$ over a certain field $\mathbb{K}$ (assuming $V$ is non-empty) is a space with "addition" ($+$) and "multiplication" ($\cdot$) operations defined obeying the following properties (Hackbusch, 2012):

$$
\begin{aligned}
&\mathbf{a} + \mathbf{b} = \mathbf{b} + \mathbf{a}, \quad \mathbf{a}, \mathbf{b} \in V \\
&(\mathbf{a} + \mathbf{b}) + \mathbf{c} = \mathbf{a} + (\mathbf{b} + \mathbf{c}), \quad \mathbf{a}, \mathbf{b}, \mathbf{c} \in V \\
&\mathbf{a} + \mathbf{0} = \mathbf{a}, \quad \mathbf{a} \in V \text{ and where } \mathbf{0} \text{ is unique in } V \\
&\mathbf{a} + (-\mathbf{a}) = \mathbf{0}, \quad \mathbf{a} \in V \text{ and where } -\mathbf{a} \text{ is unique in } V \\
&\alpha \cdot (\mathbf{a} + \mathbf{b}) = \alpha \cdot \mathbf{a} + \alpha \cdot \mathbf{b}, \quad \mathbf{a}, \mathbf{b} \in V \text{ and } \alpha \in \mathbb{K} \\
&(\alpha + \beta) \cdot \mathbf{a} = \alpha \cdot \mathbf{a} + \beta \cdot \mathbf{a}, \quad \mathbf{a} \in V \text{ and } \alpha, \beta \in \mathbb{K} \\
&\alpha \cdot (\beta \cdot \mathbf{a}) = (\alpha\beta) \cdot \mathbf{a}, \quad \mathbf{a} \in V \text{ and } \alpha, \beta \in \mathbb{K} \\
&1 \cdot \mathbf{a} = \mathbf{a}, \quad \mathbf{a} \in V \text{ and } 1 \text{ is the multiplicative unity in } \mathbb{K} \\
&0 \cdot \mathbf{a} = \mathbf{0}, \quad \mathbf{a} \in V, 0 \text{ is the additive unity in } \mathbb{K} \text{ and } \mathbf{0} \text{ the zero element of } V.
\end{aligned} \tag{2.1}
$$

In this thesis, $\mathbb{K} = \mathbb{C}$ and $V = \mathbb{C}^n$. The tensor product of two vector spaces is defined by a vector space and a bilinear map. If $V, W, X$ are three vector spaces, a bilinear map from $V \times W \to X$ is a function $H : V \times W \to X$ that satisfies

$$
\begin{aligned}
H(\alpha v_1 + v_2, w) &= \alpha H(v_1, w) + H(v_2, w), \quad v_1, v_2 \in V, w \in W, \alpha \in \mathbb{K} \\
H(v, \alpha w_1 + w_2) &= \alpha H(v, w_1) + H(v, w_2), \quad v \in V, w_1, w_2 \in W, \alpha \in \mathbb{K}.
\end{aligned} \tag{2.2}
$$

The tensor product of two vector spaces $V, W$ is also a vector space denoted $V \otimes W$ associated with a bilinear map $\eta : V \times W \to V \otimes W$ with the property that for every other vector space $X$ and every bilinear map $H : V \times W \to X$, it exists a linear mapping $T : V \otimes W \to X$ and $H = T \circ \eta$, where $\circ$ denotes the composition of two functions (Kamnitzer, 2011). The elements of a tensor product of two vector spaces are matrices. The definition can be extrapolated to products of $N$ vector spaces. A tensor of order $N$ is an element of the tensor product of $N$ vector spaces. Notice that each vector space can have its own coordinate system. Addition and scalar multiplication will be defined as point to point operations. According to De Silva and Lim (2008), it can be considered that $\mathbb{C}^{I_1} \otimes \mathbb{C}^{I_2} \otimes \cdots \otimes \mathbb{C}^{I_N} \cong \mathbb{C}^{I_1 \times I_2 \times \cdots \times I_N}$

and therefore it is equivalent to define a tensor by a $N$th-order array or as a sum of tensor product of $N$ vectors.

The order or dimensionality of a tensor is denoted $N$ (dimension is also referred to as mode). For instance, a tensor of order $N = 1$ is a vector and a tensor of order $N = 2$ is a matrix. The definitions that follow will be based on tensors of order four, which we will use throughout the following chapters. The slices of tensor $\boldsymbol{\mathcal{X}}$ are obtained by fixing all coordinates but two. For example, $\boldsymbol{\mathcal{X}}(i, j, :, :)$ and $\boldsymbol{\mathcal{X}}(:, :, k, l)$ are slices of the tensor $\boldsymbol{\mathcal{X}}$. A *fiber* is obtained by fixing every index of the tensor except one. By convention, fibers are column vectors. For example, $\boldsymbol{\mathcal{X}}(:, j, k, l)$, $\boldsymbol{\mathcal{X}}(i, :, k, l)$, $\boldsymbol{\mathcal{X}}(i, j, :, l)$ and $\boldsymbol{\mathcal{X}}(i, j, k, :)$ are mode-1, mode-2, mode-3 and mode-4 fibers, respectively, from a fourth-order tensor $\boldsymbol{\mathcal{X}}$.

### 2.3.1 Unfoldings

A tensor of order $N$ can be unfolded or flattened in $N$ different ways. The flattening of a tensor is in essence the reordering of its elements into a matrix. There are various ways of unfolding a tensor. In Chapter 3 we adopt the $n$-mode flattening described by Bader and Kolda (2006) and Kolda (2006). The latter is the convention used by the HOSVD algorithm in the Tensor Toolbox described in the article from Bader and Kolda (2010). In Chapters 4 and 5 we will use a different convention.

A tensor of order $N = 3$ has three modes $n = 1, 2, 3$, whereas a tensor of order $N = 4$ has four modes $n = 1, 2, 3, 4$. The $n$-mode fibers are the columns of the $n$-mode unfolded matrix. The flattening of a tensor $\boldsymbol{\mathcal{X}}$ in its $n$-mode is symbolized by the matrix $\mathbf{X}^{(n)}$. Figure 2.1 exemplifies the $n$-mode flattening of a tensor of order three. For example, let $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{4\times3\times2}$ be a third-order tensor defined by its two frontal slices

$$\boldsymbol{\mathcal{X}}(:, :, 1) = \begin{pmatrix} X_{111} & X_{121} & X_{131} \\ X_{211} & X_{221} & X_{231} \\ X_{311} & X_{321} & X_{331} \\ X_{411} & X_{421} & X_{431} \end{pmatrix}, \qquad \boldsymbol{\mathcal{X}}(:, :, 2) = \begin{pmatrix} X_{112} & X_{122} & X_{132} \\ X_{212} & X_{222} & X_{232} \\ X_{312} & X_{322} & X_{332} \\ X_{412} & X_{422} & X_{432} \end{pmatrix}.$$

The three ways one can unfold this tensor are given by

$$\mathbf{X}^{(1)} = \begin{pmatrix} X_{111} & X_{121} & X_{131} & X_{112} & X_{122} & X_{132} \\ X_{211} & X_{221} & X_{231} & X_{212} & X_{222} & X_{232} \\ X_{311} & X_{321} & X_{331} & X_{312} & X_{322} & X_{332} \\ X_{411} & X_{421} & X_{431} & X_{412} & X_{422} & X_{432} \end{pmatrix},$$

$$\mathbf{X}^{(2)} = \begin{pmatrix} X_{111} & X_{211} & X_{311} & X_{411} & X_{112} & X_{212} & X_{312} & X_{412} \\ X_{121} & X_{221} & X_{321} & X_{421} & X_{122} & X_{222} & X_{322} & X_{422} \\ X_{131} & X_{231} & X_{331} & X_{431} & X_{132} & X_{232} & X_{332} & X_{432} \end{pmatrix},$$

$$\mathbf{X}^{(3)} = \begin{pmatrix} X_{111} & X_{211} & X_{311} & X_{411} & X_{121} & X_{221} & X_{321} & X_{421} & X_{131} & X_{231} & X_{331} & X_{431} \\ X_{112} & X_{212} & X_{312} & X_{412} & X_{122} & X_{222} & X_{322} & X_{422} & X_{132} & X_{232} & X_{332} & X_{432} \end{pmatrix}.$$

The numbering of each unfolding varies according to the author and is irrelevant as long as there is consistency in the calculations. In general, mapping from a tensor to a matrix requires knowing the size of the tensor, the dimension (or mode) that we want to map to the rows of the matrix and which mode we map to the columns of the matrix (Kolda, 2006). In mathematical terms, given a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times \cdots \times I_N}$, the mode-$n$ unfolding is such that the element $(i_1, i_2, \cdots, i_N)$ maps to a matrix element $(i_n, j)$ given by the expression (De Lathauwer et al., 2000a; Kolda and Bader, 2009)

$$j = 1 + \sum_{\substack{k=1 \\ k \neq n}}^{N} (i_k - 1)\, J_k, \qquad J_k = \prod_{\substack{m=1 \\ m \neq n}}^{k-1} I_m \,. \tag{2.3}$$

### 2.3.2 Products

The product between a tensor and a matrix can be expressed by using the concept of $n$-mode product. In the case of a fourth-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$, the $n$-mode product of this tensor with a matrix $\mathbf{B} \in \mathbb{C}^{J \times I_n}$ with $n = 1, 2, 3, 4$ is denoted by $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} \times_n \mathbf{B}$. The four products are

$$\begin{aligned}
(\boldsymbol{\mathcal{X}} \times_1 \mathbf{B})_{j i_2 i_3 i_4} &= \sum_{i_1=1}^{I_1} X_{i_1 i_2 i_3 i_4} B_{j i_1}, \quad \text{with} \quad \mathbf{B} \in \mathbb{C}^{J \times I_1} \\
(\boldsymbol{\mathcal{X}} \times_2 \mathbf{B})_{i_1 j i_3 i_4} &= \sum_{i_2=1}^{I_2} X_{i_1 i_2 i_3 i_4} B_{j i_2}, \quad \text{with} \quad \mathbf{B} \in \mathbb{C}^{J \times I_2} \\
(\boldsymbol{\mathcal{X}} \times_3 \mathbf{B})_{i_1 i_2 j i_4} &= \sum_{i_3=1}^{I_3} X_{i_1 i_2 i_3 i_4} B_{j i_3}, \quad \text{with} \quad \mathbf{B} \in \mathbb{C}^{J \times I_3} \\
(\boldsymbol{\mathcal{X}} \times_4 \mathbf{B})_{i_1 i_2 i_3 j} &= \sum_{i_4=1}^{I_4} X_{i_1 i_2 i_3 i_4} B_{j i_4}, \quad \text{with} \quad \mathbf{B} \in \mathbb{C}^{J \times I_4}.
\end{aligned} \tag{2.4}$$

The *Kronecker product* of two matrices $\mathbf{A}$ and $\mathbf{B}$ of sizes $m \times n$ and $p \times q$ is another matrix

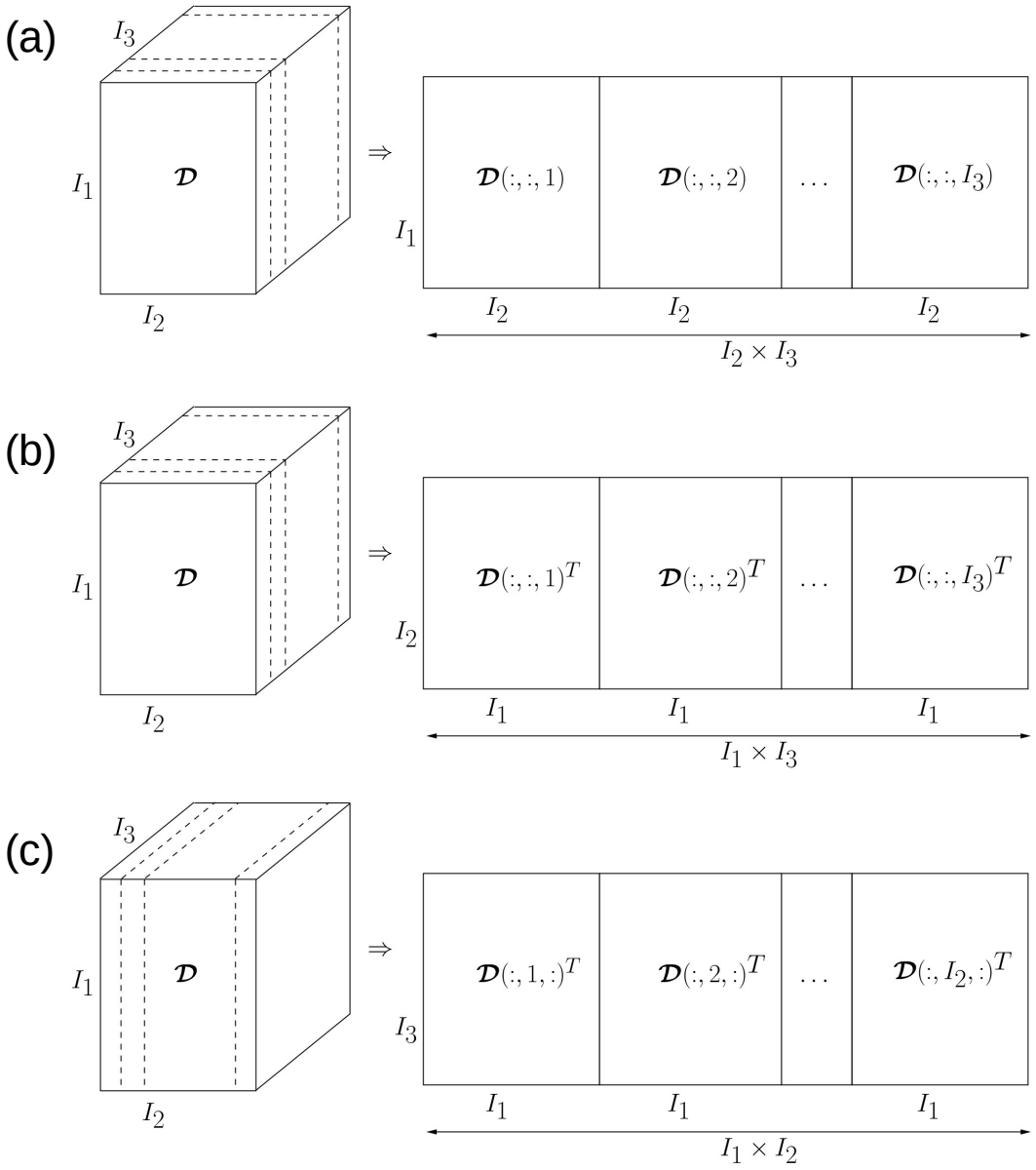



Figure 2.1: Unfolding or flattening of a third-order tensor $\boldsymbol{\mathcal{X}}$. (a) 1st mode, (b) 2nd mode, and (c) 3rd mode. We followed the unfolding convention provided in the Tensor Toolbox (Bader and Kolda, 2010).

of size $mp \times nq$

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} A_{11}\mathbf{B} & A_{12}\mathbf{B} & \cdots & A_{1n}\mathbf{B} \\ A_{21}\mathbf{B} & A_{22}\mathbf{B} & \cdots & A_{2n}\mathbf{B} \\ \vdots & \vdots & \vdots\cdots & \vdots \\ A_{m1}\mathbf{B} & A_{m2}\mathbf{B} & \cdots & A_{mn}\mathbf{B} \end{pmatrix} \tag{2.5}$$

$$= [\mathbf{a}_1 \otimes \mathbf{b}_1 \quad \mathbf{a}_1 \otimes \mathbf{b}_2 \quad \mathbf{a}_1 \otimes \mathbf{b}_3 \quad \cdots \quad \mathbf{a}_n \otimes \mathbf{b}_{q-1} \quad \mathbf{a}_n \otimes \mathbf{b}_q], \tag{2.6}$$

where $\mathbf{a}_i, \mathbf{b}_i$ are the columns of matrices $\mathbf{A}, \mathbf{B}$, respectively.

The Kronecker product of two vectors $\mathbf{x}$ and $\mathbf{y}$ of length $m$ and $n$ (Laub, 2004), respectively, is

$$\begin{aligned} \mathbf{x} \otimes \mathbf{y} &= [x_1\mathbf{y}^{\mathrm{T}} \cdots x_m\mathbf{y}^{\mathrm{T}}]^{\mathrm{T}} \\ &= [x_1y_1 \cdots x_1y_n \; x_2y_1 \; \cdots x_ny_{m-1} \; x_ny_m]^{\mathrm{T}}. \end{aligned} \tag{2.7}$$

Some properties that will prove useful are:

$$(\mathbf{A} \otimes \mathbf{B})^{\mathrm{T}} = \mathbf{A}^{\mathrm{T}} \otimes \mathbf{B}^{\mathrm{T}}, \tag{2.8}$$

$$\operatorname{rank}(\mathbf{A}^{\mathrm{T}}) = \operatorname{rank}(\mathbf{A}), \tag{2.9}$$

$$\operatorname{rank}(\mathbf{A} \otimes \mathbf{B}) = \operatorname{rank}(\mathbf{A}) \operatorname{rank}(\mathbf{B}), \tag{2.10}$$

$$\operatorname{rank}(\mathbf{A}\,\mathbf{B}) \leq \min(\operatorname{rank}(\mathbf{A}), \operatorname{rank}(\mathbf{B})). \tag{2.11}$$

The *Khatri-Rao* product between two matrices $\mathbf{A}$ of size $m \times n$ and $\mathbf{B}$ of size $p \times n$ is a column-wise Kronecker product and gives a matrix of size $mp \times n$

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{a}_1 \otimes \mathbf{b}_1 \quad \mathbf{a}_2 \otimes \mathbf{b}_2 \quad \cdots \quad \mathbf{a}_n \otimes \mathbf{b}_n]. \tag{2.12}$$

Particularly, if $\mathbf{A} = \mathbf{a}$ and $\mathbf{B} = \mathbf{b}$, i.e. if the product is among two vectors, the Khatri Rao product is equivalent to the Kronecker one $\mathbf{a} \odot \mathbf{b} = \mathbf{a} \otimes \mathbf{b}$.

The *Hadamard* product between two matrices $\mathbf{A}, \mathbf{B}$ of size $m \times n$ is an element-wise product symbolized by "$\cdot$" such that

$$\mathbf{A} \cdot \mathbf{B} = \begin{pmatrix} A_{11}B_{11} & A_{12}B_{12} & \cdots & A_{1n}B_{1n} \\ A_{21}B_{21} & A_{22}B_{22} & \cdots & A_{2n}B_{2n} \\ \vdots & \vdots & \vdots\cdots & \vdots \\ A_{m1}B_{m1} & A_{m2}B_{m2} & \cdots & A_{mn}B_{mn} \end{pmatrix}. \tag{2.13}$$

The inner product of two tensors $\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{W}}$ is the sum of the products of their entries (element to element product), which for two fourth-order tensors with elements in $\mathbb{C}$ is

$$< \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{W}} >= \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \sum_{i_3=1}^{I_3} \sum_{i_4=1}^{I_4} X_{i_1 i_2 i_3 i_4} Y^*_{i_1 i_2 i_3 i_4}, \tag{2.14}$$

where $^*$ indicates the complex conjugate of element $Y_{ijkl}$. Like other inner product spaces (vector spaces associated with a certain norm), $< \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{X}} >= \|\boldsymbol{\mathcal{X}}\|_F^2$. The Frobenius norm of a matrix $\mathbf{A} \in \mathbb{C}^{I \times J}$ is defined as the square root of the sum of the squared absolute values of the elements of the matrix $\|\mathbf{A}\|_F^2 = \sum_{i,j=1}^{I,J} |A_{ij}|^2$. Similarly, the Frobenius norm of a tensor of order $N = 4$, $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$ is defined as

$$\|\boldsymbol{\mathcal{X}}\|_F^2 = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \sum_{i_3=1}^{I_3} \sum_{i_4=1}^{I_4} |X_{i_1 i_2 i_3 i_4}|^2. \tag{2.15}$$

Clearly, one can extend these definitions to tensors of order $N > 4$.

## 2.4 Tensor decompositions and tensor rank

A tensor of order four $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$ is said to have rank one if it can be expressed as

$$\boldsymbol{\mathcal{X}} = \mathbf{u}^{(1)} \circ \mathbf{u}^{(2)} \circ \mathbf{u}^{(3)} \circ \mathbf{u}^{(4)}, \quad \text{such that} \tag{2.16}$$

$$X_{ijkl} = u_i^{(1)} u_j^{(2)} u_k^{(3)} u_l^{(4)}, \tag{2.17}$$

where $\mathbf{u}^{(i)}$, $i = 1, \cdots, 4$ are vectors and $\circ$ is the outer product between vectors (Figure 2.2). If the length of the vectors are $I_1, I_2, I_3$ and $I_4$ for $\mathbf{u}^{(1)}, \mathbf{u}^{(2)}, \mathbf{u}^{(3)}$ and $\mathbf{u}^{(4)}$, respectively, the size of $\boldsymbol{\mathcal{X}}$ is $I_1 \times I_2 \times I_3 \times I_4$. For a second-order tensor $\mathbf{X}$, i.e. a matrix, this definition reduces to $\mathbf{X} = \mathbf{u}^{(1)} {\mathbf{u}^{(2)}}^{\mathrm{H}}$.

A matrix can be approximated by another of lower rank via the Singular Value Decomposition (Eckart and Young, 1936). This is a familiar concept in linear algebra. For instance, a matrix $\mathbf{A}$ of rank $r$ can be approximated by one of lower rank $\widetilde{\mathbf{A}} = \mathbf{U}_s \boldsymbol{\Sigma}_s \mathbf{V}_s^{\mathrm{H}}$, $s < r$, where $\mathbf{U}_s$ and $\mathbf{V}_s$ are orthogonal matrices with $s$ columns and $\boldsymbol{\Sigma}_s$ of size $s \times s$ is the associated diagonal matrix of the $s$ largest singular values of $\mathbf{A}$ (the singular values are the square roots of the eigenvalues of $\mathbf{A}\mathbf{A}^{\mathrm{H}}$ and $\mathbf{A}^{\mathrm{H}}\mathbf{A}$). The matrix $\widetilde{\mathbf{A}}$ is the best rank-$s$ approximation of $\mathbf{A}$ with respect to the Frobenious norm. In many signal processing problems one wants to approximate a matrix by one of lower rank. Examples of the latter are methods for eigenimage-based denoising (Freire and Ulrych, 1988), Cadzow filtering (Cadzow, 1988) and wave extrapolation (Fomel et al., 2010). The SVD is our preferred rank reduction tool

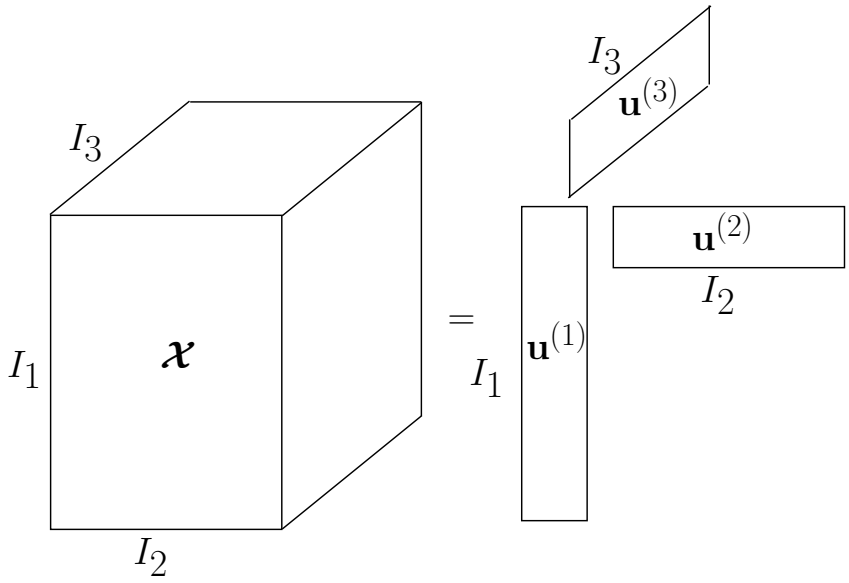



Figure 2.2: Third-order tensor $\boldsymbol{\mathcal{X}}$ of rank 1.

when we have a data structure that can be expressed in matrix form. However, there are applications where one wishes to consider multidimensional data structures saved in tensorial form. The basic question is if a tensor can also be approximated by one of lower rank, in a similar manner the SVD can be used to approximate a matrix by one of lower rank (Omberg et al., 2007). The SVD can be generalized to higher-order tensors via the CANDECOMP/PARAFAC (Hitchcock, 1927; Carroll and Chang, 1970; Harshman, 1970) decomposition and the Tucker/Higher-Order Singular Value Decomposition (Tucker, 1963; De Lathauwer et al., 2000a,b).

### 2.4.1 CP decomposition

A precursor of the CP decomposition was first proposed by Hitchcock (1927) and followed later by Cattell (1944). However, the CP decomposition became more popular mostly in the psychometrics community with the works of Carroll and Chang (1970), who called it canonical decomposition (CANDECOMP), and Harshman (1970), who suggested parallel factors (PARAFAC). This decomposition is commonly known as CP, as an abbreviation of both names (Kiers, 2000). The CP decomposition expresses a tensor as a sum of rank one

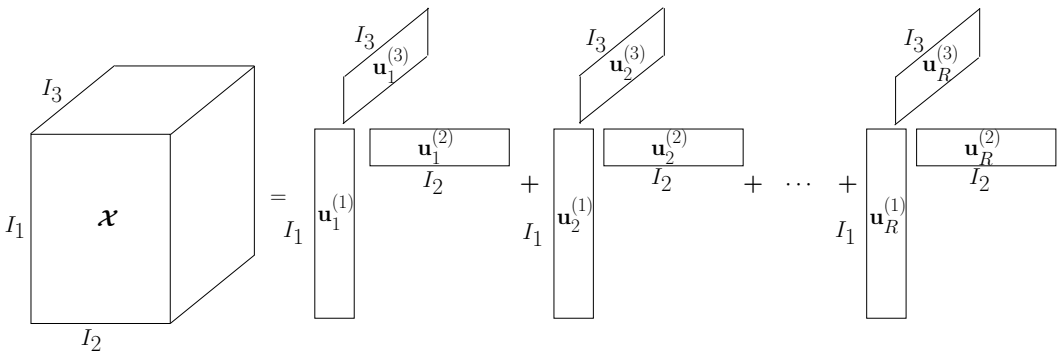



Figure 2.3: CP decomposition of a third-order tensor $\boldsymbol{\mathcal{X}}$.

tensors (Figure 2.3). A fourth-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$ can be approximated as

$$\begin{aligned} \boldsymbol{\mathcal{X}} &\approx \sum_{m=1}^{R} \mathbf{u}_m^{(1)} \circ \mathbf{u}_m^{(2)} \circ \mathbf{u}_m^{(3)} \circ \mathbf{u}_m^{(4)}, \\ X_{ijkl} &\approx \sum_{m=1}^{R} U_{im}^{(1)} U_{jm}^{(2)} U_{km}^{(3)} U_{lm}^{(4)}, \end{aligned} \tag{2.18}$$

where $\mathbf{u}_m^{(j)} \in \mathbb{C}^{I_j}, m = 1, \cdots, R, j = 1, 2, 3, 4$ are vectors and $R \in \mathbb{N}^+$. It is a common practice to make the columns $\mathbf{u}_m^{(j)}, m = 1, \cdots, R$ (for each $j$) normalized and capture the normalization constants in a vector of length $R$ that is added in the previous expression after the summation sign (Kolda and Bader, 2009).

If the approximation sign in equation 2.18 is an equal sign, the integer $R = \text{rank}(\boldsymbol{\mathcal{X}})$. Namely, the rank of a tensor $\boldsymbol{\mathcal{X}}$ is the smallest amount of rank-one tensors added together needed to synthesize that tensor (Hitchcock, 1927; Kruskal, 1977). For tensors, there is not a unique concept of rank. Although the tensor rank is defined as the matrix rank, a particularity of the rank of a tensor is that there is not a straightforward way of calculating it, it is an NP-hard problem (refer to section 1.5 in Chapter 1). For example, the rank of a tensor can be different over $\mathbb{R}$ or $\mathbb{C}$ (Kruskal, 1983) or sometimes the rank cannot even be determined (Kruskal, 1989). These difficulties give rise to other definitions of tensor rank involving statistics. For an extensive definition of tensor ranks and its calculation refer to Comon et al. (2009) and Kolda and Bader (2009).

When the decomposition in equation 2.18 is applied to matrices, the decomposition is not unique while SVD is unique because of the orthogonality conditions of $\mathbf{U}, \mathbf{V}$ and the ordering of the elements in the diagonal matrix $\boldsymbol{\Sigma}$. On the other hand, the CP decomposition is unique when applied to tensors of order greater then two, up to a permutation of the components

and scaling (Kruskal, 1977, 1989; Sidiropoulos and Bro, 2000).

The CP decomposition can be used for finding a low-rank approximation of a tensor. The SVD of a matrix $\mathbf{A}$ of rank $R$ is expressed as

$$\mathbf{A} = \mathbf{U}_R \, \mathbf{\Sigma}_R \, \mathbf{V}_R^{\mathrm{H}} = \sum_{m=1}^{R} \sigma_i \, \mathbf{u}_m \circ \mathbf{v}_m, \tag{2.19}$$

with $\sigma_i$ ordered in decreasing magnitude. As we specified before in the previous section, the best $R$-rank approximation is achieved by truncating the SVD of $\mathbf{A}$. As the CP decomposition and the SVD decomposition seem very similar, we would like to achieve the best $R$-rank approximation of a tensor in the same way. In the SVD, the low-rank approximation is found sequentially; i.e. the $R+1$-rank approximation and the $R$-rank approximation share the same information up to the $R$-component. However, this is not the case for tensors and the $R, R+1$ rank approximations can be very different (Kolda, 2001). Furthermore, the best rank-$R$ approximation might not exist, a problem called degeneracy.

The computation of the CP decomposition requires knowing the amount $R$ of rank-one tensors to involve in the sum (equation 2.18). Most algorithms calculate many alternative decompositions to find the one that fits the best. Nevertheless, this fit criterion can fail when the data tensor contains noise (as it is in most cases). One of the most used and robust algorithms is the alternating least squares (ALS) algorithm (Carroll and Chang, 1970; Harshman, 1970). This method calculates $R$ components (with $R$ given by the user) as in equation 2.18 that, when combined together, give the best fit to the original data in terms of the Frobenius norm. Let us group all $\mathbf{u}_m^{(j)}, m = 1, \cdots, R$ in matrices $\mathbf{U}^{(j)}, j = 1, 2, 3, 4$. The ALS algorithm operates by fixing all matrices $\mathbf{U}^{(j)}$ but one (reducing the problem to just one variable) and solving the least squares problem of finding that matrix which offers the best fit. After finding the best fit matrix, it continues with the other matrices which were fixed before and so on until achieving convergence (defined beforehand). Kolda and Bader (2009) present the ALS algorithm for a tensor of order $N$. Naturally, the ALS algorithm can be also used to compute a rank-reduced version of $\boldsymbol{\mathcal{X}}$ by specifying an appropriate rank value (smaller than $R$). Despite being the most widely used method for finding this decomposition, ALS can be slow to converge and can reach a local minimum or plateau (if it reaches it at all). Several authors have proposed improvements to ALS (Rajih and Comon, 2005; Tomasi, 2006; Tomasi and Bro, 2006; Navasca et al., 2008).

### 2.4.2 Tucker/HOSVD decomposition

The Tucker decomposition owes its existence to L.R. Tucker, who presented it in his article in 1963 (Tucker, 1963) and subsequent articles by him and others (Levin, 1963; Tucker,

1964, 1966). As for CP, the Tucker decomposition is considered a higher-order extension of the SVD and principal component analysis (PCA). For a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$, the Tucker decomposition expresses it as

$$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{G}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \times_4 \mathbf{U}^{(4)} = \sum_{i=1}^{I_1} \sum_{j=1}^{I_2} \sum_{k=1}^{I_3} \sum_{l=1}^{I_4} G_{ijkl}\, \mathbf{u}_i^{(1)} \circ \mathbf{u}_j^{(2)} \circ \mathbf{u}_k^{(3)} \circ \mathbf{u}_l^{(4)}$$

$$X_{mnop} = \sum_{i=1}^{I_1} \sum_{j=1}^{I_2} \sum_{k=1}^{I_3} \sum_{l=1}^{I_4} G_{ijkl}\, U_{mi}^{(1)} U_{nj}^{(2)} U_{ok}^{(3)} U_{pl}^{(4)}, \quad (2.20)$$

where $\boldsymbol{\mathcal{G}}$ is called the core-tensor, it has the same size as $\boldsymbol{\mathcal{X}}$ and measures the connection between the matrices $\mathbf{U}^{(j)}$. The matrices $\mathbf{U}^{(j)}, j = 1, 2, 3, 4$ are often orthogonal, they have sizes $I_j \times I_j$ and contain information about its respective component $j$. It is interesting to notice that the CP decomposition is a special case of the Tucker decomposition when the core-tensor is superdiagonal ($G_{ijkl} \neq 0$ only when $i = j = k = l$) and $I_1 = I_2 = I_3 = I_4$. Other variations of the Tucker decomposition exist, such as Tucker-1 and Tucker-2 (Tucker, 1966).

De Lathauwer et al. (2000a) formalized the Tucker decomposition as a higher-order extension of the SVD and called this decomposition Higher-Order SVD (HOSVD). The HOSVD theorem states that every tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$ can be expressed as in equation 2.20 with the following properties:

- The matrices $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, $\mathbf{U}^{(3)}$ and $\mathbf{U}^{(4)}$ are the unitary matrices containing the left singular vectors of the four matrices that one can obtain by flattening the tensor $\boldsymbol{\mathcal{X}}$ (Kolda and Bader, 2009). In other words, $\mathbf{U}^{(n)}$ is obtained via the SVD of $\mathbf{X}^{(n)}$, the $n$-mode matrix unfolding of the tensor $\boldsymbol{\mathcal{X}}$

  $$\mathbf{X}^{(n)} = \mathbf{U}^{(n)} \boldsymbol{\Sigma}^{(n)} {\mathbf{V}^{(n)}}^{\mathrm{H}}, \quad (2.21)$$

  where $\mathbf{U}^{(n)}$ and $\mathbf{V}^{(n)}$ are the left and right side matrices of singular vectors, respectively. The matrix $\boldsymbol{\Sigma}^{(n)}$ represents the diagonal matrix containing the singular values of $\mathbf{X}^{(n)}$.

- The core-tensor $\boldsymbol{\mathcal{G}} \in \mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$ satisfies: the subtensors

  $$\boldsymbol{\mathcal{G}}(\alpha, :, :, :) \text{ and } \boldsymbol{\mathcal{G}}(\beta, :, :, :),\ 1 \leq \alpha, \beta \leq I_1 \quad (2.22)$$

  (formed by fixing one index of the 1st coordinate) are orthogonal to each other with the definition of inner product as given in equation 2.14 and the same for all subtensors obtained when fixing the second, third and fourth coordinates. Furthermore, the Frobenius norm of these subtensors do not increase as their respective indices increase.

Since $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, $\mathbf{U}^{(3)}$ and $\mathbf{U}^{(4)}$ are orthogonal, the core-tensor $\boldsymbol{\mathcal{G}}$ can be easily estimated via the following expression (Omberg et al., 2007)

$$\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{U}^{(1)^{\mathrm{H}}} \times_2 \mathbf{U}^{(2)^{\mathrm{H}}} \times_3 \mathbf{U}^{(3)^{\mathrm{H}}} \times_4 \mathbf{U}^{(4)^{\mathrm{H}}}, \tag{2.23}$$

where $\mathbf{U}^{(i)^{\mathrm{H}}}$ denotes the complex conjugate transpose of $\mathbf{U}^{(i)}$. For a proof of this property we refer the reader to Theorem 4.1 in De Lathauwer et al. (2000b). The core-tensor $\boldsymbol{\mathcal{G}}$ plays a role similar to that of the matrix of singular values $\boldsymbol{\Sigma}$ in the SVD. In fact, one can reduce the rank of the tensor by truncating the core tensor.

The $n$-rank of a tensor $\boldsymbol{\mathcal{X}}$ of size $I_1 \times I_2 \times I_3 \times I_4$ is defined as the rank of its mode-$n$ unfolding (the column rank of $\mathbf{X}^{(n)}$). Assuming that $\text{rank}(\mathbf{X}^{(j)}) = R_j, j = 1, 2, 3, 4$, then we say that $\boldsymbol{\mathcal{X}}$ has rank-$(R_1, R_2, R_3, R_4)$. Notice that the $n$-rank is a multi-rank definition, as we are looking for the rank of each of its matrix unfoldings, which is very different from the tensor rank definition we gave in section 2.4.1. Naturally, $n$-rank $= R_n \leq I_n$ and $R_n$ are not necessarily the same for all $n$. Furthermore, even if all $n$-ranks are equal they will always obey $R_n \leq R$, where $R =$ rank $(\boldsymbol{\mathcal{X}})$.

Unlike the CP decomposition, the Tucker decomposition is not unique (Kolda and Bader, 2009). However, the orthogonality conditions in the core-tensor given in the HOSVD cause uniqueness in the $n$-mode singular values (the singular values of $\mathbf{X}^{(n)}$) and uniqueness in the factor matrices $\mathbf{U}^{(n)}$ up to the product with a particular kind of block-diagonal matrix (Property 4 in De Lathauwer et al., 2000a). An interesting property of the HOSVD is that when used for matrices (tensors of order two), the HOSVD reduces to the classical SVD (refer to Properties 5 and 12 in De Lathauwer et al., 2000a).

The "truncated" HOSVD consists of the representation of the fourth-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$ by the product of four unitary matrices $\widetilde{\mathbf{U}}^{(1)}$, $\widetilde{\mathbf{U}}^{(2)}$, $\widetilde{\mathbf{U}}^{(3)}$ and $\widetilde{\mathbf{U}}^{(4)}$ of sizes $I_j \times R_j, R_j < I_j$, respectively, and a small core-tensor $\widetilde{\boldsymbol{\mathcal{G}}} \in \mathbb{C}^{R_1 \times R_2 \times R_3 \times R_4}$

$$\boldsymbol{\mathcal{X}} \approx \widetilde{\boldsymbol{\mathcal{X}}} = \widetilde{\boldsymbol{\mathcal{G}}} \times_1 \widetilde{\mathbf{U}}^{(1)} \times_2 \widetilde{\mathbf{U}}^{(2)} \times_3 \widetilde{\mathbf{U}}^{(3)} \times_4 \widetilde{\mathbf{U}}^{(4)}. \tag{2.24}$$

We denote $\widetilde{\boldsymbol{\mathcal{X}}}$ the rank-reduced approximation of tensor $\boldsymbol{\mathcal{X}}$. Clearly, the matrices $\widetilde{\mathbf{U}}^{(n)}$, $n = 1, 2, 3, 4$ in equation 2.24 contain the first $R_n$ left singular vectors of the unfolded matrix $\mathbf{X}^{(n)}$.

The decomposition of equation 2.24 is exact if $\widetilde{\boldsymbol{\mathcal{G}}}$ is a rank-$(I_1, I_2, I_3, I_4)$ tensor (equivalent to equation 2.20). On the other hand, if the decomposition is truncated by reducing the size of the core tensor $(R_1, R_2, R_3, R_4)$, with $R_j < I_j$ for at least one $j$, the equation 2.24 provides the desired low-rank approximation to the tensor $\boldsymbol{\mathcal{X}}$ (De Lathauwer et al., 2000a). The truncated HOSVD of equation 2.24 is not an optimal approximation in terms of the Frobenius norm of the difference $\|\boldsymbol{\mathcal{X}} - \widetilde{\boldsymbol{\mathcal{X}}}\|_F^2$. Furthermore, the error in the Frobenius norm

when truncating the HOSVD is

$$\|\boldsymbol{\mathcal{X}} - \widetilde{\boldsymbol{\mathcal{X}}}\|_F^2 \leq \sum_{i_1=R_1+1}^{I_1} {\sigma_{i_1}^{(1)}}^2 + \sum_{i_2=R_2+1}^{I_2} {\sigma_{i_2}^{(2)}}^2 + \sum_{i_3=R_3+1}^{I_3} {\sigma_{i_3}^{(3)}}^2 + \sum_{i_4=R_4+1}^{I_4} {\sigma_{i_4}^{(4)}}^2, \tag{2.25}$$

where $\sigma_{i_j}^{(j)}, i_j = R_j + 1, \ldots, I_j$, are the smallest $n$-mode singular values in the matrix unfolding $\mathbf{D}^{(j)}$. The error bound is composed of the sum of the squared $I_j - R_j$ remaining singular values along each mode, which is an extension of the error expression for the SVD (De Lathauwer et al., 2000a).

The truncated HOSVD is taken as the first approximation of an iterative ALS algorithm (as for the computation of the CP decomposition). The matrices and core-tensor are updated iteratively starting with equation 2.24. The problem to solve is

$$\begin{aligned} \min_{\widetilde{\boldsymbol{\mathcal{G}}}, \widetilde{\mathbf{U}}^{(1)}, \widetilde{\mathbf{U}}^{(2)}, \widetilde{\mathbf{U}}^{(3)}, \widetilde{\mathbf{U}}^{(4)}} & \|\boldsymbol{\mathcal{X}} - \widetilde{\boldsymbol{\mathcal{X}}}\|_F = \left\| \boldsymbol{\mathcal{X}} - \widetilde{\boldsymbol{\mathcal{G}}} \times_1 \widetilde{\mathbf{U}}^{(1)} \times_2 \widetilde{\mathbf{U}}^{(2)} \times_3 \widetilde{\mathbf{U}}^{(3)} \times_4 \widetilde{\mathbf{U}}^{(4)} \right\|_F \\ & \text{such that } \ \widetilde{\boldsymbol{\mathcal{G}}} \in \mathbb{C}^{R_1 \times R_2 \times R_3 \times R_4} \\ & \text{and } \ \widetilde{\mathbf{U}}^{(n)} \in \mathbb{C}^{I_n \times R_n} \text{ are unitary, } n = 1, 2, 3, 4. \end{aligned} \tag{2.26}$$

The square of the cost function to minimize in equation 2.26 can be rewritten as

$$\|\boldsymbol{\mathcal{X}}\|_F^2 - \left\| \boldsymbol{\mathcal{X}} \times_1 \widetilde{\mathbf{U}}^{(1)^{\mathrm{H}}} \times_2 \widetilde{\mathbf{U}}^{(2)^{\mathrm{H}}} \times_3 \widetilde{\mathbf{U}}^{(3)^{\mathrm{H}}} \times_4 \widetilde{\mathbf{U}}^{(4)^{\mathrm{H}}} \right\|_F^2, \tag{2.27}$$

for the proof of this refer to the work of Andersson and Bro (1998), De Lathauwer et al. (2000b) and Kolda (2006). As $\boldsymbol{\mathcal{X}}$ is a constant, problem 2.26 reduces to maximizing the second term in equation 2.27. To simplify the problem even more, we can maximize this term while fixing all matrices except one and solve for one matrix at a time. In mathematical terms this means that for a certain $1 \leq n \leq 4$, we solve the following problem

$$\begin{aligned} \max_{\widetilde{\mathbf{U}}^{(n)}} & \left\| \boldsymbol{\mathcal{X}} \times_1 \widetilde{\mathbf{U}}^{(1)^{\mathrm{H}}} \times_2 \widetilde{\mathbf{U}}^{(2)^{\mathrm{H}}} \times_3 \widetilde{\mathbf{U}}^{(3)^{\mathrm{H}}} \times_4 \widetilde{\mathbf{U}}^{(4)^{\mathrm{H}}} \right\|_F \\ & \text{subject to } \ \widetilde{\mathbf{U}}^{(n)} \in \mathbb{C}^{I_n \times R_n} \text{ is unitary.} \end{aligned} \tag{2.28}$$

Let us assume that $n = 1$. We can rewrite the cost function from equation 2.28 in matricial form

$$\begin{aligned} \left\| \boldsymbol{\mathcal{X}} \times_1 \widetilde{\mathbf{U}}^{(1)^{\mathrm{H}}} \times_2 \widetilde{\mathbf{U}}^{(2)^{\mathrm{H}}} \times_3 \widetilde{\mathbf{U}}^{(3)^{\mathrm{H}}} \times_4 \widetilde{\mathbf{U}}^{(4)^{\mathrm{H}}} \right\|_F &= \|\widetilde{\mathbf{U}}^{(1)} \mathbf{V}\|_F, \quad \text{where} \\ \mathbf{V} &= \mathbf{X}^{(1)} (\widetilde{\mathbf{U}}^{(4)} \otimes \widetilde{\mathbf{U}}^{(3)} \otimes \widetilde{\mathbf{U}}^{(2)}). \end{aligned} \tag{2.29}$$

The norm is maximized by calculating the SVD of $\mathbf{V}$ and making $\widetilde{\mathbf{U}}^{(1)}$ equal to the first $R_1$ left singular vectors of $\mathbf{V}$. The algorithm stops when it ceases to improve the approximation or it reaches a maximum number of iterations (Kolda and Bader, 2009). This iterative method belongs to the family of ALS methods, and is called Higher-Order Orthogonal Iteration (HOOI, De Lathauwer et al., 2000b). In Chapter 3 we have adopted the Tensor Toolbox from Bader and Kolda (2010) to compute the truncated HOSVD. This toolbox uses the HOOI, delineated in Table 2.1. However, we noticed that our application does not need HOOI and only one iteration of HOOI (equivalent to using the plain truncated HOSVD) is sufficient.

| |
|---|
| **Input** $\boldsymbol{\mathcal{X}}, R_1, R_2, R_3, R_4$ |
| Initialize matrices $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}, \mathbf{U}^{(4)}$ using HOSVD (equation 2.20) |
| **While** fit decreases or maximum number of iterations not reached |
| $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times_2 \mathbf{U}^{(2)^{\mathrm{H}}} \times_3 \mathbf{U}^{(3)^{\mathrm{H}}} \times_4 \mathbf{U}^{(4)^{\mathrm{H}}}$ |
| $\mathbf{U}^{(1)}$ is formed with the $R_1$ left singular vectors from $\mathbf{Z}^{(1)}$ |
| $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{U}^{(1)^{\mathrm{H}}} \times_3 \mathbf{U}^{(3)^{\mathrm{H}}} \times_4 \mathbf{U}^{(4)^{\mathrm{H}}}$ |
| $\mathbf{U}^{(2)}$ is formed with the $R_2$ left singular vectors from $\mathbf{Z}^{(2)}$ |
| $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{U}^{(1)^{\mathrm{H}}} \times_2 \mathbf{U}^{(2)^{\mathrm{H}}} \times_4 \mathbf{U}^{(4)^{\mathrm{H}}}$ |
| $\mathbf{U}^{(3)}$ is formed with the $R_3$ left singular vectors from $\mathbf{Z}^{(3)}$ |
| $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{U}^{(1)^{\mathrm{H}}} \times_2 \mathbf{U}^{(2)^{\mathrm{H}}} \times_3 \mathbf{U}^{(3)^{\mathrm{H}}}$ |
| $\mathbf{U}^{(4)}$ is formed with the $R_4$ left singular vectors from $\mathbf{Z}^{(4)}$ |
| **end** |
| $\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{U}^{(1)^{\mathrm{H}}} \times_2 \mathbf{U}^{(2)^{\mathrm{H}}} \times_3 \mathbf{U}^{(3)^{\mathrm{H}}} \times_4 \mathbf{U}^{(4)^{\mathrm{H}}}$ |
| **Output** $\boldsymbol{\mathcal{G}}, \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}, \mathbf{U}^{(4)}$ |

Table 2.1: HOOI - ALS algorithm to compute the truncated HOSVD with $n$-ranks $(R_1, R_2, R_3, R_4)$ of a tensor $\boldsymbol{\mathcal{X}}$ of size $(I_1, I_2, I_3, I_4)$ (De Lathauwer et al., 2000b).

Figure 2.4 depicts a representation of the truncated HOSVD decomposition for a third-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$. Clearly, the expression for the truncated HOSVD provided in equation 2.24 reduces to $\boldsymbol{\mathcal{X}} \approx \widetilde{\boldsymbol{\mathcal{G}}} \times_1 \widetilde{\mathbf{U}}^{(1)} \times_2 \widetilde{\mathbf{U}}^{(2)} \times_3 \widetilde{\mathbf{U}}^{(3)}$ with $\widetilde{\boldsymbol{\mathcal{G}}} \in \mathbb{C}^{R_1 \times R_2 \times R_3}$. According to De Lathauwer et al. (2000a), the computational cost of the HOSVD is proportional to calculating four matrix SVD (as many SVD as dimensions) of the four matrix unfoldings. The products between the matrix unfoldings to calculate the core-tensor and the cost of computing $\widetilde{\boldsymbol{\mathcal{X}}}$ contribute to the total cost of the algorithm. The cost of calculating the SVD depends on the algorithm. The cost of computing $\mathbf{U}, \boldsymbol{\Sigma}, \mathbf{V}$ of a matrix $\mathbf{A}$ of size $m \times n$ is

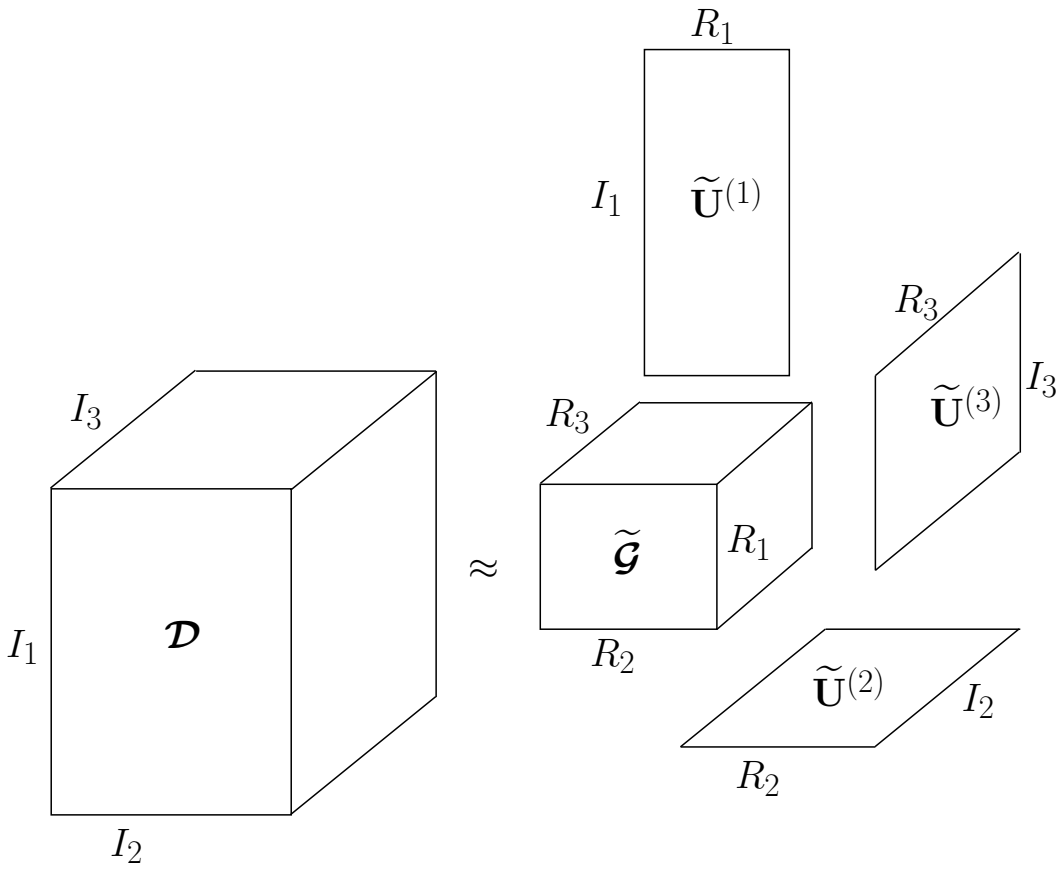



Figure 2.4: Truncated-HOSVD decomposition of a third-order tensor $\boldsymbol{\mathcal{X}}$.

about $\mathcal{O}(n^3)$ (Golub and Van Loan, 1996). In the case of a matrix with $m << n$, other SVD algorithms exist which only compute the first $m$ or $n$ singular values (whichever is smaller), with their respective rows and columns of $\mathbf{U}$ and $\mathbf{V}$, usually denoted "economy" size SVD. In this case, the computational cost decreases significantly and it is about $\mathcal{O}(mr^2)$, where $r$ is the amount of singular values computed. The algorithms of Chapters 3 and 4 use the more efficient aforementioned solvers for the SVD, because the matrix unfoldings naturally satisfy $m << n$. A similar implementation could be used in the algorithm of Chapter 5.

# CHAPTER 3

# HOSVD-based reconstruction [1]

## 3.1 Introduction

This chapter describes a new seismic signal enhancement and reconstruction algorithm that falls under the category of methods that are based on rank reduction techniques. A considerable body of work exists in the area of signal-to-noise enhancement via rank reduction methods. For instance, Freire and Ulrych (1988) and Trickett (2003) employ eigenimage filtering to attenuate random noise. Eigenimage filtering operates in the $T-X$ and $F-X-Y$ domains. Consequently, eigenimage methods can only be used to process data that depend on one or two spatial dimensions. Trickett (2008), Burroughs and Trickett (2009) and Trickett and Burroughs (2009) propose to use Cadzow filtering for denoising data that depend on more than two spatial coordinates. In a subsequent article, Trickett et al. (2010) apply Cadzow filtering to solve the trace reconstruction problem. The Multichannel Singular Spectrum Analysis (MSSA) algorithm proposed by Sacchi (2009) and Oropeza and Sacchi (2011) is equivalent to the Cadzow method proposed by Trickett (2008) for denoising and reconstructing seismic volumes. Cadzow and MSSA algorithms replace the block Hankel matrix of the incomplete and noisy multidimensional observations by its low-rank approximation. Averaging along the anti-diagonals of the rank-reduced Hankel matrix permits to recover the complete data volume (Trickett and Burroughs, 2009; Sacchi, 2009). Cadzow and MSSA algorithms can be easily implemented using the Singular Value Decomposition (Sacchi, 2009). Computational efficiency is gained by replacing the SVD with fast methods to iteratively estimate the eigen-decomposition of large Hankel matrices (Halko et al., 2011a,b; Gao et al., 2013).

[1] A version of this chapter has been published in Kreimer and Sacchi (2012b) and in Kreimer and Sacchi (2012d), Geophysics, http://dx.doi.org/10.1190/geo2011-0399.1

In this work, rather than embedding multidimensional spatial data in a block Hankel matrix, the proposed algorithm operates directly on the seismic tensor (Kreimer and Sacchi, 2012d). In other words, we propose to apply rank reduction to the multidimensional seismic volume represented via a fourth-order tensor. While there is a unique definition of rank for matrices (Strang, 2006), there is no unique decomposition for tensors and, therefore, the definition of rank depends on the decomposition that one adopts to represent the tensor (Bergqvist and Larsson, 2010). This work adopts the HOSVD to represent the data tensor by one of lower rank (refer to Chapter 2). The rank-reduced version of the prestack seismic tensor is used to estimate data at unrecorded positions. It is important to stress that other tensor decomposition methods could be adopted. For instance, CP (Carroll and Chang, 1970; Harshman, 1970) could be used instead of the HOSVD. However, an optimal CP decomposition does not always exist (De Silva and Lim, 2008), and an algorithm for the CP decomposition can give a "degenerate" solution (refer to Chapter 2). Our numerical experiments produced high quality results when the proposed algorithm was implemented via the HOSVD. It is important to stress that the HOSVD can be viewed as a generalization of the classical SVD for tensors. The latter leads to a tensor decomposition that resembles the more familiar decomposition of matrices via the SVD.

We also clarify that noise attenuation and seismic data reconstruction are often treated as separate problems. However, here we follow the framework proposed in Oropeza and Sacchi (2011) and present an algorithm that performs simultaneous multidimensional random noise attenuation and reconstruction.

## 3.2 Theory

### 3.2.1 The tensor completion problem in seismic data processing

Our analysis considers patches of 4D spatial data in the *frequency-space* domain. In other words, a large prestack volume is segmented in windows or patches of data consisting of a desired number of midpoints in the inline and crossline direction, and their associated inline and crossline offsets. Therefore, we will identify the prestack seismic volume with the symbol $D^{obs}(\omega, x, y, h_x, h_y)$ where $\omega$ represents temporal frequency, $x$ and $y$ are midpoint coordinates, and $h_x$ and $h_y$ are inline and crossline offsets, respectively. Naturally, the spatial coordinates could also be source-receiver or midpoint-offset-azimuth (refer to Chapter 1). We consider the case where seismic traces are attributed to a regular midpoint-offset grid through binning. In this case, the data can be represented via the fourth-order tensor $\boldsymbol{\mathcal{D}}^{obs}$, with elements $D^{obs}_{ijkl}$, where $i, j, k, l$ are bin indices for the spatial coordinates $x, y, h_x, h_y$, respectively. Notice that we have dropped the dependency on $\omega$ but bear in mind that the

proposed reconstruction method is applied to all frequencies. In the case that more than one trace is assigned to a bin, we average them to retain only one observation in each bin.

The ideal survey contains an observation in each element of the 4D tensor $\boldsymbol{\mathcal{D}}$. In real situations, however, mapping seismic traces from irregular source-receiver coordinates to midpoint-offset bins leads to a highly sparse tensor. In other words, a large number of bins $i, j, k, l$ are empty. This point will become clear in our field data example. The ideal survey $\boldsymbol{\mathcal{D}}$ and the observed data $\boldsymbol{\mathcal{D}}^{obs}$ are related through a 4D sampling operator $\boldsymbol{\mathcal{T}}$

$$\boldsymbol{\mathcal{D}}^{obs} = \boldsymbol{\mathcal{T}} \cdot \boldsymbol{\mathcal{D}}, \tag{3.1}$$

where the product $(\cdot)$ is in the Hadamard sense (element to element). The sampling operator is defined in the following expression

$$T_{ijkl} = \begin{cases} 1 & \text{if the grid point } \ ijkl \ \text{ contains an observation} \\ 0 & \text{if the grid point } \ ijkl \ \text{ does not contain an observation}\,. \end{cases} \tag{3.2}$$

It is clear that $\boldsymbol{\mathcal{D}}^{obs} = \boldsymbol{\mathcal{T}}.\boldsymbol{\mathcal{D}}^{obs}$. Our goal is to recover the ideal tensor $\boldsymbol{\mathcal{D}}$ from $\boldsymbol{\mathcal{D}}^{obs}$. The problem stated by equation 3.1 is underdetermined because there are more unknowns than observations. Clearly, equation 3.1 admits an infinite number of solutions. Typically, one way of solving this problem is through constrained inversion methods. Minimum Weighted Norm Interpolation (Liu, 2004), for instance, attempts to reconstruct the data by minimizing a spectral norm that imposes sparsity on the wave-number spectrum of the ideal data set. In this article, however, we propose to estimate the missing observations by assuming that the ideal data $\boldsymbol{\mathcal{D}}$ can be represented via a low-rank tensor (Liu et al., 2009; Acar and Yener, 2009). We present the proof of this statement (for the noise-free case) in Appendix A.

### 3.2.2 Reconstruction and denoising via tensor completion

We assume that the ideally sampled data can be approximated with great fidelity by equation 2.24, given in the previous chapter. The reason for this is that missing traces and noise increase the rank of the tensor, such that decreasing its rank will have the desired effect. In the presence of missing observations we propose to recover the ideal tensor by a single imputation algorithm (Nelson et al., 1996; Grung and Manne, 1998; Walczak and Massart, 2001; Tomasi and Bro, 2005; Oropeza and Sacchi, 2010)

$$\boldsymbol{\mathcal{D}}^{\nu} = \boldsymbol{\mathcal{D}}^{obs} + (\boldsymbol{1} - \boldsymbol{\mathcal{T}}) \cdot \widetilde{\boldsymbol{\mathcal{D}}}^{\nu-1}, \tag{3.3}$$

where $\nu$ denotes iteration number and $\boldsymbol{1}$ is a tensor of ones having the same dimensions of $\boldsymbol{\mathcal{T}}$. We remind the reader that the tensor $\widetilde{\boldsymbol{\mathcal{D}}}$ (see equation 2.24) is the rank-reduced version

of the tensor $\boldsymbol{\mathcal{D}}$.

This formula was also used by Oropeza and Sacchi (2009) where it was mentioned its similarity to the method of Projection onto Convex Sets (POCS) (Abma and Kabir, 2005). It is important to stress that the rank reduction operator does not conform a convex set (Dologlou et al., 1996). Therefore, we cannot use the asymptotic property of POCS for a large number of iterations to analyze the convergence of the proposed algorithm. The rank reduction operation (equation 2.24) is not sufficient to recover the missing data and for this reason the data to be rank-reduced is replaced by the expression on the right hand side of equation 3.3. Notice that the new variable $\boldsymbol{\mathcal{D}}^{\nu}$ does not contain missing values. This imputation algorithm can be considered to be an Expectation Maximization (EM) method when the variables involved are normalized and the residuals are independent. The convergence properties of equation 3.3 have been studied by Tomasi and Bro (2005) who derived the algorithm using the EM method (Little and Rubin, 1987; McLachlan and Krishnan, 1997; Hastie et al., 2001). The EM method is an iterative method used to solve maximum-likelihood estimation problems in the presence of missing data and dates back to 1925 (M'Kendrick, 1925). The EM method is based on two steps: given a first approximation of the missing values, the E-step finds the conditional expectation of the likelihood function. The M-step estimates the maximum likelihood of the parameters disregarding the missing samples. In our application, the likelihood function is $\|\boldsymbol{\mathcal{D}}^{\nu} - \widetilde{\boldsymbol{\mathcal{D}}}^{\boldsymbol{\nu-1}}\|_F$, same as the one defined in equation 2.26. Given a first approximation of the HOSVD of $\boldsymbol{\mathcal{D}}$, the E-step consists of the calculation of this cost function, the expected value of the likelihood function, including the imputation step in equation 3.3. The M-step involves maximizing this cost function as shown for equation 2.28, which is simply using HOOI to rank-reduce the newly imputed data.

The imputation algorithm preserves the original data in their original spatial bins. Missing observations, on the other hand, are replaced by rank-reduced estimates of the tensor $(\boldsymbol{1} - \boldsymbol{\mathcal{T}}) \cdot \widetilde{\boldsymbol{\mathcal{D}}}^{\nu-1}$ computed from the previous iteration. The latter can result in a solution that contains traces with different character (noisy traces at observed positions and filtered traces at recovered positions). This problem can be alleviated by a modified imputation algorithm. Essentially, we replace $\boldsymbol{\mathcal{D}}^{obs}$ in expression 3.3 by the average $a\,\boldsymbol{\mathcal{D}}^{obs} + (1-a)\,\boldsymbol{\mathcal{T}}\,\widetilde{\boldsymbol{\mathcal{D}}}^{\nu-1}$, such that now the algorithm becomes

$$\boldsymbol{\mathcal{D}}^{\nu} = a\,\boldsymbol{\mathcal{D}}^{obs} + (\boldsymbol{1} - a\,\boldsymbol{\mathcal{T}}) \cdot \widetilde{\boldsymbol{\mathcal{D}}}^{\nu-1}, \tag{3.4}$$

where $a$ is a scalar, $a \in [0, 1]$ (Oropeza, 2010). For datasets with high $SNR$, equation 3.3 applies, and therefore $a = 1$. For data with low $SNR$, the parameter $a < 1$ and is found by examining the reconstruction error. In other words, we compute the difference between the observed data and the reconstructed data at observed grid points and alter the parameter $a$

until finding a suitable one where the reconstruction error shows little coherent energy. This parameter is kept constant throughout the iterations (refer to synthetic experiment in the following section). The last iteration is a denoising step, in which we perform rank reduction on the reconstructed volume to correct for the possible discrepancies in $SNR$ that might still be present. It is important to stress that at the present time there is no automatic way of finding the optimal core-tensor size and the optimal parameter $a$.

In Cadzow or MSSA filtering and reconstruction the rank is proportional to the number of distinct plane waves (Sacchi, 2009). In tensor completion, the $n$-ranks, or the size of the core-tensor, have an upper bound proportional to the number of distinct plane waves or curved events (Appendix A). However, the curvature of the events has to be separable in the different coordinates. For a non-separable curvature there is no such a clear theoretical justification for parameter selection and we acknowledge that more research is needed to understand ways of selecting the size of the optimal core-tensor for these type of reflections.

## 3.3 Synthetic examples

We design a series of synthetic models to test the proposed method. The models are designed to evaluate the robustness of the method to curvature. They are not physically realistic and contain three parabolic events in four spatial dimensions to simulate the extreme case of curvature in all axes. We use a grid size of $12\times12\times12\times12$ and a $SNR = 1$ and 100 (where $SNR$ is defined as the variance of the data divided by the variance of the noise). Furthermore, we randomly remove 25, 50, 75 and 90% of the generated traces. A maximum number of iterations of 20 is used in the iterative algorithm for all examples. We test two scenarios. First, the noise-free case ($SNR = 100$) where $a = 1$. Second, a case with $SNR = 1$ where $a = 0.9$. The algorithm is applied in the $frequency - space$ domain for frequencies in the $1 - 70$ Hz range. We use a Ricker wavelet with central frequency of 20 Hz. The midpoint increments in $x, y$ are 50 m. The offset increments in $x, y$ are 100 m. The synthetic events start to show aliasing at a temporal frequency of about $20 - 25$ Hz.

To examine the performance of the algorithm we calculate the quality $Q$ of the recovered data in dB in the *time* domain

$$Q = 10 \log \frac{\|\boldsymbol{\mathcal{D}}_{\text{true}}\|_F^2}{\|\boldsymbol{\mathcal{D}}_{\text{recovered}} - \boldsymbol{\mathcal{D}}_{\text{true}}\|_F^2}, \tag{3.5}$$

where $\boldsymbol{\mathcal{D}}_{\text{true}}$ is the non-decimated seismic volume with no noise and $\boldsymbol{\mathcal{D}}_{\text{recovered}}$ is the reconstructed and denoised volume. The parameter $Q$ is calculated using all the traces and Table 3.1 displays this scale linearly with respect to the quotient in equation 3.5. Values of $Q$ larger than 10 dB can be considered satisfactory.

| $Q$ (dB) | $\|\boldsymbol{\mathcal{D}}_{\text{true}}\|_F^2 = \alpha\|\boldsymbol{\mathcal{D}}_{\text{recovered}} - \boldsymbol{\mathcal{D}}_{\text{true}}\|_F^2$ <br> $\alpha$ |
|---|---|
| 2 | 1.6 |
| 4 | 2.5 |
| 6 | 4 |
| 10 | 10 |

Table 3.1: Linear scale interpretation of the quality of the reconstruction $Q$. The scalar $\alpha$ is the proportionality value between the Frobenius norm of the misfit and the Frobenius norm of the fully sampled volume.

Figure 3.1 portrays $Q$ values for the simulation. As expected, the quality of the reconstruction decreases for volumes with a large amount of missing traces and low values of $SNR$. For completeness, we plot the convergence curve for three of the frequencies in the $1-70$ Hz range for the case of $SNR = 1$ (Figure 3.2). In particular we measure the normalized change of the solution using the expression $\|\boldsymbol{\mathcal{D}}_\omega^\nu - \boldsymbol{\mathcal{D}}_\omega^{\nu-1}\|_F^2/\|\boldsymbol{\mathcal{D}}_\omega^{\nu-1}\|_F^2$, where $\boldsymbol{\mathcal{D}}_\omega^0$ are the observations $\boldsymbol{\mathcal{D}}^{obs}$. We have used the subindex $\omega$ to emphasize that this normalized change has been computed for three fixed frequencies $\omega_1, \omega_2$ and $\omega_3$. It is evident that a simple stopping criterion can also be used to terminate the reconstruction prior to reaching the maximum number of iterations. In our simulations a relative change of 0.5% is sufficient to terminate the iteration.

Figure 3.3 displays the synthetic, decimated and recovered data for the case of 90% decimation and $SNR = 100$ and 1. We stress that we are plotting only a subset of the $12 \times 12 \times 12 \times 12$ spatial cube. The $Q$ values for these two examples are 12.6 and 8.36 dB, respectively. This means that the Frobenius norm of the original data are approximately 18 and 7 times larger, respectively, than the Frobenius norm of the reconstruction error. The fold map for this synthetic is shown in Figure 3.4. The running time for each of these examples is of 1 minute 36 seconds in a desktop computer with an Intel Core 2 Duo processor with a speed of 3.06GHz, running MATLAB in a single processor. For this example we select a core-tensor of size $3 \times 3 \times 3 \times 3$ and a parameter $a = 0.9$. We have already stressed that the choice of these parameters is done by inspecting the residuals. At the present time, we have not found an automatic way of estimating these parameters. It is interesting to notice that the curvature of the events does not affect the quality of the reconstruction. This appears to be an important advantage with respect to reconstruction methods based on Fourier synthesis (Liu, 2004). Fourier reconstruction methods assume that the data are composed of a superposition of plane waves, the sparsity or simplicity assumption is utilized to retrieve a model that consists of a few dominant wavenumbers that honour the observations. Clearly, the plane wave model assumption is invalid in the presence of curvature. We have applied

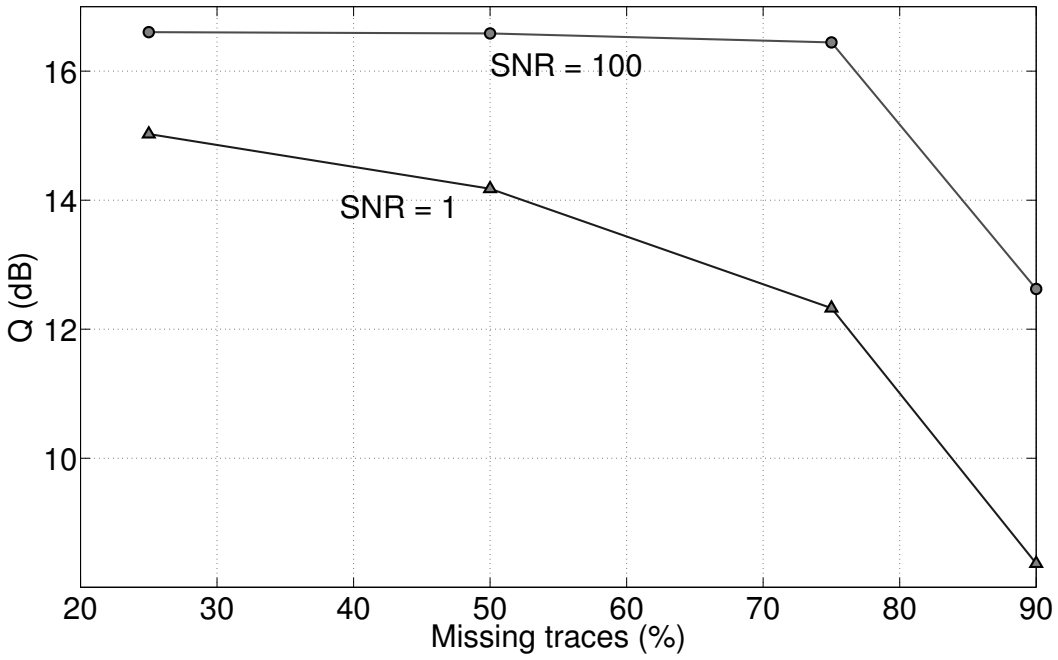



Figure 3.1: Quality of the reconstruction of the reconstructed volumes for different sparsity levels and $SNR$.

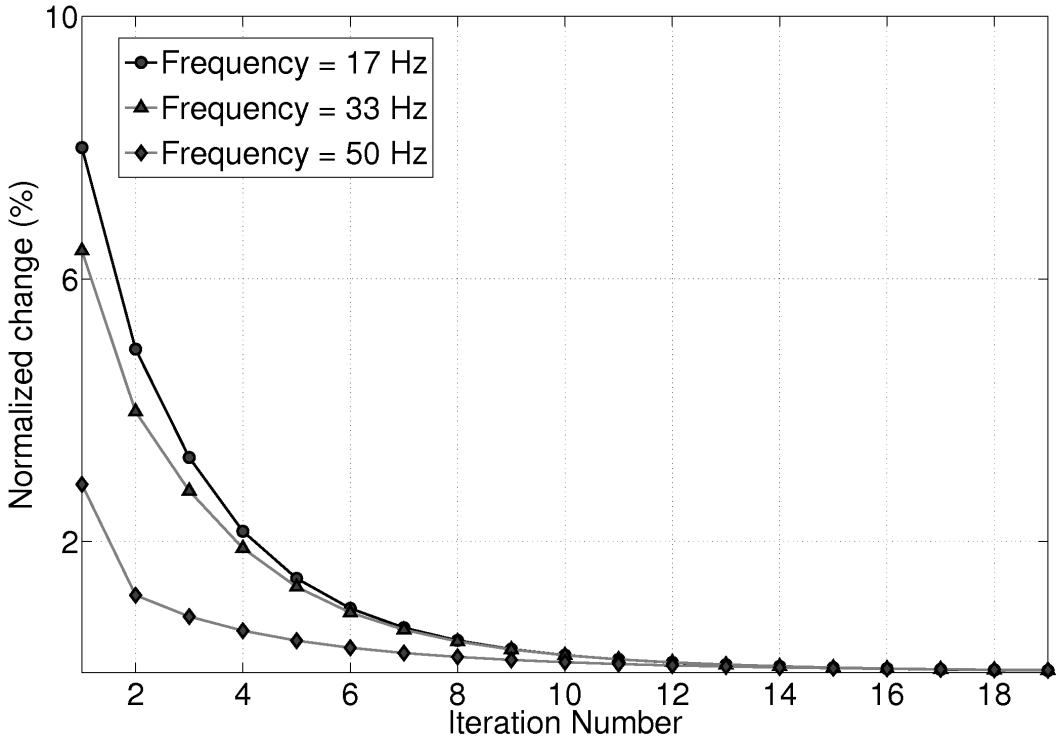



Figure 3.2: Normalized change between iterations for frequencies 17, 33 and 50 Hz.

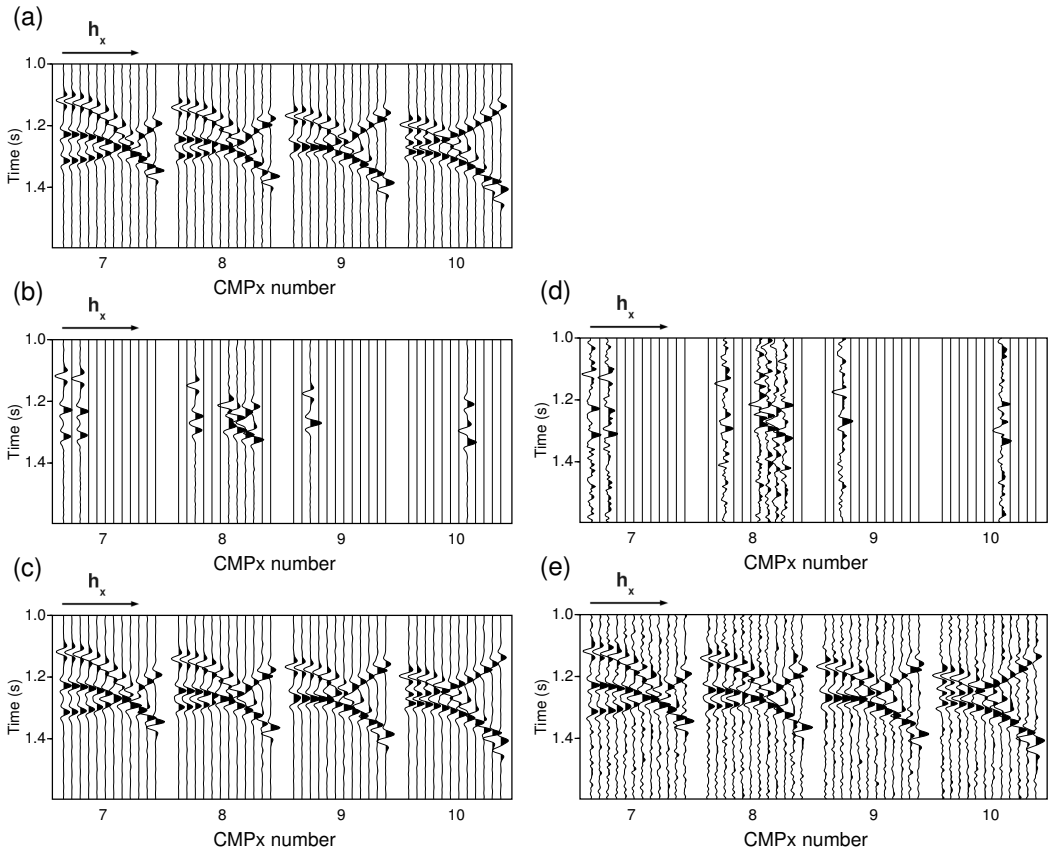



Figure 3.3: (a) Synthetic prestack volume. The data consist of $12 \times 12 \times 12 \times 12$ bins. A small subset of the data is portrayed. (b) 90% decimated data with $SNR = 100$. (c) Data recovered from (b). (d) 90% decimated data with $SNR = 1$. (e) Recovered data from (d).

Minimum Weighted Norm Interpolation (Liu, 2004) to the same data set and the results were always of very low quality. For instance, 90% decimation and $SNR = 100$ results in a quality figure of $Q = 2$ dB. One way to overcome this shortcoming is by running Fourier reconstruction methods on small spatio-temporal windows. We stress that in this example the tensor completion is applied on the $12 \times 12 \times 12 \times 12$ spatial cube without segmenting the cube into small spatio-temporal windows.

In the previous examples we have used a reinsertion parameter $a = 0.9$ based on a sensitivity experiment in which we try different values for $a$ and examine the residual panel. The synthetic model used in this test is the same one described at the beginning of this section, with $SNR = 1$ and 70% missing traces. Figure 3.5 shows the results. Values of $a$ larger than

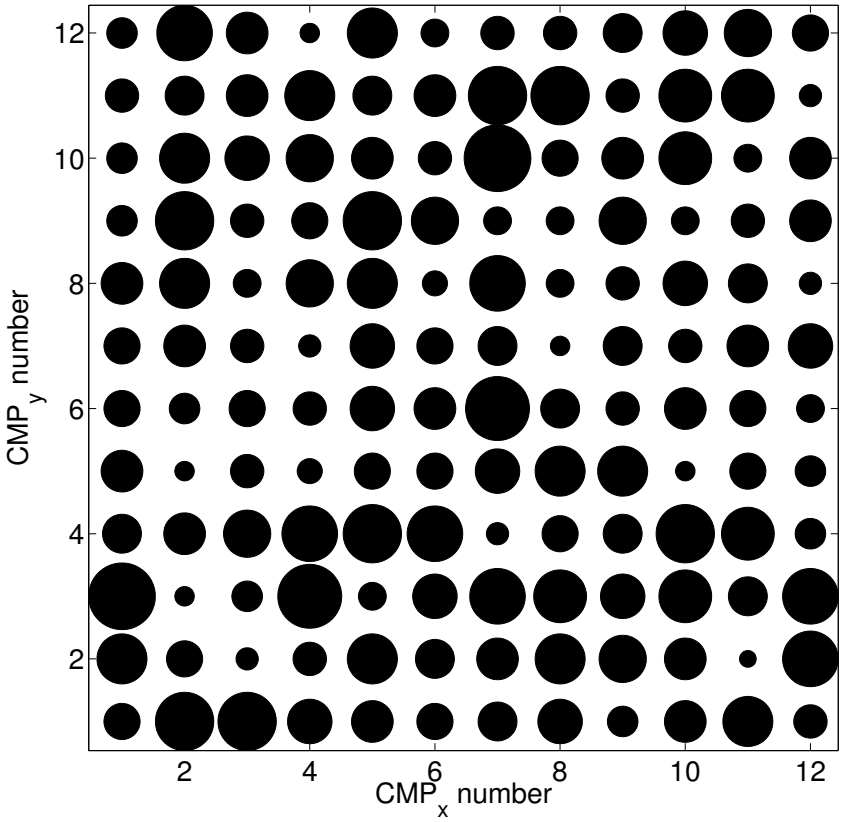



Figure 3.4: Fold map for synthetics portrayed in Figure 3.3. The area of the circles is proportional to the fold. The maximum fold is 24 and the minimum fold is 6.

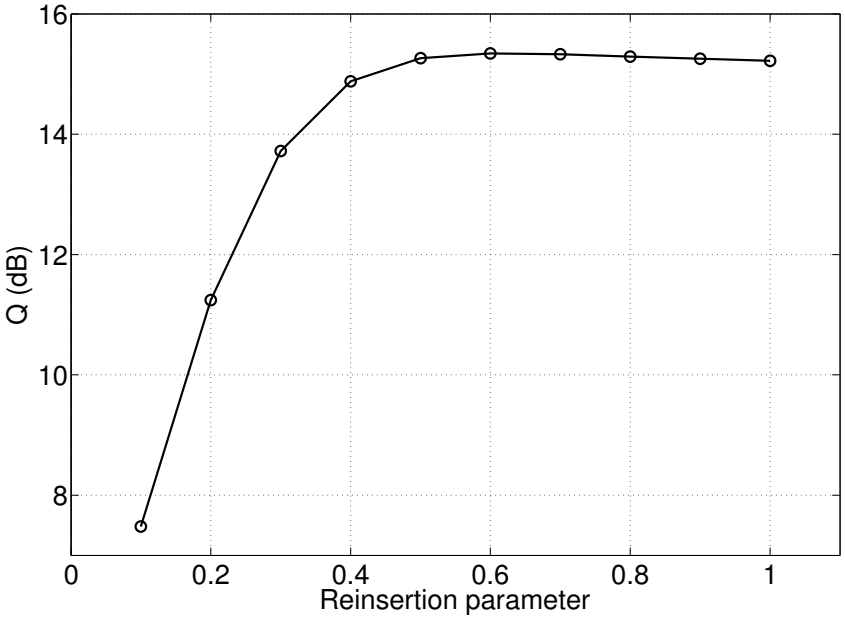



Figure 3.5: Sensitivity of $Q$ to the change of the reinsertion parameter $a$.

0.4 lead to the best balance between the reinsertion of observed traces and conservation of rank-reduced traces. Between 0.4 and 0.9, very little variation of $Q$ manifests in the curve. In our tests with real data, $a = 0.9$ was the value that gave the best results. We chose to use this value throughout all the tests.

**Performance of tensor completion for 2D, 3D and 4D spatial volumes**

We propose a test to assess the benefits of reconstructing higher-order arrays over lower-dimensional volumes. We simulate 2, 3 and 4D spatial volumes of sizes $12^n, n = 2, 3, 4$ containing two parabolic events with opposite curvatures. We randomly remove 50% of the data and add random noise with $SNR = 100$ and 1. We use core-tensors of sizes $2^n, n = 2, 3, 4$ and a parameter $a = 0.9$ . A maximum iteration number of 20 is utilized for each frequency in the range of $1 - 16$ Hz. The time sampling is 2 ms and the number of time samples is 128. The midpoint and offset increments are 10 m. We present the quality of the reconstruction $Q$ for these examples in Table 3.2. The standard deviation $\sigma_Q$ is calculated creating different realizations of random noise and sampling operators. It is clear from the table that using 4D spatial reconstruction for both cases of high and low $SNR$ leads to an improved value of the reconstruction quality. In other words, by increasing the dimensionality of the survey, we also increase the number of traces and the redundancy

| Number of Spatial Dimensions | $Q \pm \sigma_Q$ [dB] | |
|---|---|---|
| | $SNR = 100$ | $SNR = 1$ |
| 2 | $14.0 \pm 2.4$ | $1.3 \pm 0.4$ |
| 3 | $16.7 \pm 0.03$ | $9.8 \pm 0.15$ |
| 4 | $19.1 \pm 0.006$ | $16.7 \pm 0.1$ |

Table 3.2: Quality of the reconstruction for different number of spatial dimensions.

that is needed for the data completion algorithm. This is also discussed by Trad (2009) in the context of MWNI reconstruction.

### 3.3.1 Synthetic tests on different degrees of curvature

Most reconstruction methods currently in use assume that seismic events are linear in a window, e.g. MWNI (Liu and Sacchi, 2004), POCS (Abma and Kabir, 2006), ALFT (Xu et al., 2005). However, diffractions often appear in prestack data, because of fractures, or small-scale features such as channels. The aforementioned methods either fail to reconstruct these features, or they smooth them. As a consequence, prestack migration is not able to collapse the energy of these points effectively. Reconstruction methods are generally criticized because of their inability to handle non-linear events inside a window.

Reconstructing events with curvature and aliased in space requires the use of methods such as the one from Naghizadeh and Sacchi (2009), which is a modification of Spitz (1991) to handle spatially variant dip. Both of these methods make use of prediction filters and reconstruct aliased events using information from un-aliased low frequencies, and can deal with regularly missing data. However, the former method does not make the assumption of linear events in a window, and can deal with irregularly missing traces as well. Curvelet reconstruction (Naghizadeh and Sacchi, 2010a) can also be used for reconstructing curved events that are aliased in space. However, curvelets assume local linear events and therefore will not be able to reconstruct large gaps.

The next example comprises testing the algorithm with three different degrees of curvature, low, moderate and strong, in three spatial coordinates. We create synthetic data with one event and another synthetic with two events. The size of the volume used is $150 \times 20 \times 20 \times 20$. We randomly remove 50% of the traces in all these synthetics. In these examples we use different sizes for the core-tensor, with the intention of determining the influence of the

core-size on the reconstruction of events with curvature. We also add two different levels of noise.

Figure 3.6 displays the quality of the reconstruction $Q$ for the case of one curved event versus the core-tensor size, and $SNR = 100$ and 1. The core-tensor size should be interpreted as the number given in the horizontal axis to the power of 3. The dotted line represents events with low degree of curvature, the continuous line represents events with moderate curvature, and the dashed line represents events with strong curvature. We can extract some conclusions from this figure. First, the core-tensor size does not seem to influence the quality of the reconstruction for the different types of curvature. However, the core-tensor size does influence the quality of the recovery when the $SNR$ is low. In this case, the smaller the core-tensor size is, the more effectively the method can filter the noise. Figure 3.7 displays the case of $SNR = 1$ for one CMP gather, and for the three different curvatures designed. The core-tensor size in this case is 1. The $F - K$ plots for the same gathers from Figure 3.7 are displayed in Figure 3.8. The frequency and wavenumber axis in this figure and Figure 3.13 are normalized, such that one needs to divide by $\Delta t$ or $\Delta x$ respectively to obtain the axis in Hz or in Cycles/m.

Figure 3.9 displays the quality of the reconstruction $Q$ for the case of two curved events versus the core-tensor size, and $SNR = 100$ and 1. The same convention for the lines as in Figure 3.6 applies. In this case, it is more important for the quality of the reconstruction to use an appropriate core-tensor size. As a general criteria, we deduce from this experiment that is preferable to use a larger rather than a smaller core-tensor size. We also see that the lower the curvature of the events, the better the quality of the reconstructions. This is true if the data have a low level of noise. Another observation we make is that both cases of moderate and high curvature are better represented by a core-tensor of size $4^3$ when the $SNR = 100$. We cannot observe the same in the case of low $SNR$, most probably because the noise is filtered by a small sized core-tensor. The case of two events and $SNR = 1$ is portrayed in Figure 3.10, for only one CMP gather, and for the three different curvatures. For the results shown in this figure we used a core-tensor size of $2^3$. Figure 3.11 shows the $F - K$ plots for the gathers presented in Figure 3.10. As before, the frequency and wavenumber axis are normalized.

### 3.3.2 Core-tensor size role

In the previous examples, we used a core-tensor size that gave a reasonable value of $Q$. To explore this important parameter for the reconstruction algorithm, we design the following test. We create synthetics with two curved events, and a volume size of $150 \times 20 \times 20 \times 20$. There are four different synthetics in this experiment, as follows

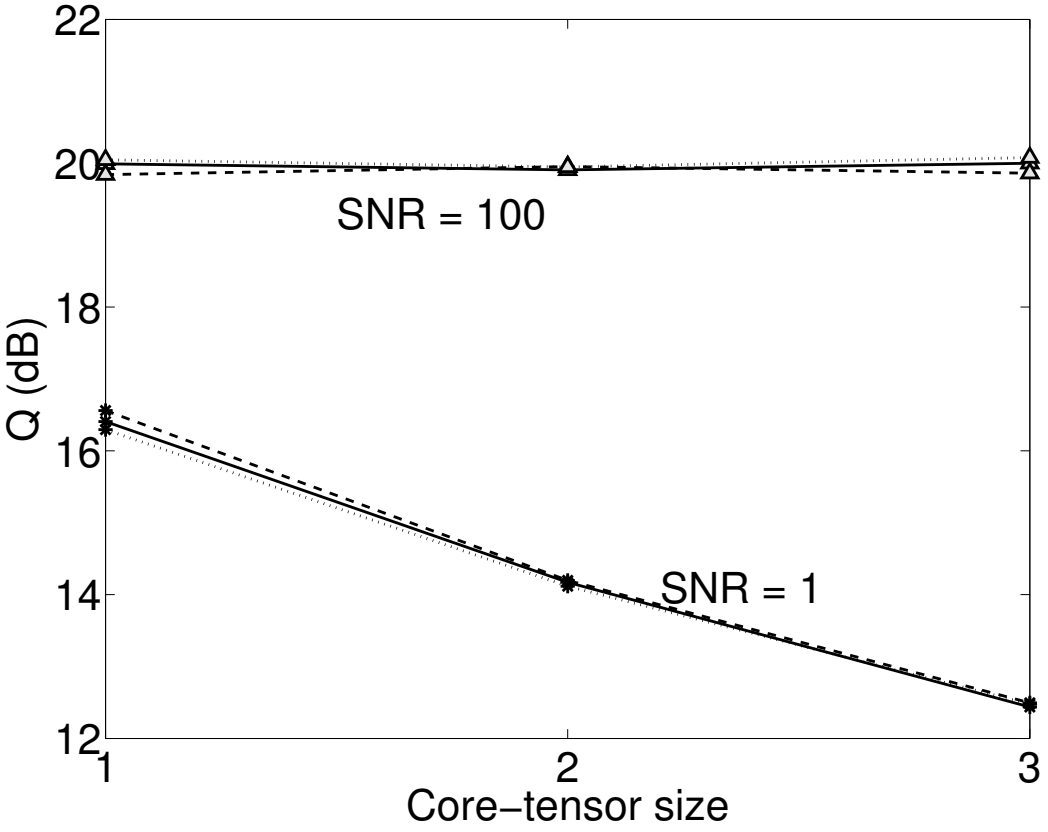



Figure 3.6: Quality of the reconstruction for one curved event with different degrees of curvature, three different core-tensor sizes, and different SNR.

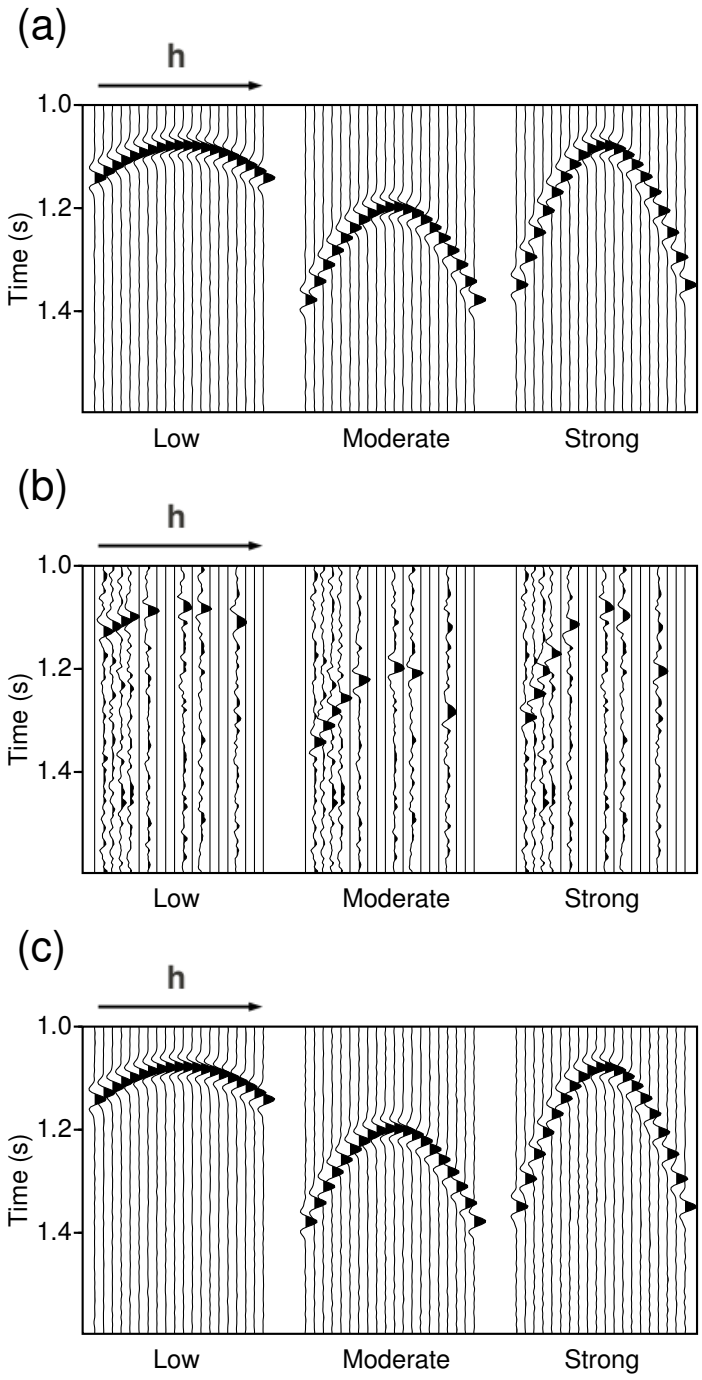



Figure 3.7: CMP-$x$ gather number 10 with CMP-$y = 10$ for three different degrees of curvature, and $SNR = 1$. (a) CMP gathers with $SNR$=100 and no decimation. (b) Decimated gathers with noise added, $SNR = 1$. (c) Data recovered from (b).

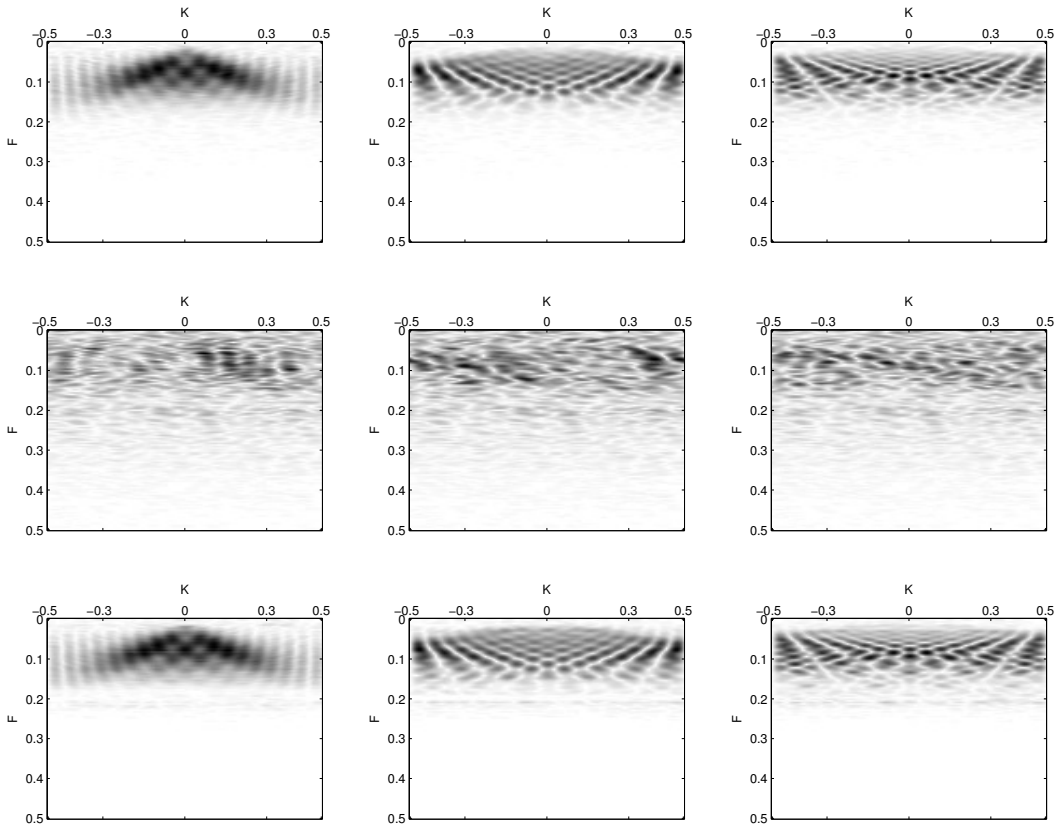



Figure 3.8: $F-K$ plots from the gathers shown in Figure 3.7. First row shows the complete data with $SNR = 100$, second row shows the decimated data with noise added ($SNR = 1$) and third row shows the reconstructed and denoised data. The columns correspond to the low, moderate and strong curvature case, respectively, from left to right.

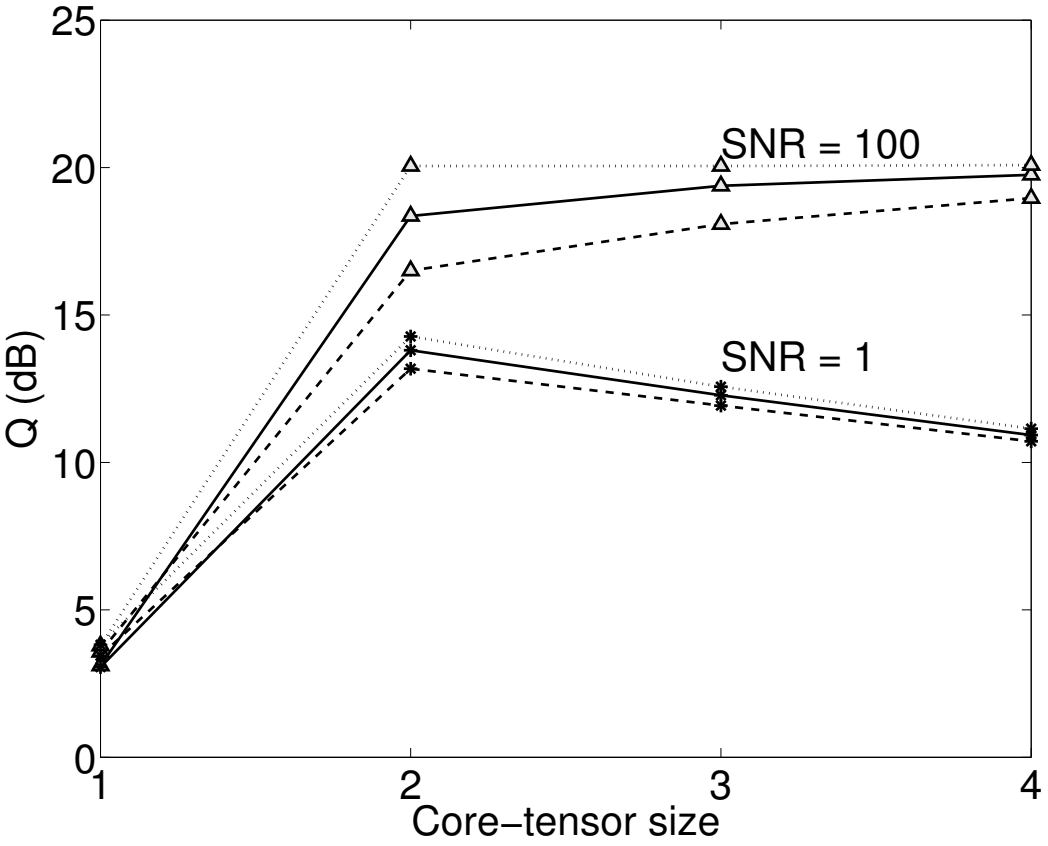



Figure 3.9: Quality of the reconstruction for two curved events with different degrees of curvature, four different core-tensor sizes, and different $SNR$.

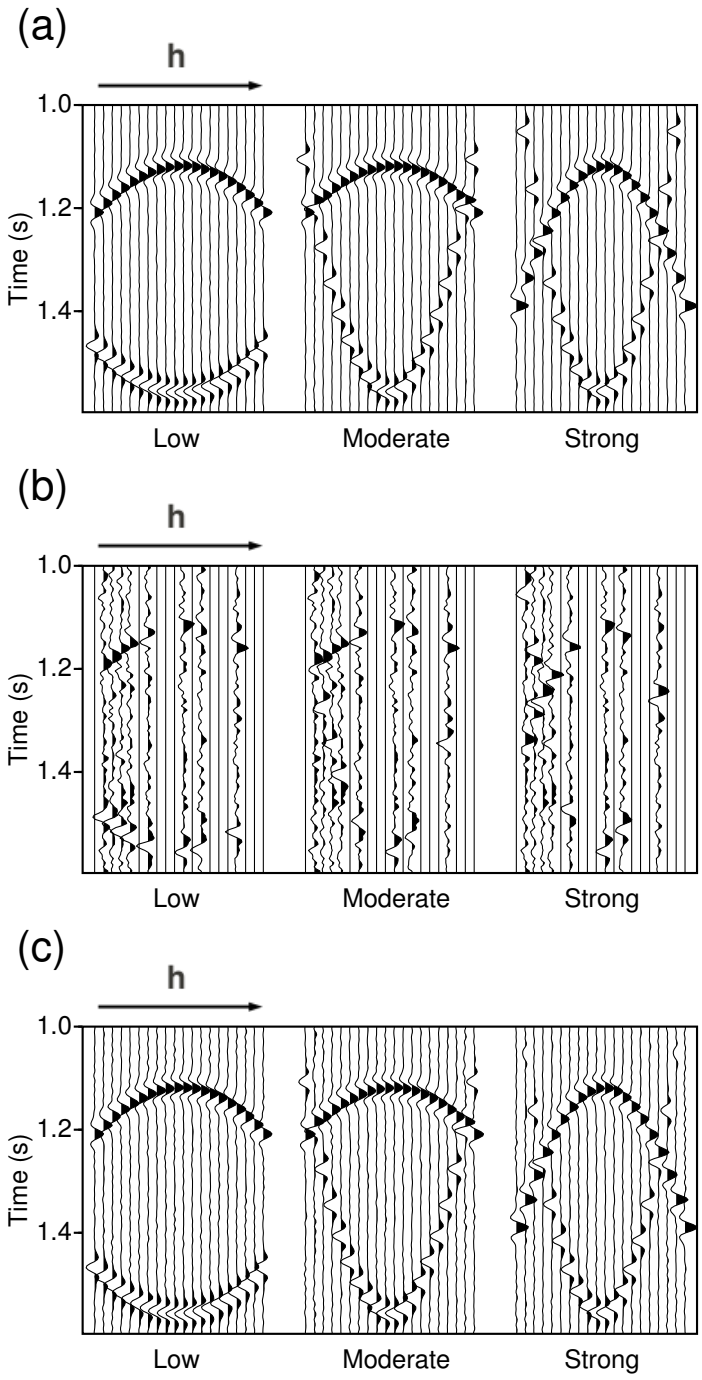



Figure 3.10: CMP-$x$ gather number 10 with CMP-$y = 10$ for three different degrees of curvature, and $SNR = 1$. (a) CMP gathers with $SNR$=100 and no decimation. (b) Decimated gathers with noise added, $SNR = 1$. (c) Data recovered from (b).

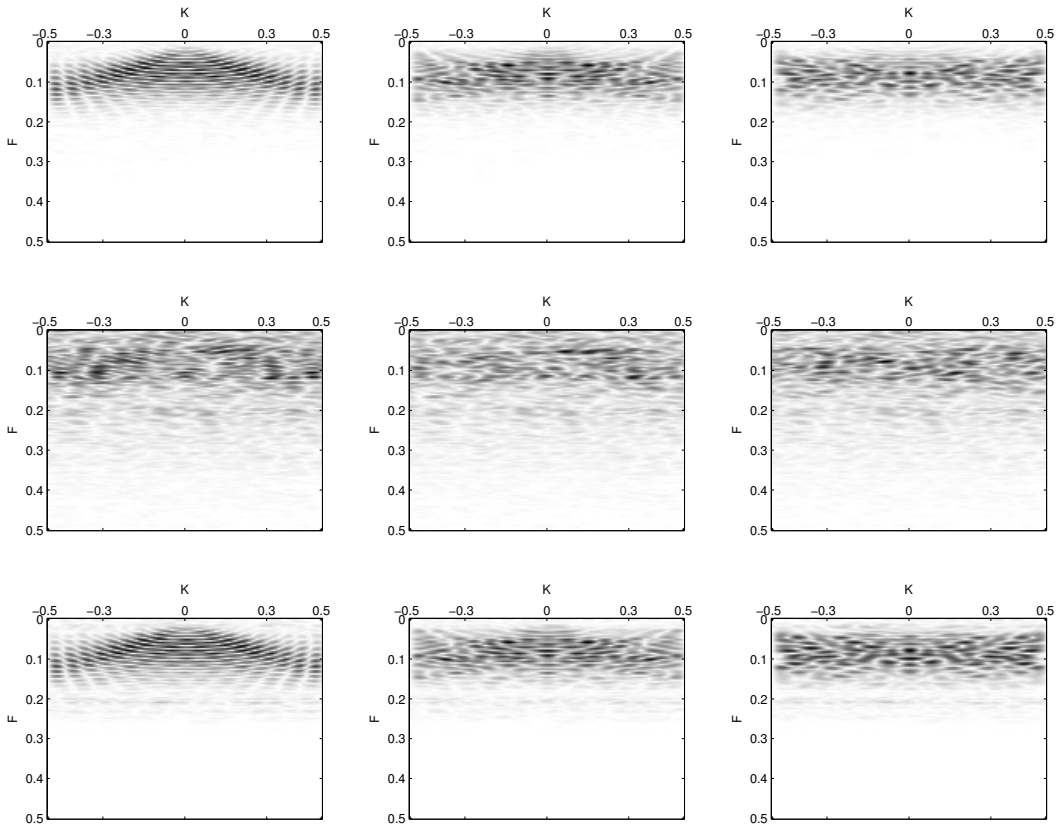



Figure 3.11: $F-K$ plots from the gathers shown in Figure 3.7. First row shows the complete data with $SNR = 100$, second row shows the decimated data with noise added ($SNR = 1$) and third row shows the reconstructed and denoised data. The columns correspond to the low, moderate and strong curvature case, respectively, from left to right.

**a)** $SNR = 100$ and no missing traces, symbolized by the blue line in Figure 3.14.

**b)** $SNR = 100$ and 50% missing traces, symbolized by the black line in Figure 3.14.

**c)** $SNR = 1$ and no missing traces, symbolized by the green line in Figure 3.14.

**d)** $SNR = 1$ and 50% missing traces, symbolized by the red line in Figure 3.14.

As in the previous examples, the core-tensor size should be interpreted as the number given in the horizontal axis in Figure 3.12, to the power of 3. For each synthetic we use different sizes for the core-tensor, from 1 to $20^3$. The optimal value of $Q$ depends on the level of decimation and the level of $SNR$. From Figure 3.12, we can see that case a) has an optimal $Q$ for a core-tensor size of $20^3$. This is the only case from all the examples in which compressing the core-tensor does not lead to a higher value of $Q$. The reason for this is that this volume is full, no missing data, and has a very small amount of noise. Therefore, the compression of the core-tensor has the unwanted effect of removing part of the signal. Case b) has an optimal $Q$ value for core-tensor sizes ranging from $4^3$ to $7^3$. The effect of shrinking the core-tensor in this case is not to filter noise, but to aid in the reconstruction of the missing traces. Case c) and d) have similar behaviour, with the best values being those of case c). For both cases, using a core-tensor of size $2^3$ gives the optimal value of $Q$. In this case, reducing the core-tensor size has the simultaneous effect of filtering most of the random noise and recovering the missing traces.

## 3.4 Field data example

Figure 3.13 displays the shot and receiver geometry for an orthogonal survey from the Western Canadian Sedimentary Basin. The area in the center of the patch highlights the midpoints that are reconstructed using the distribution of sources and receivers portrayed in the figure. In addition, the fold map for this area is included in Figure 3.14. The area of the circles is proportional to the fold. The maximum fold is 66 and the minimum fold is 25. The $x$ and $y$ coordinates represent Easting and Northing in metres, respectively. Figure 3.15 displays five common-midpoint gathers for a fixed CMP-$y = 7$ and offset-$y = 6$. The position of these gathers is indicated by the dashed line rectangle on Figure 3.14. Figure 3.16 portrays four common-offset gathers for a fixed CMP-$y = 10$ and offset-$y = 5$. The midpoint-offset grid has dimensions $16 \times 18 \times 12 \times 12$ ($16 \times 18$ midpoint bins and $12 \times 12$ offset bins). The CMP numbers are indicated in the corners of the fold map in Figure 3.14. The midpoint spacing in $x$ is 47 m and 84 m in $y$. The minimum offset is $-3400$ m in $x$ and $-2600$ m in $y$, and the maximum is 3000 m , in both $x$ and $y$ directions. The total amount of traces in the original data patch is 16060. The midpoint-offset grid contains

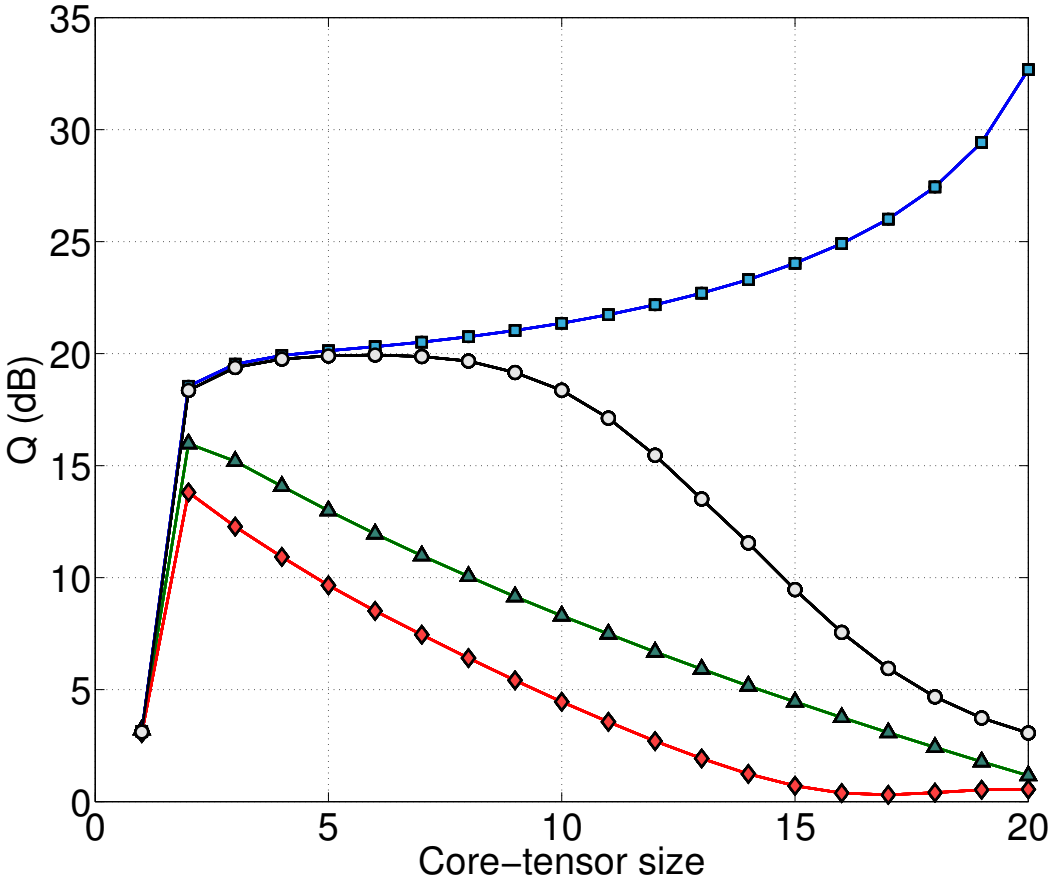



Figure 3.12: Quality of the reconstruction for all core-tensor sizes. The blue line with square icons represents the case of $SNR = 100$ and no missing traces. The black line with circle icons represents the case of $SNR = 100$ and 50% missing traces. The green line with triangular icons represents the case $SNR = 1$ and no missing traces. The red line with rhombus icons represents the case $SNR = 1$ and 50% missing traces.

$16 \times 18 \times 12 \times 12 = 41472$ grid points. In essence, only 39% of the 4D grid is populated. In other words, the proposed algorithm recovers 61% of the traces. To confirm that the reconstruction has not created new events in the volume, Figure 3.17 shows the stack of the non-reconstructed and reconstructed volumes. In this case the algorithm operates on each frequency of the NMO-corrected dataset, in the band 0 to 120 Hz using a maximum number of 30 iterations per frequency. We also add the stopping criteria where a normalized change of 0.5% between two iterations is sufficient to terminate the iterations. The relaxation parameter $a = 0.9$ is selected.

Figure 3.18 contains the spectrum before and after the reconstruction, for a common offset gather and a fixed inline. The distribution of singular values before and after rank reduction is highlighted in Figure 3.19 for three frequencies (low, mid and high frequencies). For each unfolding of the 4D tensor, we calculate its singular value distribution and, as all the magnitudes are similar for all modes, we average them to display just one curve. We use a core-tensor size of $5 \times 5 \times 5 \times 5$ for the rank reduction, which is evident from Figure 3.19, where we can observe that the singular values are nearly 0 after the fifth singular value after reconstruction. Although we do not observe a sharp decrease in singular values magnitude until the 12th singular value, for all three frequencies, the rank reduction to a smaller number of singular values is still effective for the reconstruction because the signal information is mostly concentrated in the first singular values. It is clear that tests with more complex datasets are required to fully understand the limits of the algorithm presented in this chapter.

This particular example took about 2 minutes on a single processor Intel Xeon(R) running at 3.16Ghz using MATLAB. This is not too far off from the time needed by our own MATLAB implementation of Fourier reconstruction using MWNI (Trad, 2009) and sparse inversion (Zwartjes and Hindriks, 2001; Zwartjes and Gisolf, 2007). We stress that commercial codes for seismic data reconstruction are designed using a plethora of cost-cutting strategies that are often not discussed in scientific journals. It is difficult, without having access to industrial codes, to perform proper time comparisons to access the computational efficiency of various existing algorithms.

## 3.5 Conclusions

We have introduced a new method for reconstructing and denoising prestack data that depend on four spatial dimensions. The HOSVD was utilized to develop a rank reduction strategy for tensor denoising and completion. The algorithm operates in the $frequency - space$ domain and solves the tensor completion problem at each temporal frequency via an iterative algorithm.

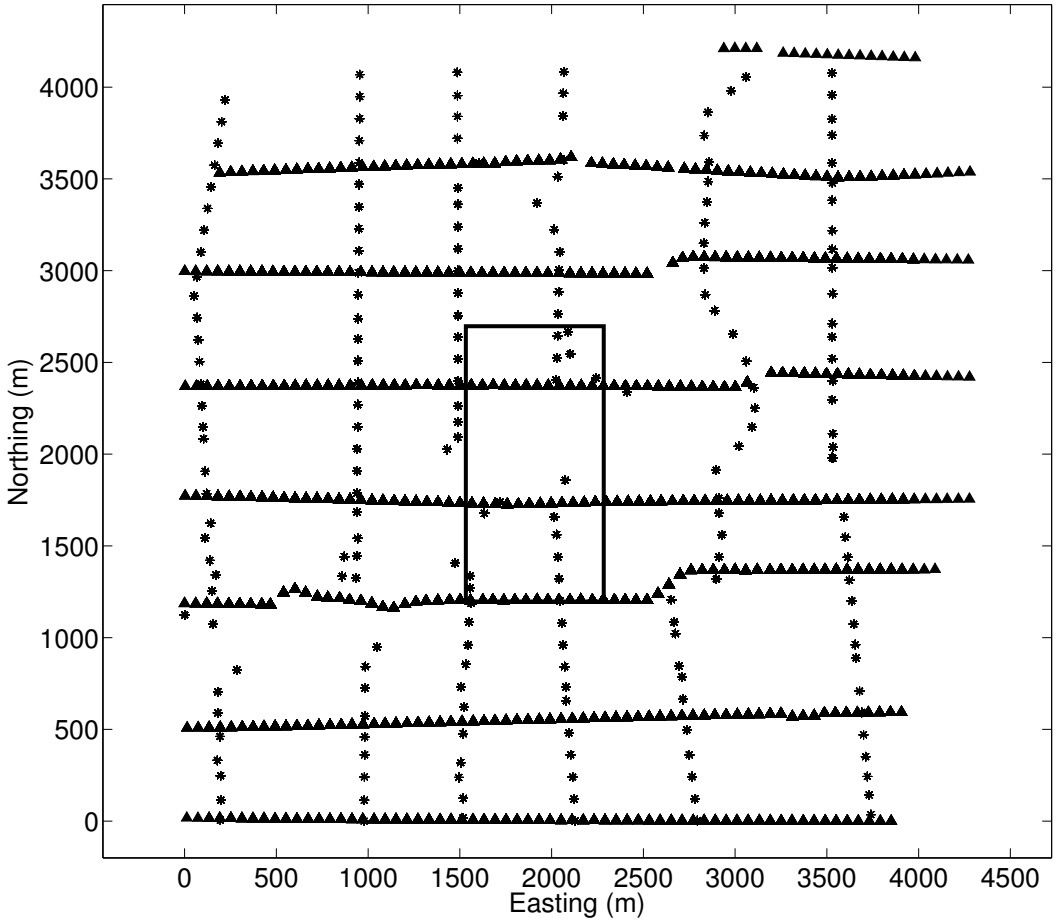



Figure 3.13: Distribution of sources and receivers for an orthogonal survey in $x, y$ coordinates. The figure is given in true aspect ratio. The area in the center of the patch highlights the midpoints that are reconstructed using this distribution of sources and receivers. Triangles represent the receivers and stars represent the sources.

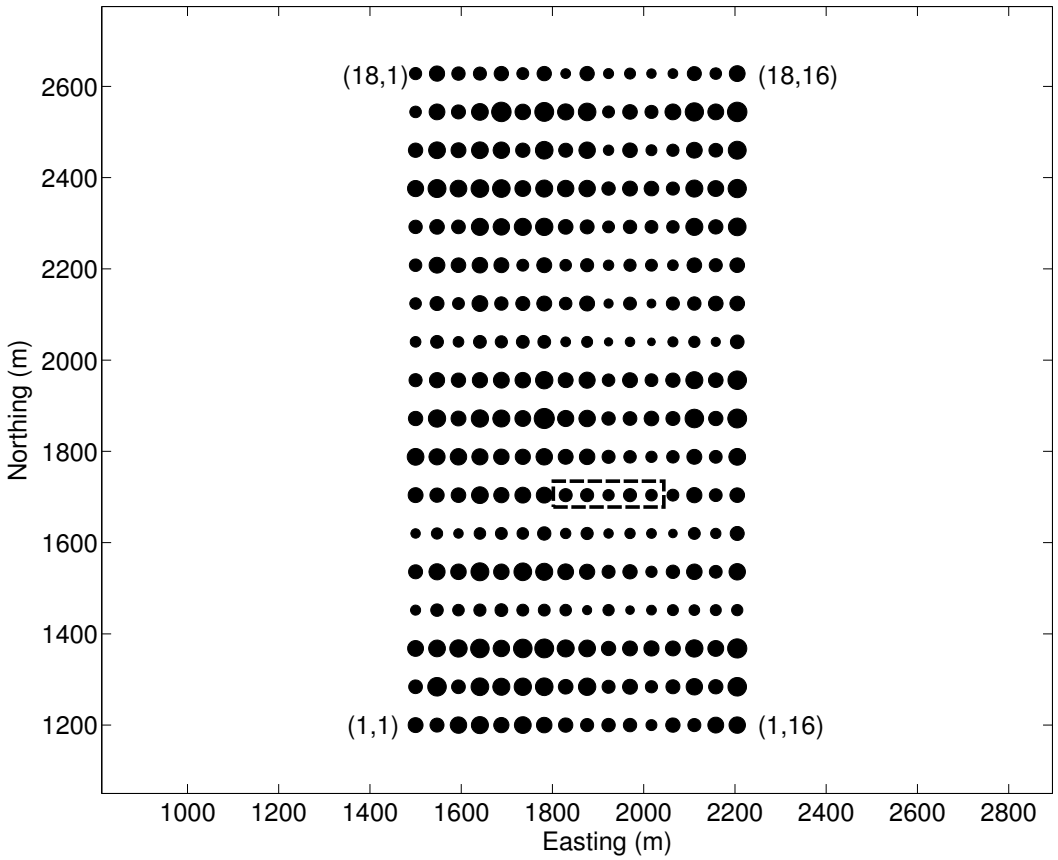



Figure 3.14: Fold map corresponding to the rectangle in Figure 3.13. The figure is given in true aspect ratio. The area of the circles is proportional to the fold. The dashed line rectangle indicates the position of the data shown in Figure 3.15. Numbers in brackets are CMP numbers corresponding to the corners of the rectangle.

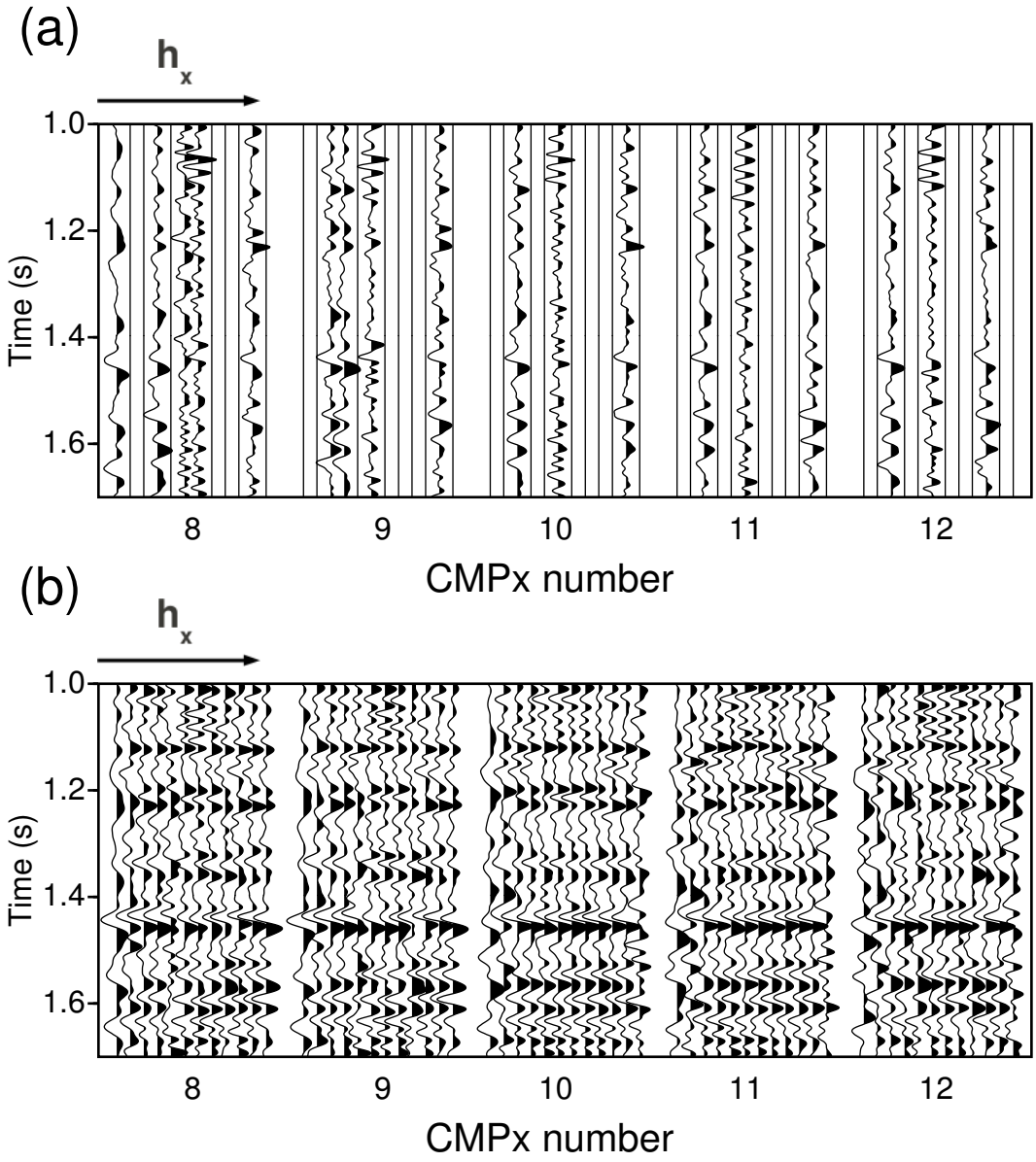



Figure 3.15: Common-midpoint gathers, a portion of the data, associated to the acquisition patch shown in Figure 3.13. The position of these gathers is indicated by the dashed line rectangle on Figure 3.14. (a) Observations prior to reconstruction. (b) Reconstructed data via tensor completion.

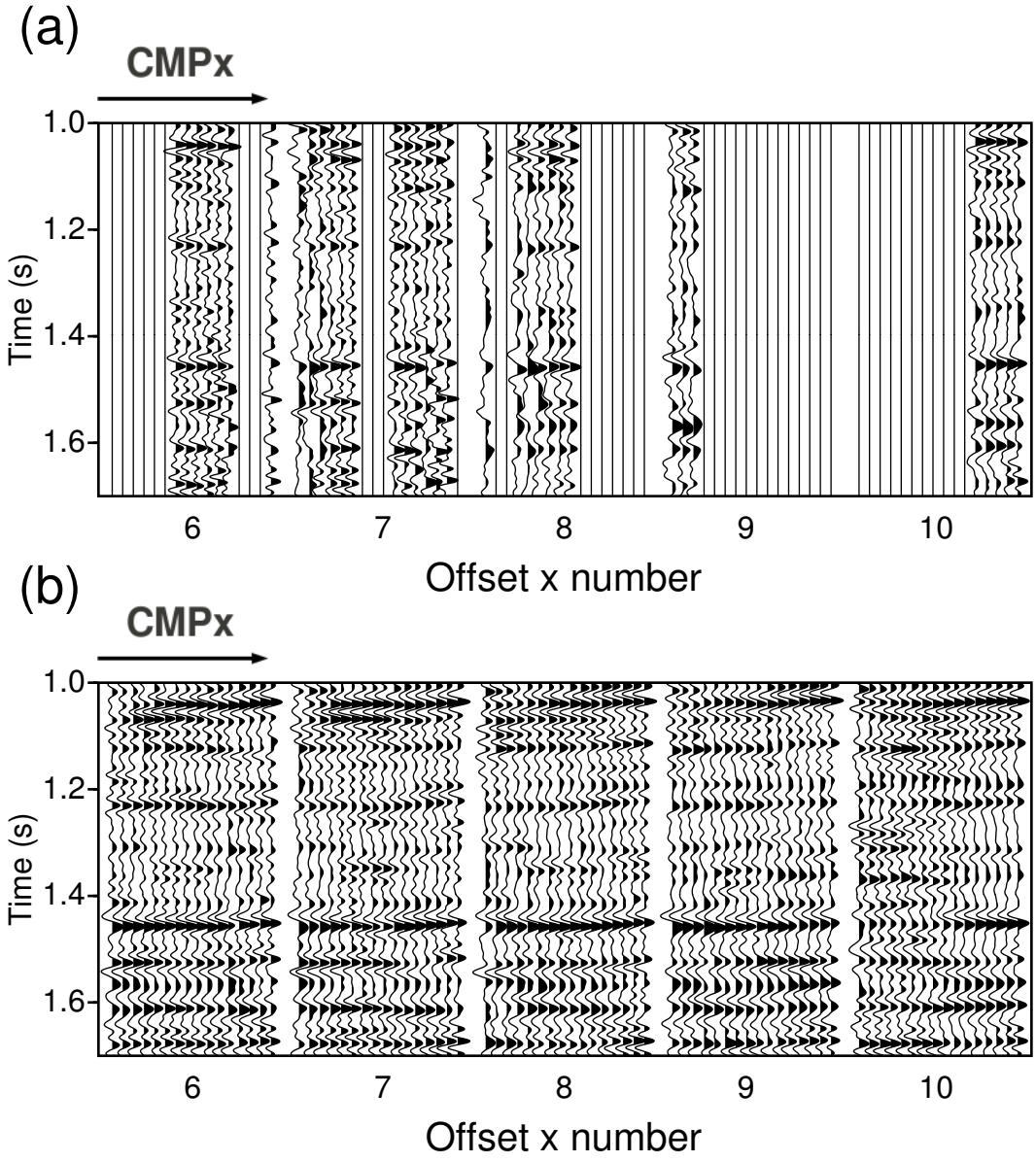



Figure 3.16: Common-offset gathers, portion of the data associated to the acquisition patch shown in Figure 3.13. (a) Observations prior to reconstruction. (b) Reconstructed data via tensor completion.

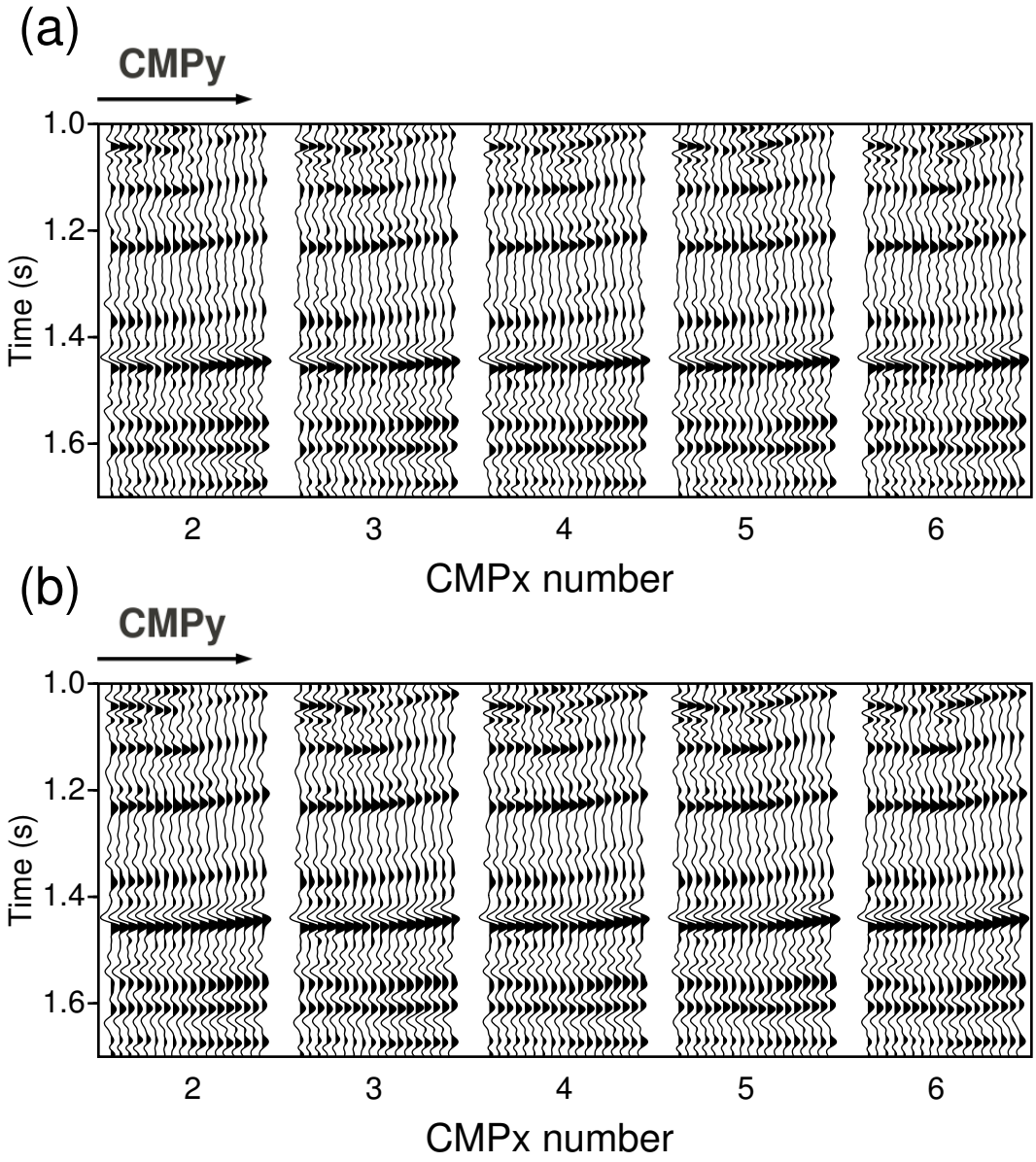



Figure 3.17: Portion of the stack associated to the acquisition patch shown in Figure 3.13. (a) Stack prior to reconstruction. (b) Stack after reconstruction.

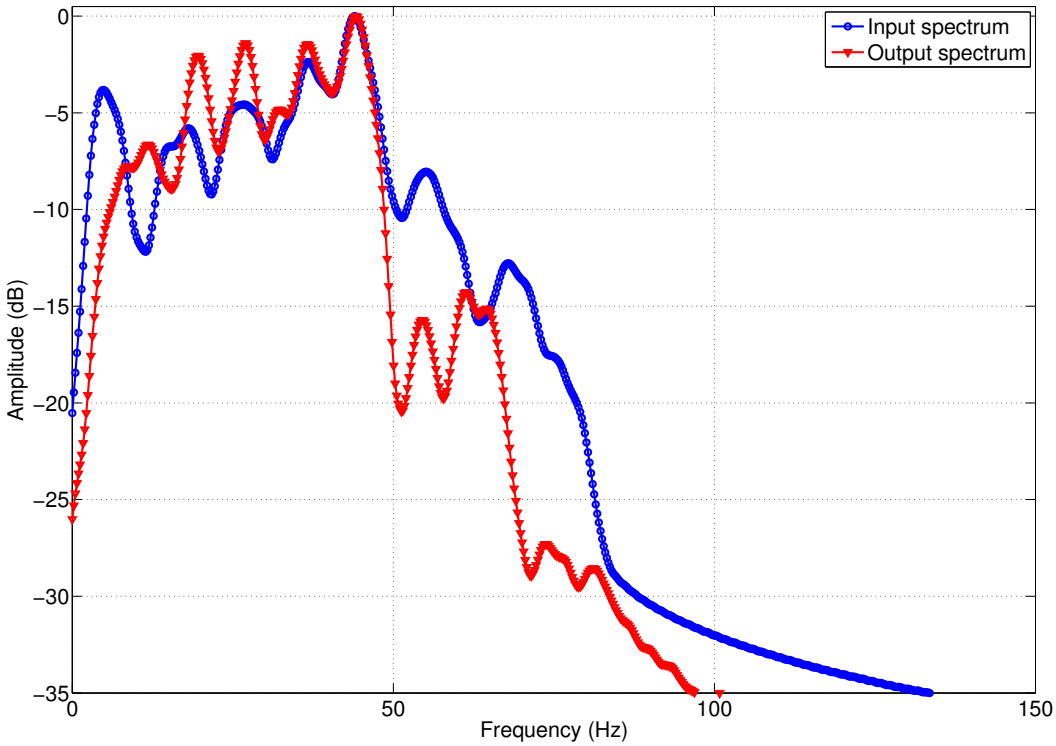



Figure 3.18: Spectrum of a common-offset gather (offset-$x$ = offset-$y$ = 6) for CMP-$x = 1$ before and after reconstruction.

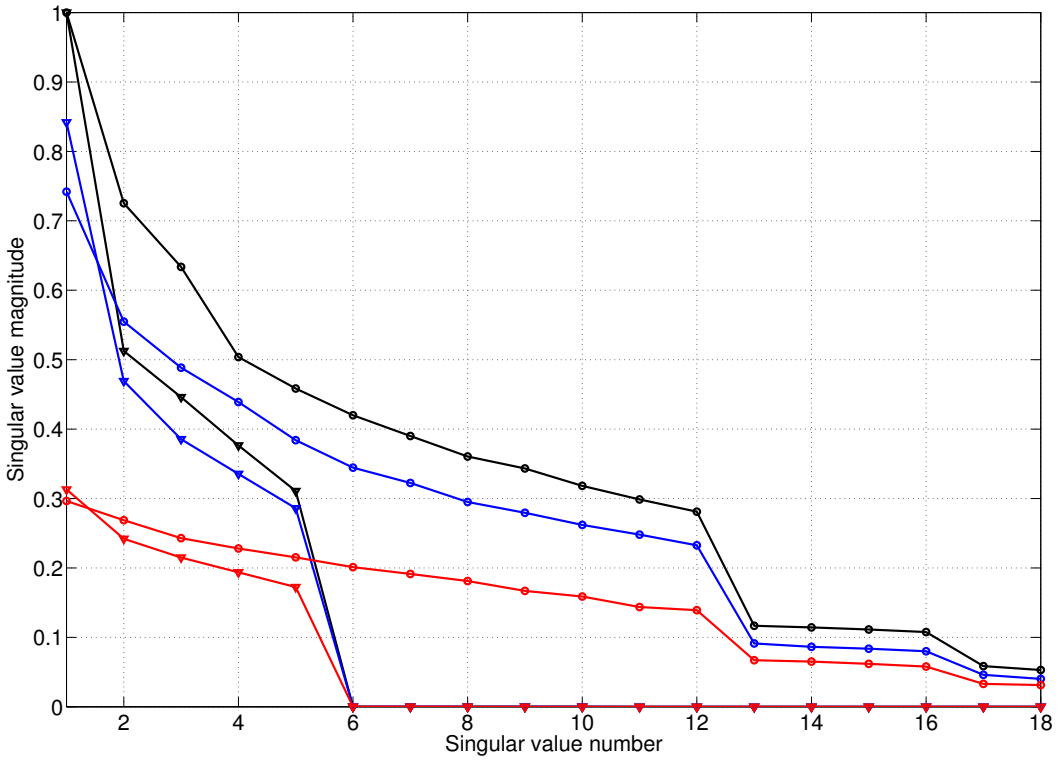



Figure 3.19: Distribution of singular values, averaged over all modes, before and after rank reduction for three frequencies. The black, blue and red lines represent low, mid and high frequencies, respectively. The circular and triangular icons symbolize before and after rank reduction, respectively.

A synthetic simulation was utilized to gain understanding of the limits of the proposed tensor completion strategy. In particular, we found that the method is robust to curvature, additive noise and can tolerate high levels of data sparsity. The algorithm is controlled by a series of parameters: size of the core-tensor, maximum number of iterations, and a scalar $a$ that permits to add denoising capabilities to the reconstruction process. Furthermore, we verified with synthetics that 4D spatial volumes lead to more accurate reconstructions than those with 2D and 3D spatial volumes.

Reconstruction of prestack data is an underdetermined inverse problem. Any method to solve the reconstruction problem needs to make a hypothesis about the data in order to estimate realistic reconstructions. Techniques like MWNI (Liu, 2004) or Fourier interpolation via sparse inversion (Sacchi et al., 1998; Zwartjes and Gisolf, 2007), for example, impose a constraint in the Fourier spectrum of the data to deal with this indeterminacy. Here we assume that the ideal data can be represented via a low-rank tensor in the $F-X$ domain. Cadzow (or MSSA) filtering and reconstruction (Trickett and Burroughs, 2009; Trickett et al., 2010) makes the same assumption. However, the Cadzow method utilizes Hankel matrices in its calculations. In contrast, the low-rank approximation with HOSVD takes full advantage of the tensor structure. Our synthetics show that the presented method can reconstruct curved events.

It must be stressed that the approach presented in this chapter cannot handle spatially aliased data, but only irregularly missing traces in a regular grid. There is no constraint imposed in the spectrum nor a prediction made from the non-aliased low frequencies as in Spitz (1991) or Naghizadeh and Sacchi (2007, 2010a). Therefore, with the proposed method, both the aliased and non-aliased data are recovered with the low-rank assumption.

Most reconstruction methods in the literature require windowing to be successful. The use of windows allow us to assume locally linear events. The linearity assumption is one of the most common assumptions made by reconstruction techniques. Our experience with the synthetic simulations showed that tensor completion with HOSVD is not affected by the curvature of the events.

Our experiments with the core-tensor size reveal that when the level of the noise is low, a core-tensor size that is slightly larger than the number of events in the data permits to recover the data with a good quality. When there is a larger level of noise present, we might want to adopt a core-tensor size that is around the amount of events in the data. It is preferable to over-estimate the size of the core-tensor rather that using one that is too small. However, noisy data with missing traces need a smaller size (or lower rank) for the core-tensor than when the data have low level of noise.

Real data from the Western Canadian Sedimentary Basin with a very sparse geometry were also tested. The running time for this patch was comparable to the time needed by our

Fourier reconstruction algorithm. The stack section present a similar aspect before and after applying tensor completion. The latter confirms that the method is not introducing artifacts in the reconstructed data.

# CHAPTER 4

# Reconstruction via sequential SVD [1]

## 4.1 Introduction

In the previous chapter we introduced a simultaneous reconstruction and noise attenuation strategy that used the HOSVD on the frequency slices of prestack data to perform rank reduction. In this chapter we investigate a new method that uses the unfolding of the 4D frequency slices to perform sequential rank reduction and we will call it SEQ-SVD. The only difference between the algorithm presented in the previous chapter and the algorithm presented in this chapter is the rank reduction "engine". Although simple in nature, in practice the procedure suggested in this chapter proves to be comparable to the HOSVD. As in the previous chapter, this algorithm leads to a reconstruction algorithm where the multidimensional seismic data does not need to be embedded in multi-level Hankel matrices (Trickett et al., 2010; Oropeza and Sacchi, 2011).

## 4.2 Theory

Consider a seismic volume with four spatial coordinates $D(t, x_1, x_2, x_3, x_4)$. The DFT can be used to transform the volume to the $f-x_1-x_2-x_3-x_4$ domain. For each temporal frequency $f$, this volume can be arranged into a fourth-order tensor that depends on $x_1, x_2, x_3, x_4$. The spatial coordinates can correspond to source-receiver, inline-crossline-offset (as in Chapter 3) or inline-crossline-offset-azimuth (refer to Chapter 1).

Assuming that each one of the unfolded matrices obtained from the tensor is a low-rank structure (Appendix A), we propose an algorithm that applies rank reduction sequentially

[1] A version of this chapter has been published in The Recorder, Kreimer and Sacchi (2012a).

to each unfolded tensor (Table 4.1). The tensor is unfolded in its first mode, followed by a shrinkage of its singular values (the user specifies the amount of shrinkage) and folding to a tensor structure. Next, it continues with its second mode in the same way as with the first mode and continues until the shrinkage has been applied to all the modes. Notice that the shrinkage of the singular values is not implemented in one mode only, but in all four modes sequentially. We tested the possibility of doing the shrinking in all modes but one, in only one mode, etc, but the results from these tests were not as successful as the ones obtained via the algorithm in Table 4.1.

| |
|---|
| **Input** fourth-order tensor $\boldsymbol{\mathcal{D}}$ |
| Initialize with $\tilde{\boldsymbol{\mathcal{D}}} = \boldsymbol{\mathcal{D}}$ |
| **for** $i = 1$ to 4 |
| $\mathbf{D}^{(n)} = \text{unfold}\left(\tilde{\boldsymbol{\mathcal{D}}}\right)$ |
| $[\mathbf{U}, \boldsymbol{\Sigma}, \mathbf{V}] = \text{SVD}\left(\mathbf{D}^{(n)}\right)$ |
| $\mathbf{D}_r^{(n)} = \mathbf{U}_r \, \boldsymbol{\Sigma}_r \, \mathbf{V}_r^{\text{H}}$ |
| $\tilde{\boldsymbol{\mathcal{D}}} = \text{fold}\left(\mathbf{D}_r^{(n)}\right)$ |
| **end** |
| **Output** $\tilde{\boldsymbol{\mathcal{D}}}$ is the low-rank tensor |

Table 4.1: Sequential rank reduction on the unfoldings of tensor $\boldsymbol{\mathcal{D}}$.

The abovementioned algorithm can be expressed in operator form via $\tilde{\boldsymbol{\mathcal{D}}} = \mathcal{R}(\boldsymbol{\mathcal{D}})$ and we will call it SEQ-SVD. The rank $r$ is the desired rank and it is possible for the $n$-ranks to be different for each unfolding. Unfold and fold are the operations of creating a matrix from a tensor and a tensor from a matrix, respectively (refer to Chapter 2).

This procedure performs denoising of multidimensional seismic data in a similar way $F - X - Y$ eigenimage filtering (Trickett, 2003) implements denoising of data that depend on two spatial dimensions. While the tensor rank reduction in our algorithm is implemented via shrinking, Tomioka et al. (2011) suggest to minimize the nuclear norm of one particular unfolding, among other strategies for rank reduction for tensors. Clearly, the nuclear norm minimization could be applied iteratively for each mode, replacing the shrinkage we use in algorithm 4.1. Refer to Chapter 5 for definition of nuclear norm.

The algorithm in Table 4.1 consists of applying a hard-thresholding operation to the singular values of the matrix unfoldings $\mathbf{D}^{(n)}$. Methods developed for matrix completion are based

on a hard-thresholding operator when trying to address the non-convex problem

$$\text{minimize} \quad \text{rank}\,(\mathbf{D}^{(n)}) \quad \text{subject to} \quad \mathbf{T}^{(n)}\mathbf{D}^{(n)} = \mathbf{D}^{(n),obs}, \tag{4.1}$$

where $\mathbf{T}^{(n)}$ is a matricized version of the sampling operator $\boldsymbol{\mathcal{T}}$ and $\mathbf{D}^{(n),obs}$ are the observations in matrix format. Jain et al. (2010) and Goldfarb and Ma (2011) developed methods to solve this problem, known as Singular Value Projection (SVP) and Iterative Hard Thresholding (IHT) algorithms, respectively, and presented recovery conditions for these methods. The update given in equation 4.2 (for $a = 1$) added to the rank reduction algorithm given in Table 4.1 is very similar to the methods proposed by the aforementioned authors, with the exception that they focus on the matrix completion problem, while we deal with tensors. However, if we split our problem in matrix completion problems for each mode $n$, we can find a good analogy between them.

The rank-minimization problem is non-convex and NP-hard and thus no tractable algorithm solves the problem unless we introduce some relaxation or constraint into the problem. For instance, setting an optimum bound for the amount of missing elements is a possible constraint. Jain et al. (2010) and Goldfarb and Ma (2011) introduced a constraint in the form of the sampling operator. Their sampling operator satisfies the restricted isometry property, popular in compressive sensing. It is interesting to notice that the formula in equation 4.2 for $a = 1$ can be interpreted as a gradient update of $\mathbf{D}^{(n)}$, or the gradient of the misfit $\|\mathbf{T}^{(n)}\mathbf{D}^{(n)} - \mathbf{D}^{(n),obs}\|_F^2$ (Goldfarb and Ma, 2011). According to Tanner and Wei (2012), most matrix completion algorithms are iterative and based on updating $\mathbf{D}^{(n)}$ to decrease the misfit measure, which is achieved by updating it in the direction of the negative gradient of the misfit. This is the same as in equation 4.2 for $a = 1$ and considering tensors instead of matrices. We will later focus on the convex relaxation of problem 4.1 for tensors in Chapter 5.

### 4.2.1 A reconstruction algorithm

The rank reduction iterative algorithm adopted in the previous chapter for simultaneous denoising an reconstruction is given by

$$\boldsymbol{\mathcal{D}}^k = a\,\boldsymbol{\mathcal{D}}^{obs} + (1 - a\boldsymbol{\mathcal{T}})\,\mathcal{R}(\boldsymbol{\mathcal{D}}^{k-1}), \tag{4.2}$$

where $a$ is a parameter between 0 and 1 that controls the level of reinsertion of noisy observations. As in the previous chapter, the parameter $a$ is equal to 1 when trying to reconstruct data with a high $SNR$. The operator $\boldsymbol{\mathcal{T}}$ is the sampling operator with the same dimensions of the data, filled with zeros in the bins with missing traces and ones in the bins

containing samples. The index $k$ indicates iteration number. The operator $\mathcal{R}$ symbolizes the rank reduction operation. In this case, it is equivalent to rank reduction via the SEQ-SVD. This algorithm is similar to reconstruction via Projection Onto Convex Sets (POCS). However, we notice that the POCS algorithm is implemented via an iterative amplitude thresholding process (Abma and Kabir, 2006), whereas expression 4.2 applies tensor rank reduction in each iteration.

## 4.3 Examples

### 4.3.1 Synthetic data

To demonstrate in practice that unfolded tensors are a low-rank structure we design a volume of size $128 \times 12 \times 12 \times 12 \times 12$ that contains three plane waves (for a theoretical proof, refer to Appendix A). Figure 4.1 displays the distribution of singular values averaged over the four unfoldings. The line with circular icons belongs to the fully sampled data with $SNR = 100$. Clearly, only a few singular values are different from zero. The line with asterisk icons corresponds to the case where the data were decimated and contaminated with noise ($SNR = 1$). The spectrum of singular values shows three dominant components immersed in smaller ones that model the noise subspace. The line with triangular icons represents the singular value distribution of the reconstructed data, which follows closely the distribution of the fully sampled data.

Figure 4.2 shows the result of applying the rank reduction procedure explained in the previous section. We have kept the first three singular values in each unfolding and used a parameter $a = 0.3$. Column (a) in the figure is the non-decimated data prior to noise contamination, the column (b) is the decimated data after addition of noise ($SNR = 1$). The latter is also the input to our algorithm. Column (c) is the reconstructed and denoised data and column (d) is the difference between the first and third column. The reconstructed and noise attenuated volume has negligible errors. One can confirm that SEQ-SVD is able to recover the events with a high degree of accuracy. The reconstructed volume has a Frobenius norm that is about 32 times the Frobenius norm of the reconstruction error ($Q = 33$ dB). As a comparison, we also use the truncated HOSVD from the previous chapter (Kreimer and Sacchi, 2012d). The results are very similar to those obtained with the proposed algorithm. Additionally, we show five consecutive CMP gathers in Figure 4.3.

We also examine the spectrum of singular values averaged over all unfoldings for a data set composed of three curved events with 50% decimation and contaminated with noise. Figure 4.4 displays the distribution of singular values for this exercise, averaged over the four unfoldings. The line with circular icons belongs to the fully sampled data with $SNR = 100$.

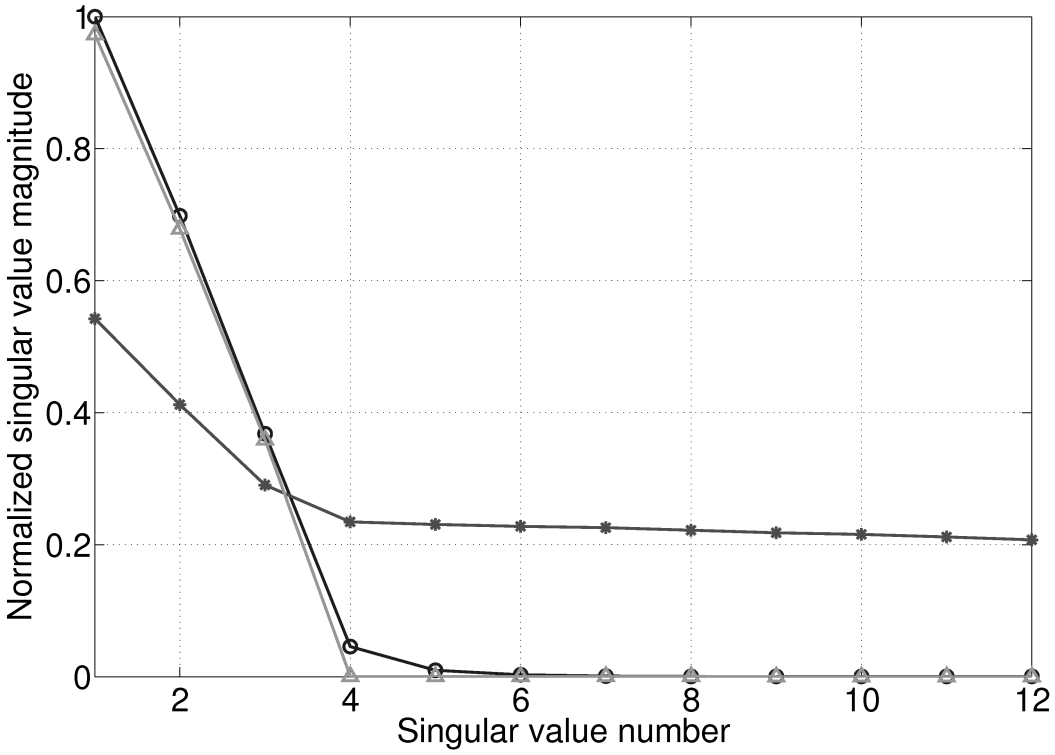



Figure 4.1: Distribution of singular values averaged over all four unfoldings, for one frequency. Case of data with linear events, 50% missing traces.

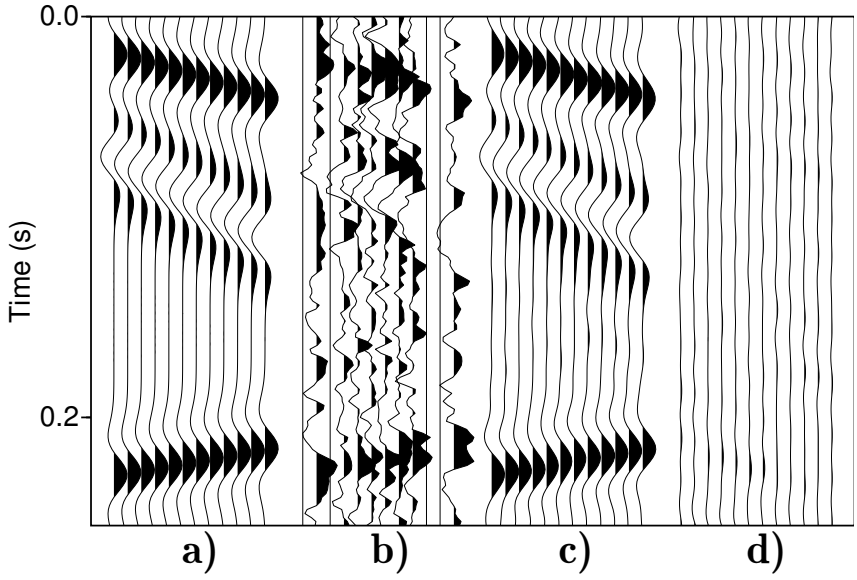



Figure 4.2: Reconstruction and noise attenuation for a 5D seismic volume with three linear events and $SNR = 1$. Only a subset of the volume is displayed in the figure.

The spectra of singular values does not show an abrupt change in amplitude; a feature that can be used to reveal if the underlying data are a low-rank structure. The line with asterisk icons corresponds to the case where the data were decimated and contaminated with noise ($SNR = 1$). The line with triangular icons represents the singular value distribution of the reconstructed data. The quality of reconstruction slightly degrades with curvature. However, the reconstruction results are within an acceptable level of accuracy. The reconstructed volume has a Frobenius norm that is about 19 times the Frobenius norm of the reconstruction error ($Q = 29$ dB). Figure 4.5 highlights four CMPs prior to decimation and after reconstruction.

As we did in Chapter 3, we perform a sensitivity experiment in which we tried different values for $a$. A priori, we cannot assume that the same reinsertion parameter that gave satisfactory results for HOSVD will have the same effect in the method presented in this chapter. The synthetic model used in this test coincides with the one used in the same experiment in Chapter 3. Figure 4.6 shows the results. Overall, we observe the same sensitivity to the change of $a$ as in the case of HOSVD. Between 0.4 and 0.9, we see very little difference between different choices of the parameter $a$. As in the case of HOSVD, $a = 0.9$ was the value that gave better results with real data and we utilized this value for all our implementations.

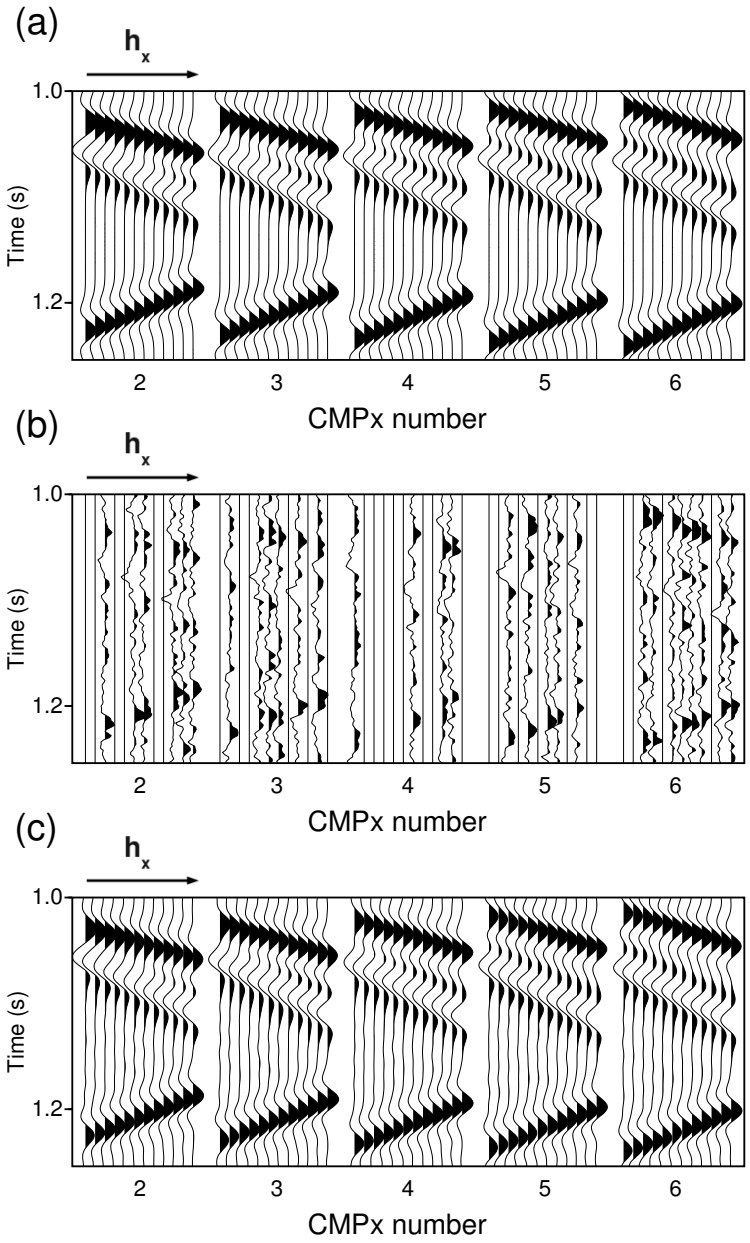



Figure 4.3: Reconstruction and noise attenuation of a 5D seismic volume with three linear events and $SNR = 1$. Only five CMPs are displayed in the figure. a) Ideal noise-free and fully sampled data. b) Decimated and noisy data. c) Reconstructed and noise attenuated data.

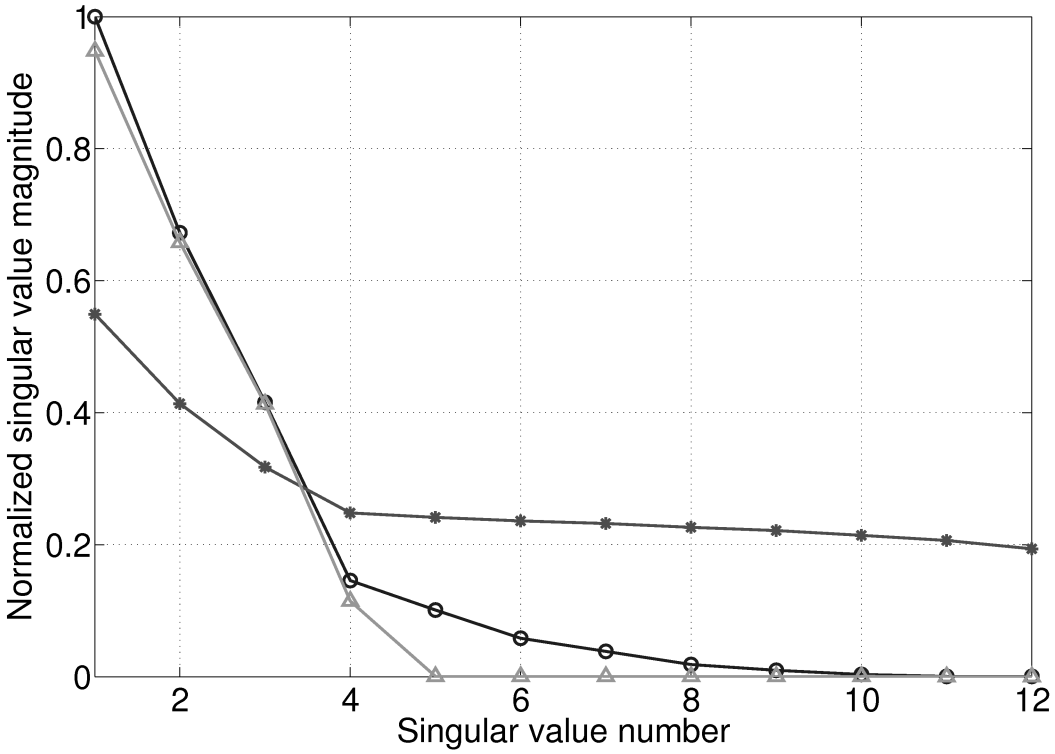



Figure 4.4: Distribution of singular values averaged over all four unfoldings, for one frequency. Case of data with curved events, 50% missing traces.

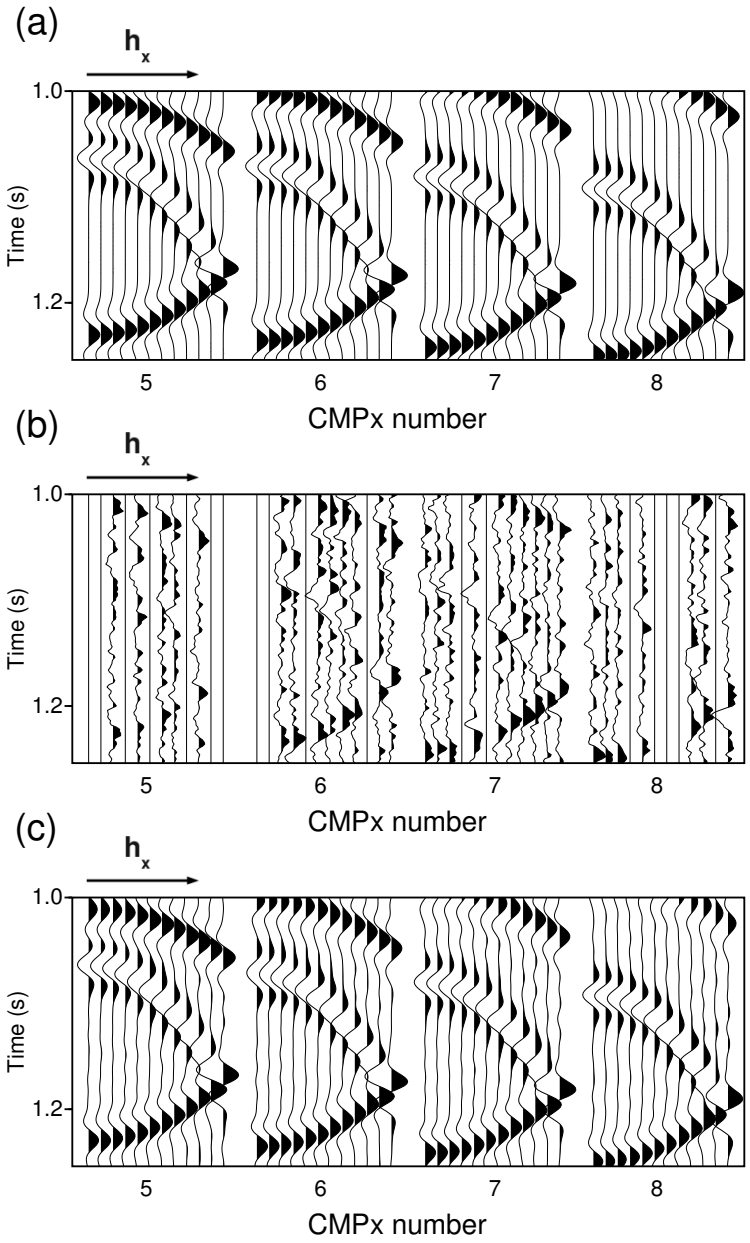



Figure 4.5: Reconstruction and noise attenuation of a 5D seismic volume with three curved events and $SNR = 1$. Only four CMPs are displayed in the figure. a) Ideal noise-free and fully sampled data. b) Decimated and noisy data. c) Reconstructed and noise attenuated data.

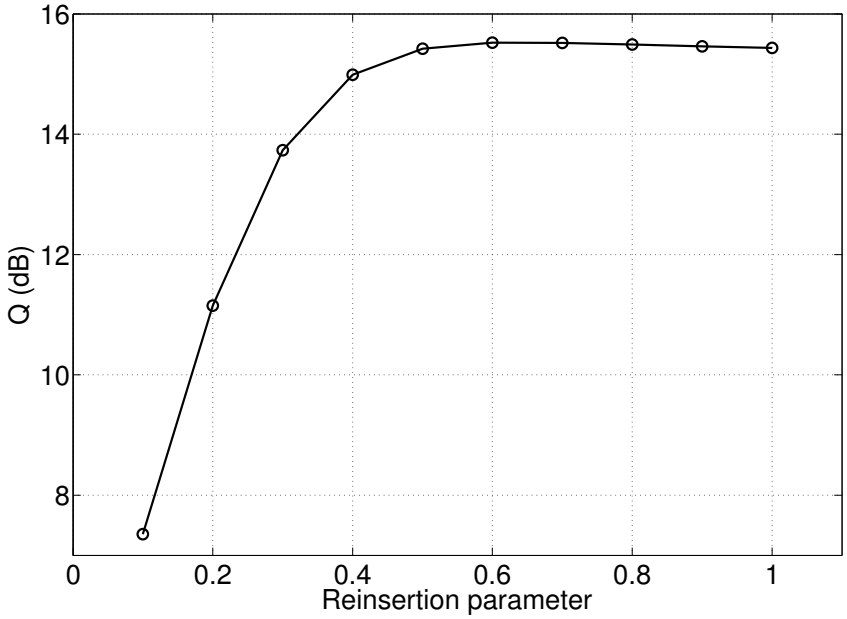



Figure 4.6: Sensitivity of $Q$ to the change of the reinsertion parameter $a$.

### 4.3.2 Real data

Our real data example is from the Western Canadian Basin. The reconstruction is carried out in the inline/crossline midpoint-offset domain, with a grid of size $16 \times 18 \times 12 \times 12$ (same grid as that used in the real data example in Chapter 3). The input is NMO corrected and contains 16060 traces, populating 39% of the grid. Figures 4.7 and 4.8 show five CMP and offset gathers prior and after reconstruction, respectively. Comparing these figures to Figures 3.15 and 3.16 in Chapter 3, we notice that the results obtained with SEQ-SVD are comparable to those of HOSVD. To assess the presence of any damage to the data during the reconstruction process, we present five stacked CMPs in Figure 4.9. As expected, we can observe some level of noise attenuation in the stacked data after the reconstruction. To confirm that the reconstruction process has not damaged the data, we plot the spectrum of a CMP gather and a fixed inline. The total running time is approximately one minute on a single processor Intel Xeon(R) running at 3.07 Ghz using MATLAB.

## 4.4 Conclusions

We have presented an application of the singular value decomposition for rank reduction of a tensor (SEQ-SVD). The algorithm operates directly on matrices obtained by unfolding

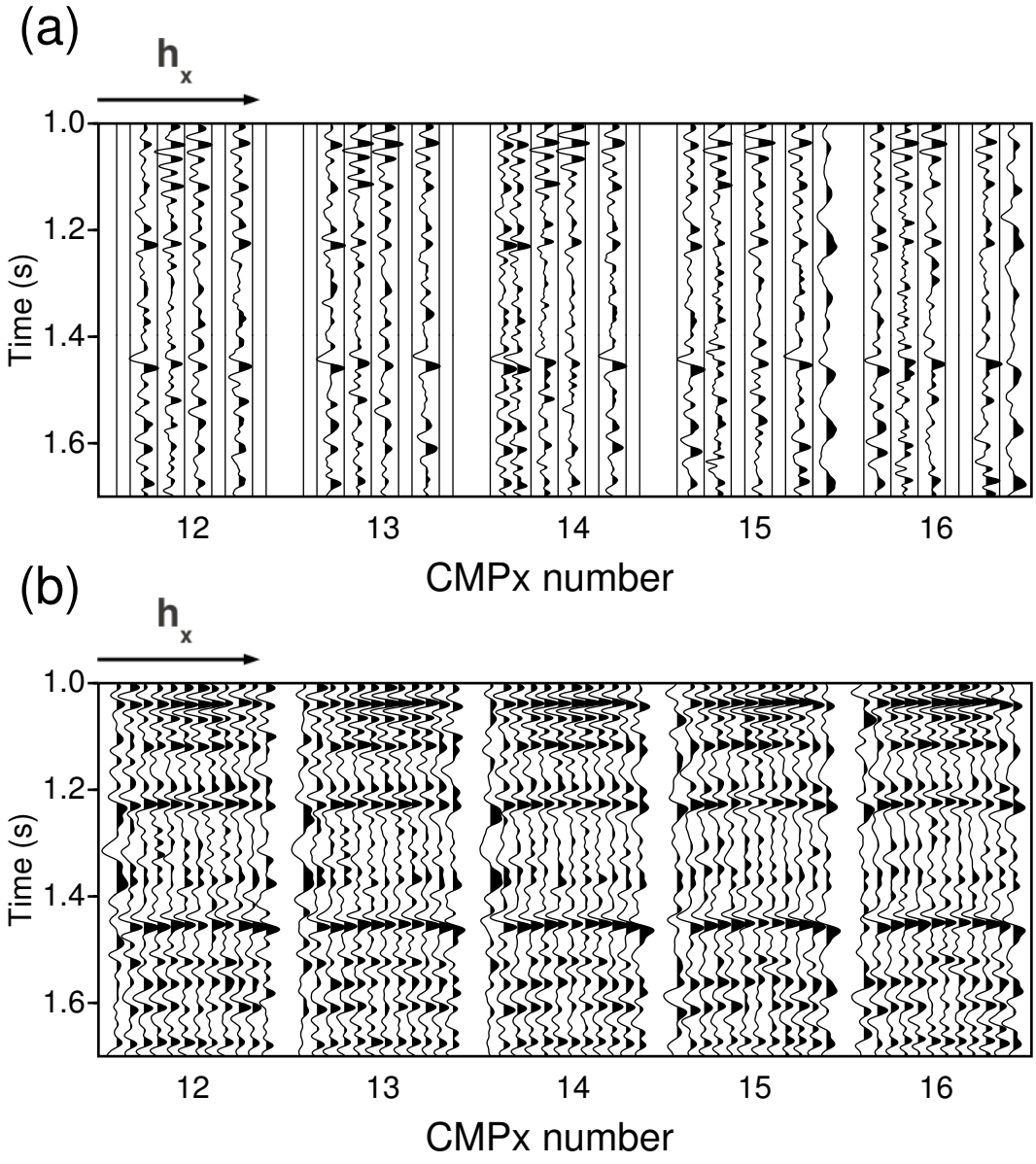



Figure 4.7: WCB data. CMP gathers for the 9th CMPx bin and the 4th offset-y bin. a) Input to the reconstruction algorithm, b) Output. Only five CMP gathers are shown in this figure.

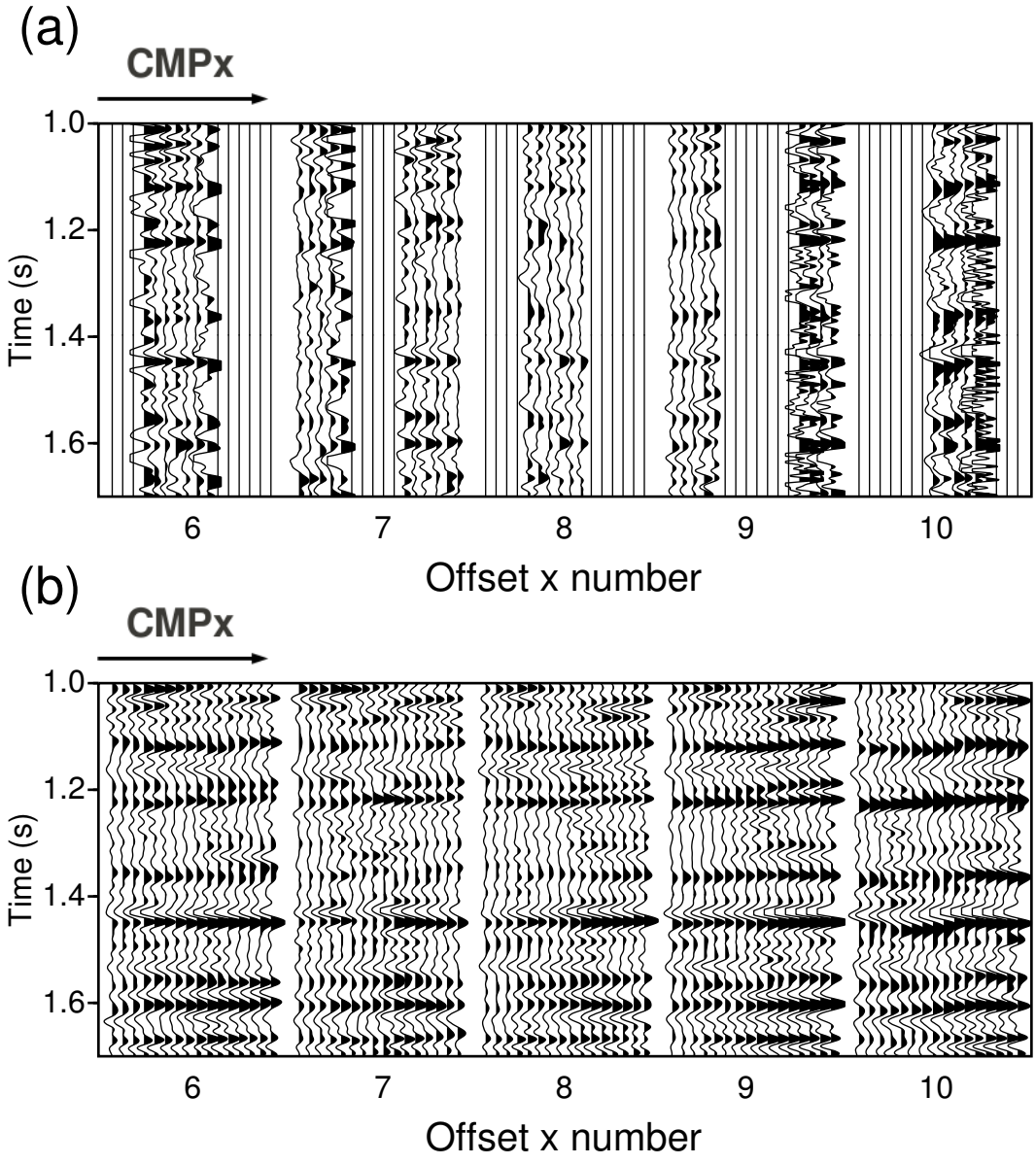



Figure 4.8: WCB data. Offset gathers for the 8th CMPy bin and the 1st offset-y bin. a) Input to the reconstruction algorithm, b) Output. Only five offset gathers are shown in this figure.

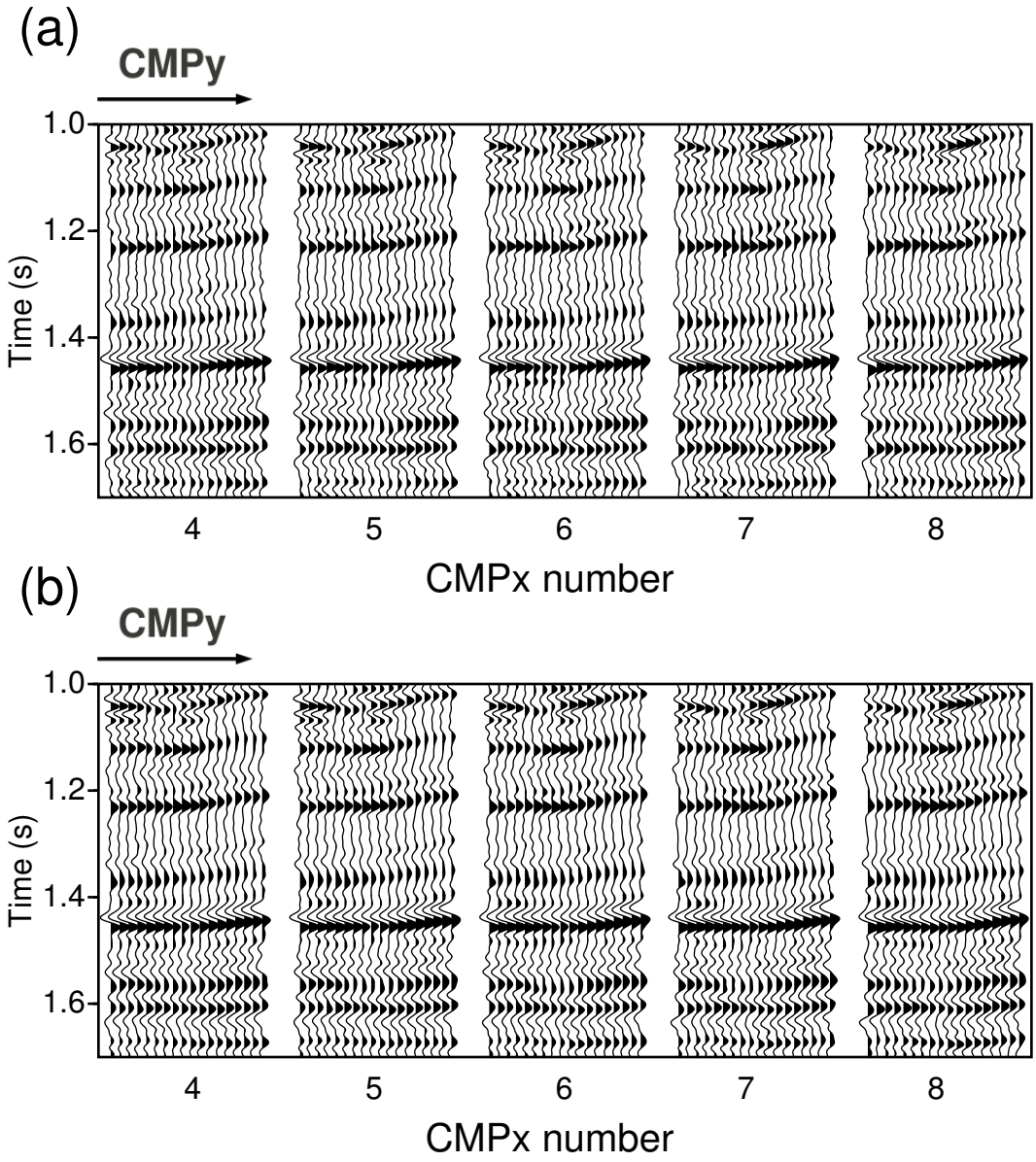



Figure 4.9: Stacks prior and after reconstruction. Only five CMPs are displayed.

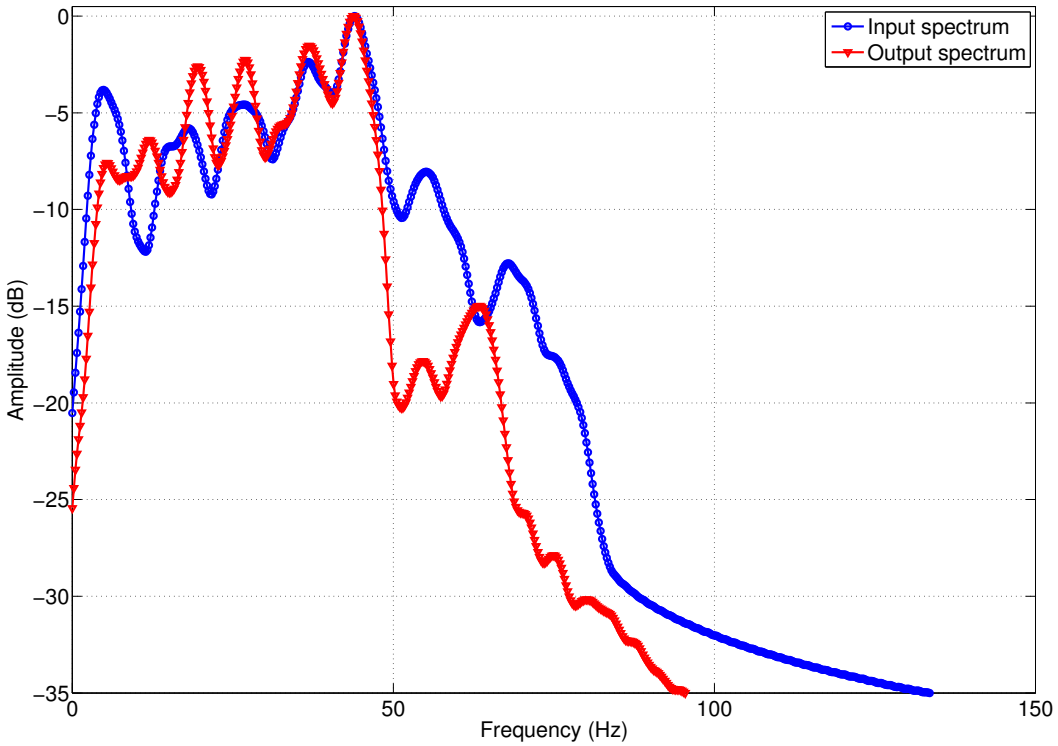



Figure 4.10: Spectrum of a common-offset gather (offset-$x$ = offset-$y$ = 6) for CMP-$x = 1$ before and after reconstruction.

the tensor that represents the prestack seismic data in the frequency-space domain. The proposed method applies an iterative rank reduction that permits to denoise and reconstruct seismic data. We made an interesting connection between SEQ-SVD for the noise-free case and the problem of minimizing the non-convex function containing the rank of the unfolded tensor subject to the observations. The shrinkage of the singular values that we use in SEQ-SVD can be considered to be a hard thresholding operation. The importance of this will be better understood in the next chapter.

As we observed in the previous chapter, this method is robust to curvature in the reflections as well as low levels of signal-to-noise ratio. The sensitivity test proved that the SEQ-SVD is not very sensitive to the choice of the reinsertion parameter for values larger than 0.4. The implementation in a real dataset from the Western Canadian Basin showed satisfactory results, similarly to those obtained with synthetic datasets.

Although the iterative procedure presented in this chapter is simpler than HOSVD-based reconstruction, it is surprising to notice that both methods give comparable results in practice. We present a more thorough comparison with other methods in Chapter 6.

# CHAPTER 5

# Nuclear norm reconstruction [1]

## 5.1 Introduction

In this chapter, we venture into the problem of tensor completion from a completely different perspective than in Chapters 3 and 4. We still make the same assumption as before: the fully sampled volume is a low-rank tensor in $F - X$. As we mentioned before, noise and missing traces increase the rank of the tensor. Consequently, the reconstruction problem is posed as a low-rank tensor completion problem that we solve via convex optimization. We base our reconstruction on an algorithm for tensor completion proposed by several authors (Gandy et al., 2011; Signoretto et al., 2010; Tomioka et al., 2011). The completion problem entails finding the fully sampled data tensor with minimum nuclear norm. The nuclear norm is the sum of the singular values of the tensor across all its unfoldings and plays a role similar to that of the $\ell_1$ norm in compressive sensing (Candès and Recht, 2009; Candès and Plan, 2010). In fact, when a matrix is diagonal, minimizing its nuclear norm is equivalent to minimizing its $\ell_1$ norm, namely finding the sparsest approximation (Fazel et al., 2001).

The algorithm used for the optimization is the alternating direction method of multipliers (ADMM), a modification of the more classical method of multipliers (Hestenes, 1969; Powell, 1969). The algorithm iteratively estimates the low-rank tensor in the $F - X$ domain that fits the data up to the level of the noise and fills the missing entries, while forcing the final tensor unfoldings to have minimum nuclear norm (a solution dominated by a few singular values).

[1] Parts of this chapter have been published in Kreimer and Sacchi (2012c) and Kreimer and Sacchi (2013).

## 5.2 Theory

### 5.2.1 The tensor completion problem

Prestack seismic data in the $F-X$ domain form a 4D structure. We map the prestack seismic volume $\boldsymbol{\mathcal{D}}^{obs}$ to a regular grid by binning. For instance, $D^{obs}_{ijkl}(\omega)$ represents a trace from the volume $\boldsymbol{\mathcal{D}}^{obs}$ in the bin $ijkl$, where $ijkl$ can represent binned inline, crossline, offset and azimuth coordinates. We will drop the reference to $\omega$ and we will refer to a 4D frequency slice by $\boldsymbol{\mathcal{D}}^{obs}$. Naturally, some bins of $\boldsymbol{\mathcal{D}}^{obs}$ do not contain traces and are zero. The fully sampled ideal volume is called $\boldsymbol{\mathcal{D}}$. We make the assumption that this volume $\boldsymbol{\mathcal{D}}$, with all its samples, is such that its $n$-ranks are small. If the volume contains missing traces and/or random noise, the $n$-ranks of the tensor increase. This is demonstrated in Figure 5.5 in the section devoted to examples. Under this assumption, the problem of recovering missing traces is equivalent to minimizing an objective function that contains a measure of the misfit between the original observations and the reconstructed volume, and a measure of the rank of the tensor unfoldings. In mathematical terms, this is equivalent to solving the following problem

$$\text{minimize} \quad \sum_{i=1}^{4} \text{rank}(\mathbf{D}^{(i)}) \quad \text{subject to} \quad \mathcal{T}\boldsymbol{\mathcal{D}} = \boldsymbol{d}^{obs}, \tag{5.1}$$

where $\mathbf{D}^{(i)}$ are the mode-$i$ matrix unfoldings of the tensor $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{d}^{obs}$ is the vector of observations, both in the $F-X$ domain. The operator $\mathcal{T}$ is the retrieval operator that extracts the observations from tensor $\boldsymbol{\mathcal{D}}^{obs}$ and maps from $\mathbb{C}^{I_1\times I_2\times I_3\times I_4} \rightarrow \mathbb{C}^m$, $m$ being the amount of observed traces (notice that the retrieval operator here is not equal to the sampling operator defined in Chapters 3 and 4). Problem 5.1 is often referred to as the tensor completion problem. This problem is non-convex and NP-hard (Fazel et al., 2001; Candès and Recht, 2009), as defined in Chapter 1. Fazel et al. (2001) approximated the solution by replacing the rank of a matrix $\mathbf{A}$ by the nuclear norm of this matrix, defined as $\|\mathbf{A}\|_* = \sum_{i=1}^{n} \sigma_i$, where the $\sigma_i$'s are the singular values of the matrix $\mathbf{A}$. The latter leads to a tractable convex optimization problem. The objective function to minimize becomes

$$\text{minimize} \quad J \;=\; \sum_{i=1}^{4} \|\mathbf{D}^{(i)}\|_* \;+ \frac{\lambda}{2}\, \|\mathcal{T}\boldsymbol{\mathcal{D}} - \boldsymbol{d}^{obs}\|_F^2, \tag{5.2}$$

where $\lambda$ is a trade-off parameter that balances the terms of the objective function $J$. Notice that the nuclear norm for tensors is defined as the sum of the nuclear norm of the four matrix unfoldings $\mathbf{D}^{(i)}$, $i = 1, 2, 3, 4$ of the fourth-order tensor $\boldsymbol{\mathcal{D}}$ (Liu et al., 2009; Signoretto et al., 2010).

In Chapter 3 we proposed using the HOSVD as a tool for reconstructing and denoising prestack seismic volumes. The method is based on the truncated HOSVD of 4D $F-X$ slices, followed by an imputation/reinsertion expectation-maximization (EM) algorithm to recover the missing traces. The truncated HOSVD is the tool used for rank reduction while the modified EM algorithm is responsible for the completion. The HOSVD requires the specification of the rank of the tensor and a parameter that controls the reinsertion. On the other hand, the method proposed in this chapter does not require selecting the rank of the tensor and uses the optimization of a cost function to find the reconstructed data. Nevertheless, our new method requires certain parameters to be tuned before the inversion.

### 5.2.2 Alternating direction method of multipliers for tensor completion

For the minimization of the objective function in equation 5.2 we use the ADMM method (Arrow et al., 1968; Glowinski and Marrocco, 1975; Gabay and Mercier, 1976) as proposed for tensor completion by Gandy et al. (2011), with a minor modification (refer to Chapter 1 for an explanation of the method).

We need to rephrase the algorithm presented above for our particular problem (equation 5.2). First, the ADMM method requires the introduction of a new variable. Therefore, each unfolding of the data tensor will constitute a new tensor variable called $\boldsymbol{\mathcal{Y}}_i,\ i=1,2,3,4$ (one $\boldsymbol{\mathcal{Y}}_i$ per mode-$i$, same as the variable $\boldsymbol{\mathcal{W}}_i$), such that $\mathbf{D}^{(i)}=\mathbf{Y}_i^{(i)},\ i=1,2,3,4$. This means that the unfolding in mode-$i$ of $\boldsymbol{\mathcal{Y}}_i$ coincides with the unfolding in mode-$i$ of $\boldsymbol{\mathcal{D}}$. With these new variables $\boldsymbol{\mathcal{Y}}_i$, we add four additional constraints to the original objective function in equation 5.2, that is $\boldsymbol{\mathcal{Y}}_i=\boldsymbol{\mathcal{D}},\ i=1,2,3,4$. The algorithm needs one variable $\boldsymbol{\mathcal{Y}}_i$ per mode $i$ to iterate individually in each unfolding. The latter turns the problem of minimizing the original objective function (equation 5.2) into smaller minimization problems. The same explanation applies to the Lagrange multipliers $\boldsymbol{\mathcal{W}}_i,\ i=1,2,3,4$. The variables $\boldsymbol{\mathcal{Y}}_i, \boldsymbol{\mathcal{W}}_i$ could be labeled differently (for instance, $\boldsymbol{\mathcal{K}}, \boldsymbol{\mathcal{L}}, \boldsymbol{\mathcal{M}}, \boldsymbol{\mathcal{N}}$). However, we use subscripts to denote the four different variables for $\boldsymbol{\mathcal{Y}}$ and $\boldsymbol{\mathcal{W}}$ to keep the notation and equations compact, as we will notice in the next paragraphs. We remind the reader that the superscript $^{(i)}$ denotes unfolding in mode-$i$.

Let us identify the two convex functions that ADMM will use to solve the original problem (refer to section 1.4.4 in Chapter 1). These are

$$f(\boldsymbol{\mathcal{D}}) = \frac{\lambda}{2}\,\|\mathcal{T}\boldsymbol{\mathcal{D}} - \boldsymbol{d}^{obs}\|_F^2 \tag{5.3}$$

$$g(\boldsymbol{\mathcal{Y}}_i) = \sum_{i=1}^{4} \|\mathbf{Y}_i^{(i)}\|_*. \tag{5.4}$$

The relation between the split variables $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{Y}}_i$ will be $\boldsymbol{\mathcal{D}} = \boldsymbol{\mathcal{Y}}_i,\ i = 1, 2, 3, 4$, or equivalently

$$\boldsymbol{\mathcal{Y}}_1 = \boldsymbol{\mathcal{D}}, \quad \boldsymbol{\mathcal{Y}}_2 = \boldsymbol{\mathcal{D}}, \quad \boldsymbol{\mathcal{Y}}_3 = \boldsymbol{\mathcal{D}}, \quad \boldsymbol{\mathcal{Y}}_4 = \boldsymbol{\mathcal{D}}. \tag{5.5}$$

We have written $g(\boldsymbol{\mathcal{Y}}_i)$ implying $g(\boldsymbol{\mathcal{Y}}_1, \boldsymbol{\mathcal{Y}}_2, \boldsymbol{\mathcal{Y}}_3, \boldsymbol{\mathcal{Y}}_4)$ with the intention of shortening the notation and the same applies for the following formulas. In our problem, where the arguments of the objective function are tensors, the third term of equation 1.20 is rewritten as $< \boldsymbol{\mathcal{W}}_i, \boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{Y}}_i >$. The symbol $<>$ denotes the inner product in $\mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$, the sum of the (element-to-element) product of the entries of the tensor arguments (refer to Chapter 2). The matrix $\mathbf{G}$ in equation 1.20 is the identity operator in $\mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$ in our case (the unit superdiagonal tensor).

The augmented objective function to minimize is

$$J(\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{Y}}_i, \boldsymbol{\mathcal{W}}_i) \ = \ \frac{\lambda}{2}\|\mathcal{T}\boldsymbol{\mathcal{D}} - \boldsymbol{d}^{obs}\|_F^2 \ + \ \sum_{i=1}^{4}\left(\|\mathbf{Y}_i^{(i)}\|_* - \ < \boldsymbol{\mathcal{W}}_i, \boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{Y}}_i > \ + \frac{\beta}{2}\|\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{Y}}_i\|_F^2\right). \tag{5.6}$$

The ADMM method works by minimizing the objective function (equation 5.6) first with respect to $\boldsymbol{\mathcal{D}}$ assuming all other variables constant. Second, it minimizes $J$ with respect to $\boldsymbol{\mathcal{Y}}_i$ for all $i$ while keeping all other variables fixed. Finally, the multipliers $\boldsymbol{\mathcal{W}}_i$ are updated and iterations continue until the algorithm converges.

### 5.2.3 Minimization of $J$ with respect to $\boldsymbol{\mathcal{Y}}_i$

The minimum of $J$ with respect to $\boldsymbol{\mathcal{Y}}_i$ can be computed by using the following theorem. Given a matrix $\mathbf{A}$ and a scalar $\tau \geq 0$, let $\mathbf{B}$ be an approximation to $\mathbf{A}$ such that $\mathbf{B}$ has minimum nuclear norm. In other words,

$$\mathbf{B} = \arg\min_B \left\{ \|\mathbf{B}\|_* + \frac{1}{2\tau}\|\mathbf{B} - \mathbf{A}\|_F^2 \right\}. \tag{5.7}$$

The solution to this problem (Cai et al., 2010) is given by

$$\mathbf{B} = \text{shrink}(\mathbf{A}, \tau) = \mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}^{\text{H}}, \tag{5.8}$$

where

$$\tilde{\mathbf{\Sigma}} = \text{diag}[\max(\sigma_1 - \alpha,\, 0) \ldots \max(\sigma_r - \alpha,\, 0)] \tag{5.9}$$

and $\mathbf{A}$ is given by $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\text{H}}$. In other words, the solution to equation 5.7 is the SVD representation of $\mathbf{A}$ with singular values given by equation 5.9.

Now consider $\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{W}}_i$ fixed. For simplicity, we minimize the cost function with respect to $\boldsymbol{\mathcal{Y}}_i$. The minimum of the objective function given by equation 5.6 with respect to $\boldsymbol{\mathcal{Y}}_i$ is denoted $\widetilde{\boldsymbol{\mathcal{Y}}_i}$ and it is given by

$$\widetilde{\boldsymbol{\mathcal{Y}}_i} = \underset{\mathbf{Y}_i^{(i)}}{\arg\min} \left\{ \|\mathbf{Y}_i^{(i)}\|_* - < \mathbf{W}_i^{(i)}, \mathbf{D}^{(i)} - \mathbf{Y}_i^{(i)} > + \frac{\beta}{2} \|\mathbf{D}^{(i)} - \mathbf{Y}_i^{(i)}\|_F^2 \right\}. \tag{5.10}$$

By completing the square, the last expression can be rewritten as follows

$$\begin{aligned} \widetilde{\boldsymbol{\mathcal{Y}}_i} &= \underset{\mathbf{Y}_i^{(i)}}{\arg\min} \left\{ \|\mathbf{Y}_i^{(i)}\|_* - \frac{1}{2} \|\mathbf{W}_i^{(i)}\|_F^2 + \frac{1}{2} \|\mathbf{W}_i^{(i)}\|_F^2 - < \mathbf{W}_i^{(i)}, \mathbf{D}^{(i)} - \mathbf{Y}_i^{(i)} > \right. \\ &\qquad \left. + \frac{\beta}{2} \|\mathbf{D}^{(i)} - \mathbf{Y}_i^{(i)}\|_F^2 \right\} \\ &= \underset{\mathbf{Y}_i^{(i)}}{\arg\min} \left\{ \frac{1}{\beta} \|\mathbf{Y}_i^{(i)}\|_* + \frac{1}{2} \left\| \mathbf{Y}_i^{(i)} - \left( \mathbf{D}^{(i)} - \frac{1}{\beta} \mathbf{W}_i^{(i)} \right) \right\|_F^2 - \frac{1}{\beta} \|\mathbf{W}_i^{(i)}\|_F^2 \right\}. \end{aligned} \tag{5.11}$$

The last term in equation 5.11 is a constant and is unaffected by the derivative with respect to $\mathbf{Y}_i^{(i)}$. Using the property in expression 5.8, the minimum for $\boldsymbol{\mathcal{Y}}_i$ is

$$\widetilde{\boldsymbol{\mathcal{Y}}_i} = \text{fold} \left( \text{shrink} \left( \mathbf{D}^{(i)} - \frac{1}{\beta} \mathbf{W}_i^{(i)}, \frac{1}{\beta} \right) \right), \tag{5.12}$$

where "fold" means transforming a matrix into a tensor. This expression applies to all four variables $\boldsymbol{\mathcal{Y}}_i, i = 1, 2, 3, 4$.

### 5.2.4 Minimization of $J$ with respect to $\boldsymbol{\mathcal{D}}$

To find $\boldsymbol{\mathcal{D}}$, we consider the other two variables $\boldsymbol{\mathcal{Y}}_i, \boldsymbol{\mathcal{W}}_i$ fixed. The gradient of equation 5.6 is

$$\frac{\partial J(\boldsymbol{\mathcal{D}})}{\partial \boldsymbol{\mathcal{D}}} = \lambda \mathcal{T}' (\mathcal{T}\boldsymbol{\mathcal{D}} - \boldsymbol{d}^{obs}) - \sum_{i=1}^{4} \boldsymbol{\mathcal{W}}_i + 4\beta \boldsymbol{\mathcal{D}} - \sum_{i=1}^{4} \beta \boldsymbol{\mathcal{Y}}_i \tag{5.13}$$

where $\mathcal{T}'$ denotes an operator that maps from $\mathbb{C}^m \rightarrow \mathbb{C}^{I_1 \times I_2 \times I_3 \times I_4}$. This operator obeys $\mathcal{T}' \boldsymbol{d}^{obs} = \boldsymbol{\mathcal{D}}^{obs}$ and satisfies the following property

$$\mathcal{T}'\mathcal{T} (\boldsymbol{\mathcal{D}})\big|_{ijkl} = \begin{cases} D_{ijkl} & \text{if } ijkl \text{ contains an observation} \\ 0 & \text{if } ijkl \text{ is empty} \end{cases}, \tag{5.14}$$

equivalent to the operator presented by Liu and Sacchi (2004). Setting equation 5.13 equal to zero, we obtain

$$\widetilde{\boldsymbol{\mathcal{D}}} = (\lambda \boldsymbol{\mathcal{T}}' \boldsymbol{\mathcal{T}} + 4\beta \boldsymbol{\mathcal{I}})^{-1} \left[ \sum_{i=1}^{4} \boldsymbol{\mathcal{W}}_i + \sum_{i=1}^{4} \beta \boldsymbol{\mathcal{Y}}_i + \lambda \boldsymbol{\mathcal{T}}' \boldsymbol{d}^{obs} \right]. \tag{5.15}$$

If $\boldsymbol{\mathcal{A}}$ represents the argument in brackets in equation 5.15, the inverse of $(\lambda \boldsymbol{\mathcal{T}}' \boldsymbol{\mathcal{T}} + 4\beta \boldsymbol{\mathcal{I}})$ depends on which element of tensor $\boldsymbol{\mathcal{A}}$ it is applied to

$$[(\lambda \boldsymbol{\mathcal{T}}' \boldsymbol{\mathcal{T}} + 4\beta \boldsymbol{\mathcal{I}})^{-1} (\boldsymbol{\mathcal{A}})]\big|_{ijkl} = \begin{cases} (\lambda + 4\beta)^{-1} A_{ijkl} & \text{if } ijkl \text{ contains an observation} \\ (4\beta)^{-1} A_{ijkl} & \text{if } ijkl \text{ is empty} \end{cases} \tag{5.16}$$

Therefore, the minimum of $\boldsymbol{\mathcal{D}}$ is

$$\widetilde{\boldsymbol{\mathcal{D}}}\big|_{ijkl} = \begin{cases} (\lambda + 4\beta)^{-1} \left[ \sum_{i=1}^{4} \boldsymbol{\mathcal{W}}_i + \sum_{i=1}^{4} \beta \boldsymbol{\mathcal{Y}}_i + \lambda \boldsymbol{\mathcal{D}}^{obs} \right]_{ijkl} & \text{if } ijkl \text{ contains an observation} \\ (4\beta)^{-1} \left[ \sum_{i=1}^{4} \boldsymbol{\mathcal{W}}_i + \sum_{i=1}^{4} \beta \boldsymbol{\mathcal{Y}}_i \right]_{ijkl} & \text{if } ijkl \text{ is empty} \end{cases}, \tag{5.17}$$

where we replaced $\boldsymbol{\mathcal{T}}' \boldsymbol{d}^{obs} = \boldsymbol{\mathcal{D}}^{obs}$ and we accounted for $\boldsymbol{\mathcal{D}}^{obs}\big|_{ijkl} = 0$ when the bin $ijkl$ is empty. The ADMM algorithm needs two parameters $\lambda$ and $\beta$ to be specified. In contrast to the algorithm presented by Gandy et al. (2011), in our implementation $\lambda$ and $\beta$ remain constant throughout the iterations. However, we acknowledge that the trade-off parameter $\lambda$ should be dependent on the frequency to better suit the reconstruction problem at each particular frequency $\omega$. The main difference between our implementation and that from Gandy et al. (2011) is that we apply this algorithm for each frequency. Consequently, the cost of the algorithm if we have $N$ frequencies will be $N$ times the cost of applying it for one frequency. The cost of ADMM is dominated by the cost of the shrinkage operation. The computational cost could be reduced if we adopted the Lanczos bidiagonalization algorithm to replace the SVD calculation (Golub and Van Loan, 1996; Gao et al., 2013).

### 5.2.5 Final Tensor Completion Algorithm

Finally, we present the algorithm that solves equation 5.2 in Table 5.1. The proof of the convergence of this algorithm is presented in Gandy et al. (2011). We want to stress that, for a particular frequency, this algorithm converges to a rank that is constant for all unfoldings

(same $n$-ranks for all $n$). However, Tomioka et al. (2011) proposed an alternative formulation where not all the tensor unfoldings are assumed to be jointly low-rank. Their algorithm is able to automatically detect the lowest rank possible in each mode without constraining all of them to have the same rank. An approach like this could be used if we wanted to allow more freedom for the rank reduction operation to adjust to different ranks across different modes.

**Initialize** $\boldsymbol{\mathcal{D}}^0$, $\boldsymbol{\mathcal{Y}}_i^0, \boldsymbol{\mathcal{W}}_i^0,\ i = 1,2,3,4$ with the null tensor

**Repeat** until maximum number of iterations $k_{max}$ is reached

$\boldsymbol{\mathcal{D}}^{k+1}$ as in equation 5.17

**for** $i = 1,2,3,4$

$$\boldsymbol{\mathcal{Y}}_i^{k+1} = \text{fold}\left(\text{shrink}\left(\mathbf{D}^{(i)\,k+1} - \frac{1}{\beta}\mathbf{W}_i^{(i)\,k}, \frac{1}{\beta}\right)\right)$$

$$\boldsymbol{\mathcal{W}}_i^{k+1} = \boldsymbol{\mathcal{W}}_i^k - \beta\left(\boldsymbol{\mathcal{D}}^{k+1} - \boldsymbol{\mathcal{Y}}_i^{k+1}\right)$$

**end**

**end**

**Output** is $\boldsymbol{\mathcal{D}}^{k_{max}}$

Table 5.1: ADMM algorithm applied to the tensor completion problem, for one frequency slice.

## 5.3 Examples

### 5.3.1 Synthetic examples

In order to explore the performance of the presented algorithm, we consider a simple 3D model that contains two dipping planes with normals

$$n_1 = (0.02, 0.05, 0.99) \quad \text{and} \quad n_2 = (0.03, 0.02, 0.99). \tag{5.18}$$

The velocities used are 1500 m/s and 2300 m/s for the top and bottom planes, respectively, and the intersections of these planes with the vertical axis occur at 250 m and 1000 m (Figure 5.1). The travel-times are calculated in the midpoint-offset domain. The volume corresponds to a spatial tensor of size $12 \times 16 \times 12 \times 16$ (midpoints $x, y$-offset $x, y$), 512 time samples and a time sampling rate of 2 ms. Additionally, we randomly remove 50% of the traces and add random noise to produce a volume with $SNR = 1$, where $SNR = \frac{\sigma^2_{\text{data}}}{\sigma^2_{\text{noise}}}$ ($\sigma^2_{\text{data}}$ is the

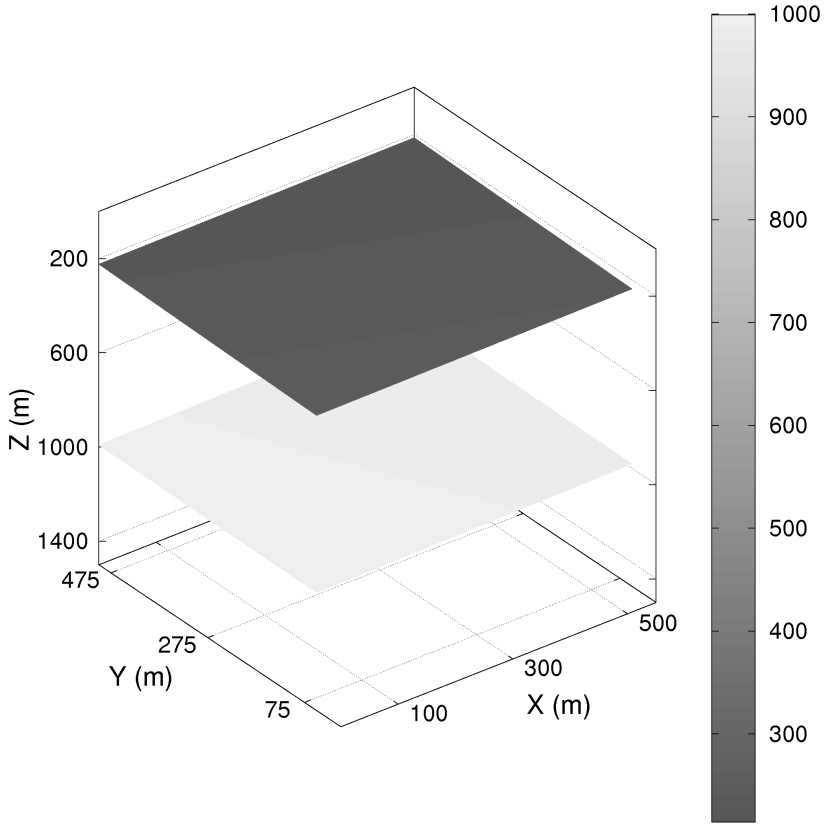



Figure 5.1: Model of two dipping planes used in testing the parameters of the algorithm. The colorbar on the right indicates depth. The normal to the planes are $n_1 = (0.02, 0.05, 0.99)$ and $n_2 = (0.03, 0.02, 0.99)$ for the first and second reflector, respectively. We compute synthetic data from this model using ray tracing.

variance of the data and $\sigma^2_{\text{noise}}$ is the variance of the noise). Figure 5.2 shows a portion of the simulated data.

A common concern when using these type of algorithms is how to correctly choose the parameters $\lambda$ and $\beta$. With the aid of synthetics, we investigate the behaviour of the algorithm for different values of these parameters. We compute the quality of the reconstruction for different values of $\lambda, \beta$ while keeping one of them fixed. As in previous chapters, the quality of the reconstruction in dB units is $Q = 10 \log \frac{\|\boldsymbol{\mathcal{D}}^{\text{true}}\|^2}{\|\boldsymbol{\mathcal{D}}^{\text{out}} - \boldsymbol{\mathcal{D}}^{\text{true}}\|^2}$, where $\boldsymbol{\mathcal{D}}^{\text{true}}$ is the complete and noise-free volume and $\boldsymbol{\mathcal{D}}^{\text{out}}$ is the reconstructed and noise-attenuated volume (in time). If $Q = 2$ dB, the Frobenius norm of $\boldsymbol{\mathcal{D}}^{\text{true}}$ is 1.6 times bigger than the Frobenius norm of the misfit; while if $Q = 10$ dB, the Frobenius norm of $\boldsymbol{\mathcal{D}}^{\text{true}}$ is 10 times bigger than the Frobenius norm of the misfit. Overall, we consider any result with a $Q$ larger than 10 dB an

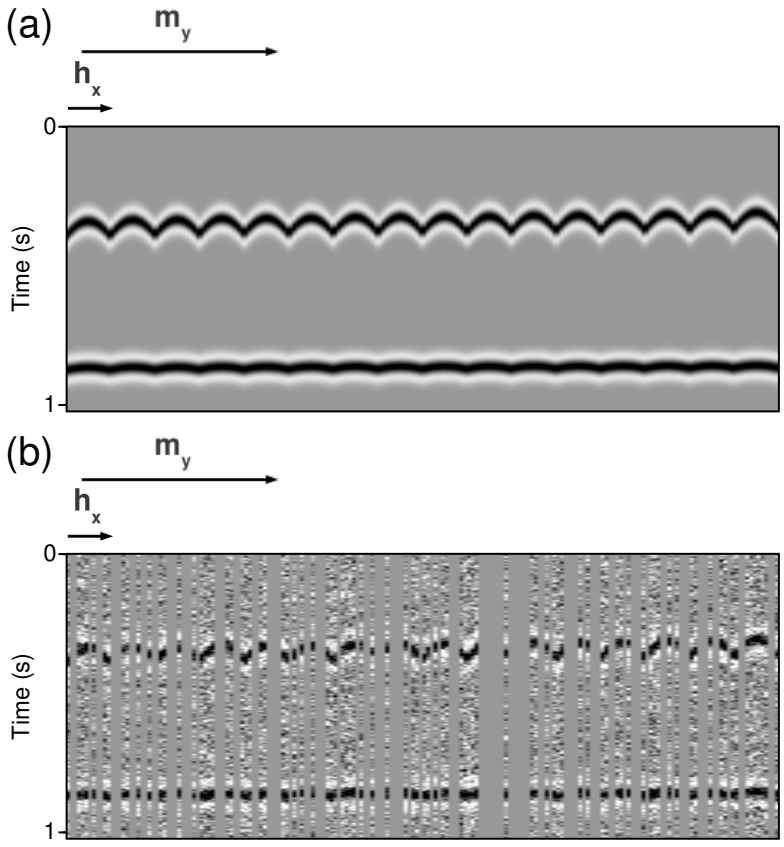



Figure 5.2: Portion of the data generated using the model from Figure 5.1, for fixed $m_x$ and $h_y$. The fast varying coordinate is $h_x$, while the slow varying coordinate is $m_y$. a) is the fully sampled volume with $SNR = 100$ and b) is the decimated volume with $SNR = 1$.

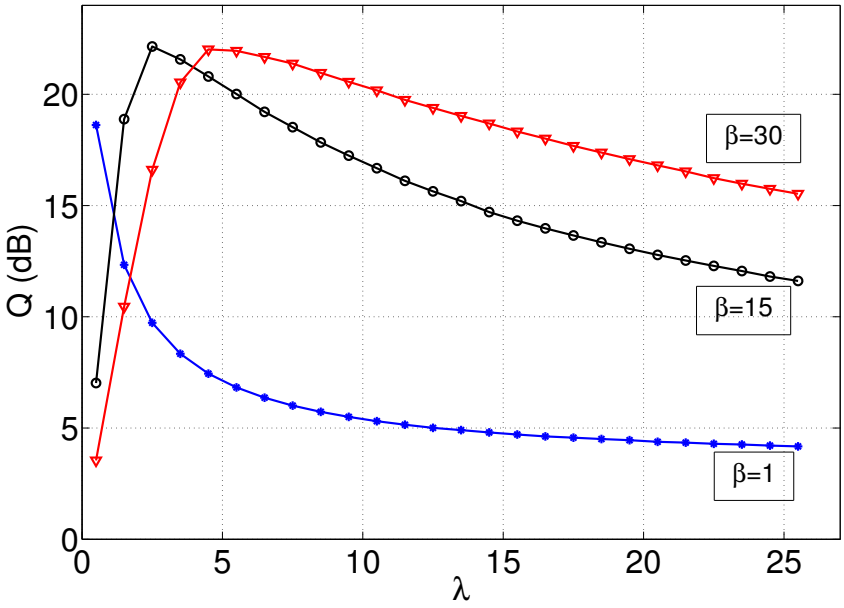



Figure 5.3: Quality of the reconstruction $Q$ versus $\lambda$ for $\beta = 1, 15$ and 30. A higher value of $Q$ indicates a higher reconstruction quality. For a fixed value of $\beta$, quality tends to decrease as $\lambda$ increases.

acceptable reconstruction. We use 200 iterations for each frequency to allow the algorithm to converge. We must stress that these tests have not been developed for one frequency but for all the frequencies. In other words, we are analyzing the global behaviour of the parameters $\lambda$ and $\beta$.

Figure 5.3 shows the variation of the quality of the reconstruction $Q$ with the trade-off parameter $\lambda$ for different values of $\beta$. We observe that $Q$ deteriorates for the lowest value of $\beta$ and a few combinations of $\lambda, \beta$ give comparable values of $Q$. From these tests, we conclude that larger values of $\beta$ and smaller values of $\lambda$ give the best reconstruction. The value $\lambda = 2.5$ gives the best $Q$ when $\beta = 15$. These tests demonstrated that values of $\beta$ larger than 30 tended to deteriorate the reconstruction. We will use the values of $\beta = 15$ and $\lambda = 2.5$ throughout the rest of our calculations. An alternative to our choice of $\beta$ was given by Boyd et al. (2011, p. 20).

The parameter $\beta$ controls the fit of $\boldsymbol{\mathcal{D}}$ to $\boldsymbol{\mathcal{Y}}_i, i = 1, 2, 3, 4$ and contributes to the robustness of the algorithm. In theory, the quality of the reconstruction $Q$ is weakly affected by the parameter $\beta$ since it does not influence the fitting of the results to the observations (actually, it controls the fit between the dummy variables and the data). Our simulations support this except for small values of $\beta$, which might be caused by an insufficient number of iterations

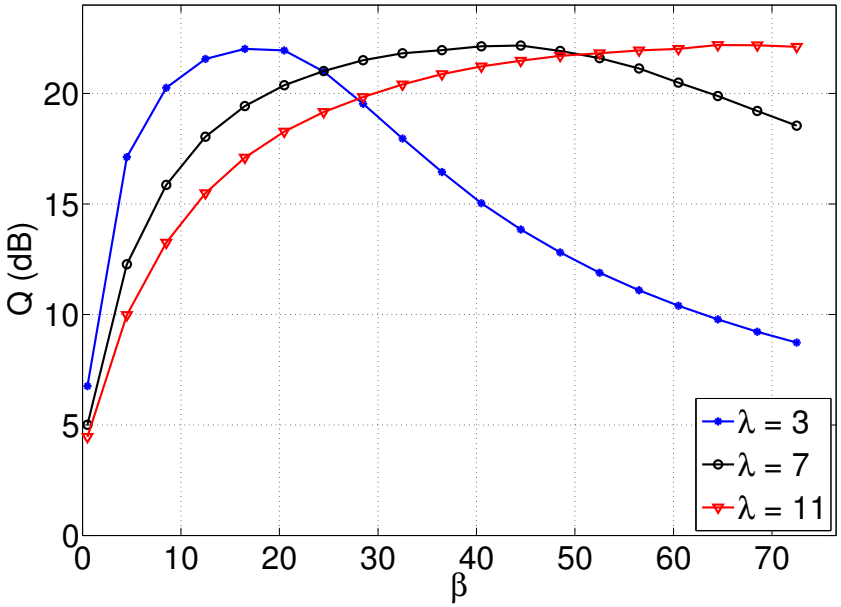



Figure 5.4: Quality of the reconstruction $Q$ versus $\beta$, for $\lambda = 3, 7$ and 11.

or the presence of noise in the data. As observed in Figure 5.4, different combinations of $\lambda, \beta$ give a good reconstruction quality. As a general rule, smaller $\beta$ requires smaller $\lambda$ and larger $\beta$ requires larger $\lambda$, similarly to what we observe in Figure 5.3. As expected, we also find a large value of $Q$ when $\lambda \simeq 2.5$ and $\beta \simeq 15$ in Figure 5.4.

The presented algorithm provides an automatic rank determination. The user does not need to specify the rank value to perform rank reduction, such as in the algorithm in Chapter 3 (Kreimer and Sacchi, 2012d). We need to confirm that the singular value distribution for a fixed frequency before the decimation and after the reconstruction remain similar. Figure 5.5 displays the distribution of singular values for one frequency in the reconstruction. Each of the curves represent the average of the singular values for the four unfoldings of the tensor. For each of these unfoldings, the singular values are similar in magnitude, therefore we do not lose information by averaging them. The black asterisk line represents the singular value distribution for the complete and noise-free volume, while the red open circle line corresponds to the decimated and noisy volume. The blue triangle line belongs to the reconstructed and denoised volume. From this figure, we see that the reconstructed volume has a similar singular value distribution to the original volume. Furthermore, we notice that the singular values for the noisy, decimated volume never approximate zero.

Using the optimal values of $\lambda$ and $\beta$ found with our simulations, we reconstruct a larger 5D synthetic volume of size $512 \times 20 \times 19 \times 20 \times 19$. The model (Figure 5.6) is similar to the

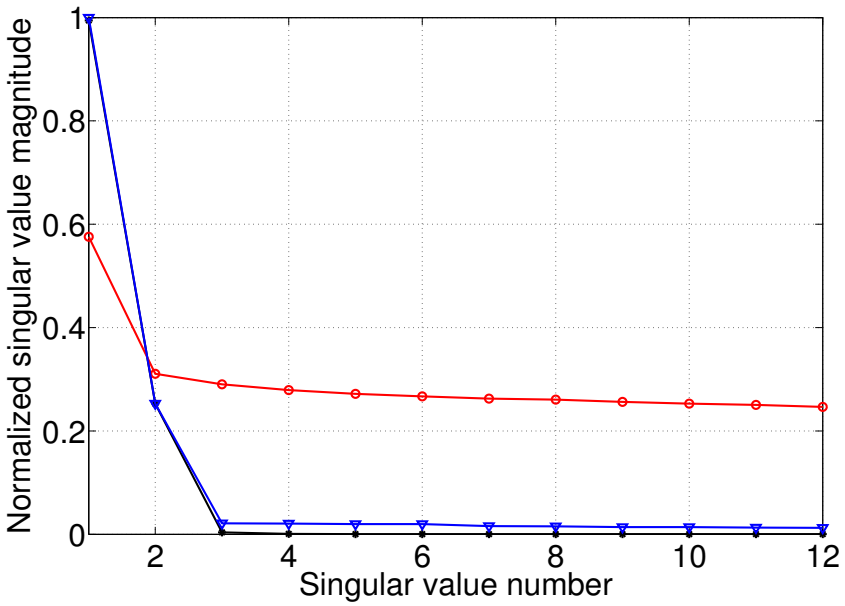



Figure 5.5: Normalized singular values distribution for one frequency. Each curve is an average of the singular values for the four unfoldings of the 4D tensor. The line annotated with asterisks identifies the complete and noise-free volume, the line marked with open circles corresponds to the decimated and noisy volume. The line annotated with triangles indicates the reconstructed and denoised volume. The proposed reconstruction method results in a singular values distribution that is much closer to the intended target than the input decimated data.

one considered previously and contains two dipping planes with normals

$$n_1 = (0.26, 0.17, 0.95) \quad \text{and} \quad n_2 = (0.12, 0.16, 0.98). \tag{5.19}$$

The intersection of the planes with the vertical axis occurs at 350 m and 1000 m and the velocities coincide with those in the previous model. We remove 40% of the traces randomly and add randomly distributed Gaussian noise with a $SNR = 1$. Figure 5.7 shows CMP gathers, for a subset of the data, with two spatial coordinates fixed. Furthermore, Figure 5.8 displays offset gathers, with two spatial coordinates fixed. The quality of the reconstruction for this case is $Q = 20$ dB and the running time is 4 h 28 min in MATLAB (on the same computer specified in the field data section). It is important to notice that the reflections have significant curvature in some dimensions. NMO correction is not applied prior to reconstruction. This is an important difference compared to Fourier based reconstruction methods that use sparsity in the wave-number domain, where NMO is needed to minimize the number of distinct dips in the data (Trad, 2009).

We perform some simple tests to visualize the difference between the HOSVD-based reconstruction algorithm and the presented ADMM-based technique (more thorough testing presented in Chapter 6). We apply both methods on the first synthetic example presented in this section. We set the number of iterations to 200 per frequency for the ADMM-based tensor completion, 20 iterations per frequency for the HOSVD-based tensor completion and randomly remove from 10% to 90% of the samples. Each decimation percentage is achieved by using 10 different seeds for the random number generator. Therefore, the quality of the reconstruction $Q$ has a standard deviation associated with its mean value. Figure 5.9 contains the results from this simulation. The sampling ratio is defined as the ratio of input traces to the total amount of traces in the grid we want to reconstruct. We see that the ADMM-based method gives slightly better reconstruction results for the whole decimation range, although at a larger computational cost. For this example, the average running time for the ADMM method is 45 minutes while the average running time for HOSVD is 2 minutes. Stanton et al. (2012) presented a comparison between commonly used reconstruction methods and HOSVD. Although we did not reproduce the tests in their article for the ADMM-based method, we expect a similar behaviour with our new method versus POCS (Abma and Kabir, 2006) and MWNI (Liu and Sacchi, 2004). Stanton et al. (2012) found that the tensor completion method was robust to both low and high $SNR$ and surpasses the other methods especially in the presence of high levels of decimation and curvature. In their real data example, they observed that tensor completion gave a small amount of amplitude smearing in the near offsets and offered more denoising capabilities than MWNI (although denoising is not always a desired feature in reconstruction).

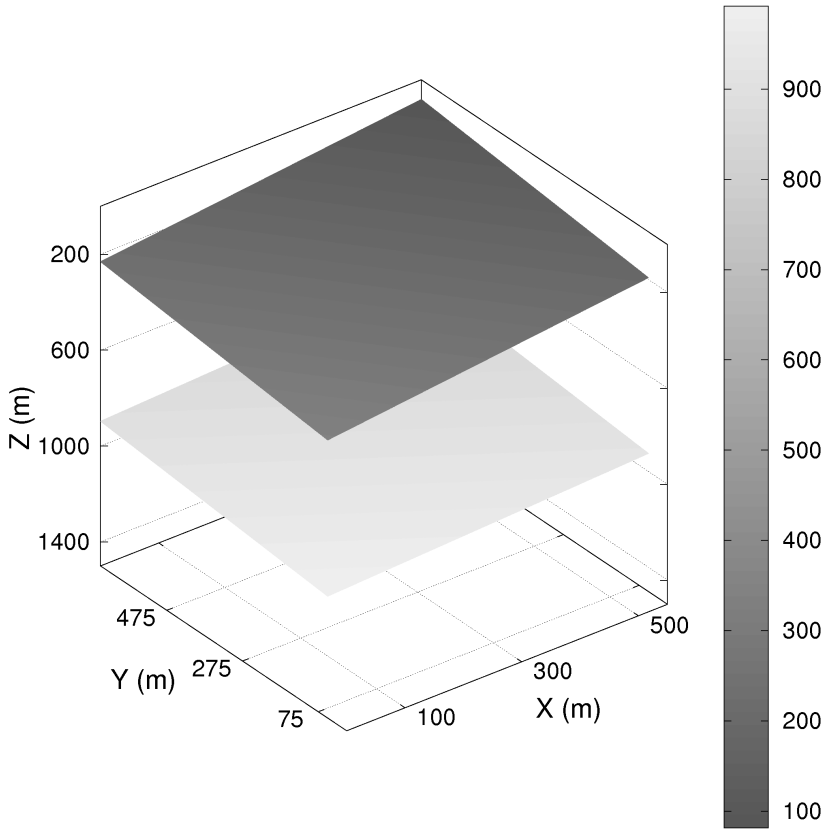



Figure 5.6: Model of two dipping planes corresponding to the data in Figures 5.7 and 5.8. The colorbar indicates depth.

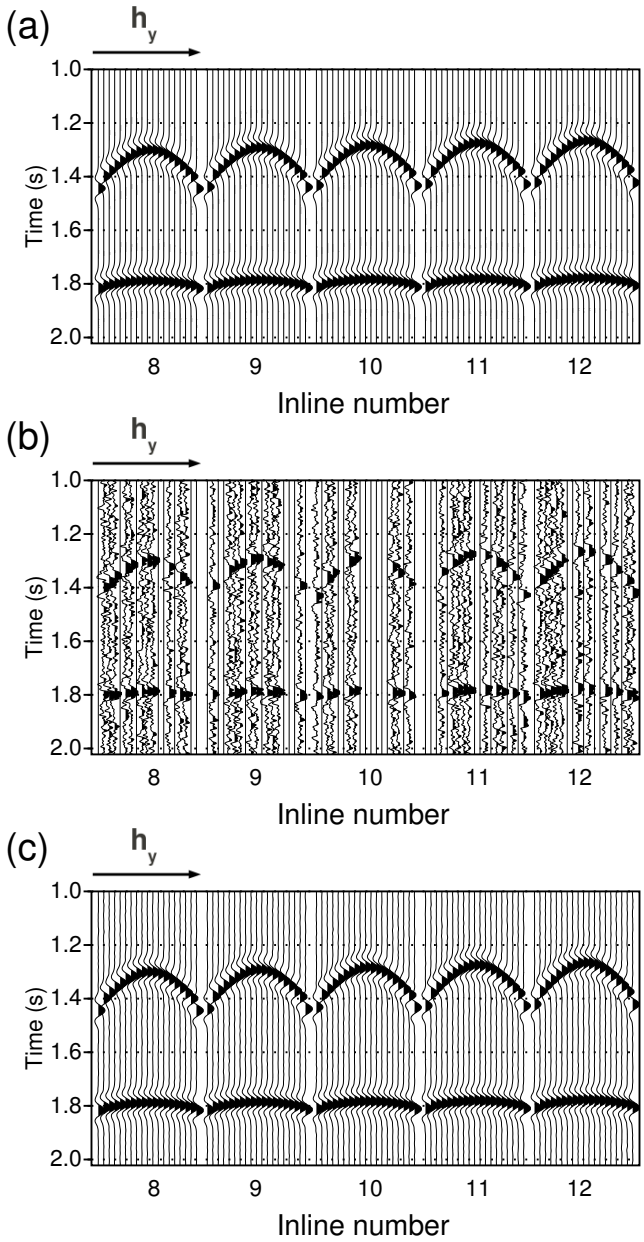



Figure 5.7: Common midpoint gathers with 40% randomly decimated traces and $SNR = 1$. (a) Portion of the 5D desired volume. (b) Decimated and noisy data. This is the input to the algorithm. (c) Reconstructed data.

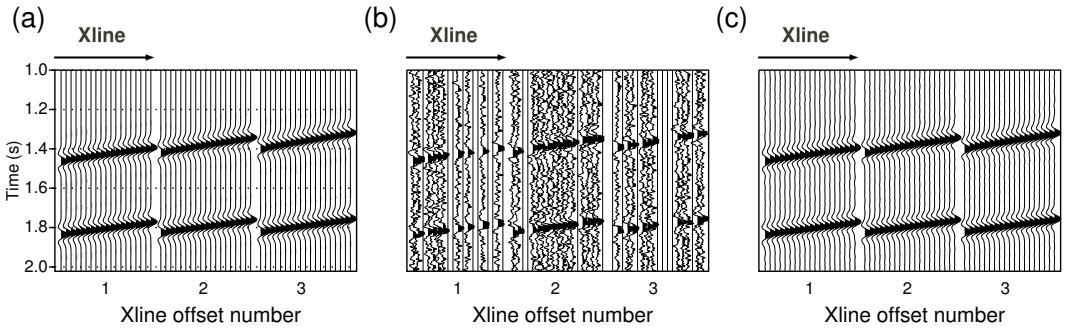



Figure 5.8: Common offset gathers with 40% randomly decimated traces and $SNR = 1$. (a) Portion of the 5D desired volume. (b) Decimated and noisy data. This is the input to the algorithm. (c) Reconstructed data.

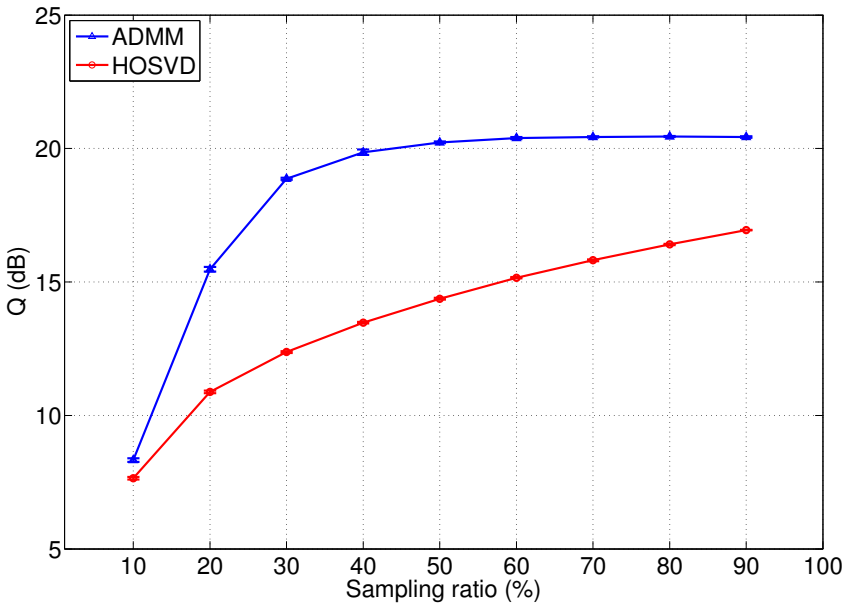



Figure 5.9: Quality of the reconstruction versus sampling ratio for a synthetic volume. The quality of the reconstruction increases as the sampling ratio increases for both the proposed method and HOSVD reconstruction.

### 5.3.2 Field data example

Our real data example is from an orthogonal survey acquired over a heavy oil field in Alberta, Canada. The target in this field is the McMurray formation. The Clearwater formation overlays the reservoir at $\sim 400$ ms while the Paleozoic lays under the McMurray at $\sim 800$ ms. The McMurray formation produces heavy oil and it is exploited by a thermally assisted technique. It is believed that this technique induces increased stress within the reservoir interval. In this field, Horizontal-Transversely Isotropic (HTI) effects are better characterized when having a complete azimuthal sampling.

It is a common practice in the reconstruction of seismic data to use overlapping windows in both space and time. The main reasons for using windows are that the assumptions of stationarity of the wavefield and sparsity are better satisfied in smaller windows. In addition, the reconstruction tends to be computationally faster in smaller pieces of data. We reconstruct a crossline swath from this survey by dividing it into 21 overlapping blocks of inline/crossline (13 inlines overlap length). For each block, the dimensions of the grid used are

Inline (CMPx): $n_1 = 26$, increment $= 5\,\text{m}$ (with an inline overlap of 13 CMP positions)
Crossline (CMPy): $n_2 = 26$, increment $= 5\,\text{m}$
Offset: $n_3 = 5$, increment $= 100\,\text{m}$, [ min, max ]=$[50, 450]\,\text{m}$
Azimuth: $n_4 = 8$, increment $= 45°$, [ min, max ] $= [22.5°, 337.5°]$

The azimuth is measured counterclockwise from East. Furthermore, we use overlapping time windows of 200 samples for the reconstruction of each block, where the total number of samples is 1000 (5 windows in total with an overlap of 10 samples plus a smaller sixth window). Figure 5.10 displays the shots and receivers that form the 21 blocks of data and highlights both the 21 blocks and the region spanned by the midpoints used in the reconstruction. Figure 5.11 contains the distribution of azimuth and offset for only one block. For clarity, we present the source-receiver geometry for one midpoint bin in the centre of the survey after binning in Figure 5.12, before and after reconstruction. The total number of grid points per block are $26 \times 26 \times 5 \times 8 = 27040$ whereas the total amount of traces in the area are such that 40-45% of the grids are populated. The data are NMO-corrected and low-pass filtered below 100 Hz prior to the reconstruction. The frequencies considered in the algorithm are from 0.1 to 100 Hz, with a sampling rate of 1 ms. From the analysis implemented with synthetic tests, we set $\lambda = 2.5$, $\beta = 15$ and the number of iterations to 200. The running time on a single processor Intel Xeon(R) running at 3.16 Ghz using MATLAB is approximately 1 h for each block.

Figure 5.13 shows a CMP gather for a fixed crossline and inline. The panels on top of (a)

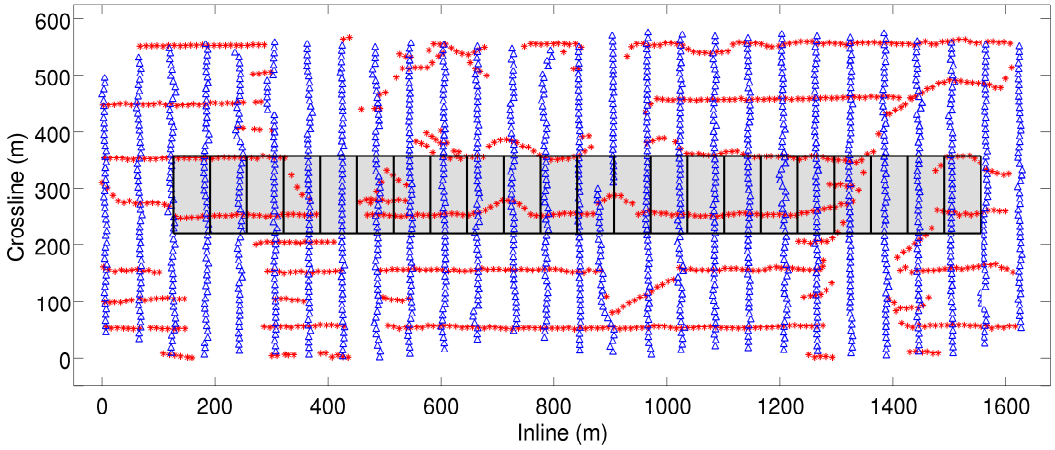



Figure 5.10: Survey geometry for the real data example, in true aspect ratio. Stars indicate sources and triangles indicate receivers. The grey area shows the midpoints that were reconstructed with the proposed algorithm. We use 21 blocks of size $26 \times 26$ midpoints with an overlap of 13 inline CMP positions. Each block is represented by two adjacent rectangles.

and (b) display the distribution of offset and azimuth for this gather in red and blue lines, respectively. Notice that because we are performing the reconstruction in 5D, the algorithm makes use of the information of these dimensions for each CMP. The four dimensions considered aid in the reconstruction of larger gaps than possible when fewer spatial coordinates are involved and increase the robustness of the method.

We portray common offset/azimuth gathers in Figures 5.14-5.16 for three different offset ranges, a constant crossline and for $az = 22.5°$ and $112.5°$. Figure 5.17 displays the power spectrum of the gather shown in Figure 5.14 before and after reconstruction, demonstrating that the reconstruction has not damaged the spectrum of the data. Furthermore, Figure 5.18 contains a stack section for a constant crossline with and without reconstruction. Many of the reflections, especially below 0.4 s, are denoised when using reconstruction and look continuous, in contrast to what was observed in the stack without reconstruction.

## 5.4 Conclusions

In this chapter, we formulate the problem of prestack seismic data reconstruction and denoising as an inverse problem, where we minimize the nuclear norm of the tensor subject to data constraints. The objective function is minimized utilizing the alternating direction

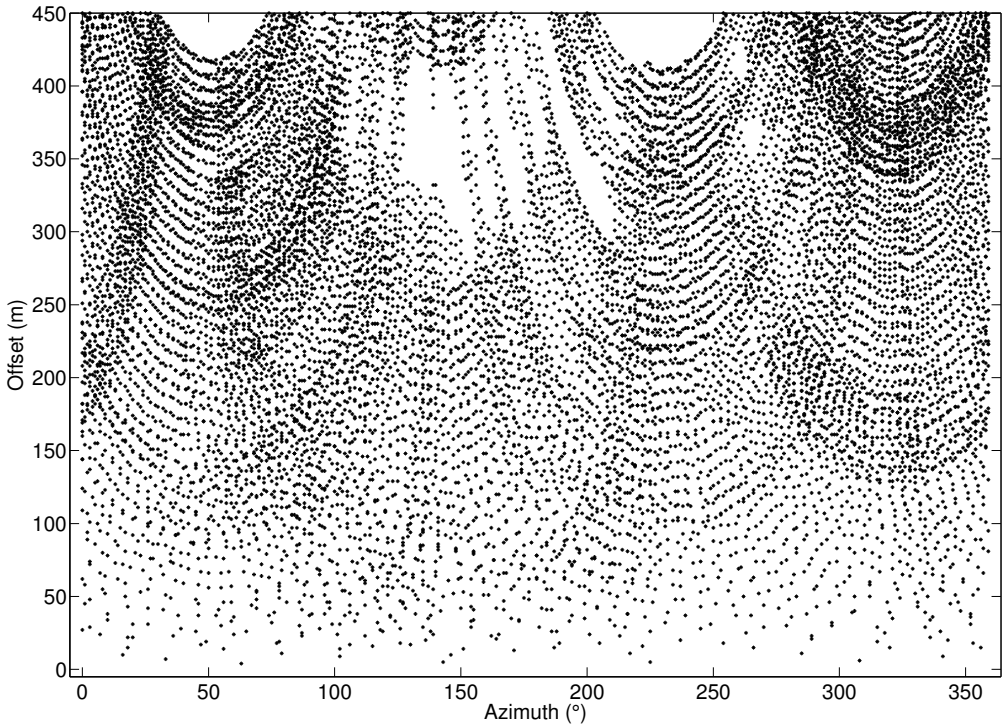



Figure 5.11: Offset azimuth distribution for one block of data used in the reconstruction.

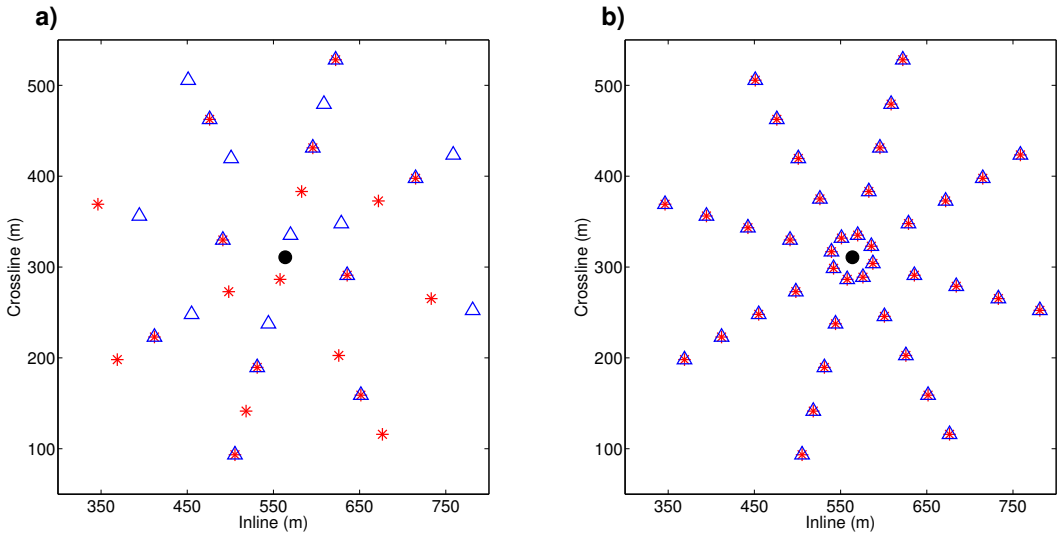



Figure 5.12: Source receiver geometry for the bin with its center at $(564, 311)$m. The black dot marks the position of the midpoint, triangles indicate receivers and stars indicate sources. (a) Position of the available source-receiver pairs before reconstruction. (b) Source-receiver pairs after reconstruction.

method of multipliers one frequency at a time. Contrary to other multidimensional reconstruction techniques, the proposed method is based on tensor completion and minimizes the nuclear norm of the 4D tensor in the $F - X$ domain. This strategy leads to an automatic rank determination of the reconstructed data, bypassing the need for the user to know the optimal rank. The trade-off parameters of the algorithm are tuned by running simple tests. Using the correct parameters for the inversion leads to a high-quality reconstruction even in the presence of dipping events and low $SNR$. Considering this method's assumptions to be similar to those of HOSVD-based reconstruction, we verify that both methods give practically the same results. Our land data example demonstrates the performance of the algorithm in a real case scenario.

Although the running times for the proposed algorithm are longer than for HOSVD, ADMM-based reconstruction presents an explicit formulation to the rank reduction based reconstruction problem. The set-up for this problem is along the same lines as current research in the compressive sensing field (more precisely, the matrix/tensor completion problem). So far the method presented here has proven to be computationally too expensive to be attractive for industrial applications. Further research should be focused on finding an automatic case-independent tuning of the parameters $\lambda$ and $\beta$ and improving the convergence rate of the algorithm.

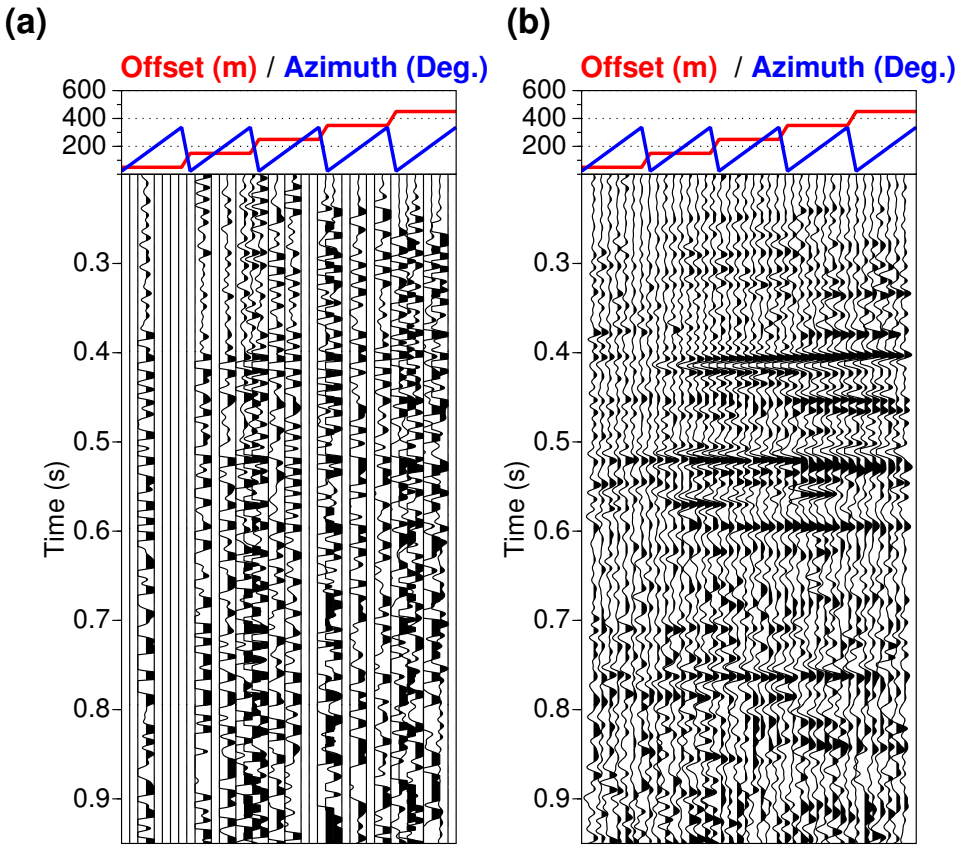



Figure 5.13: CMP gather for a fixed inline and crossline. (a) Before reconstruction. (b) After reconstruction.

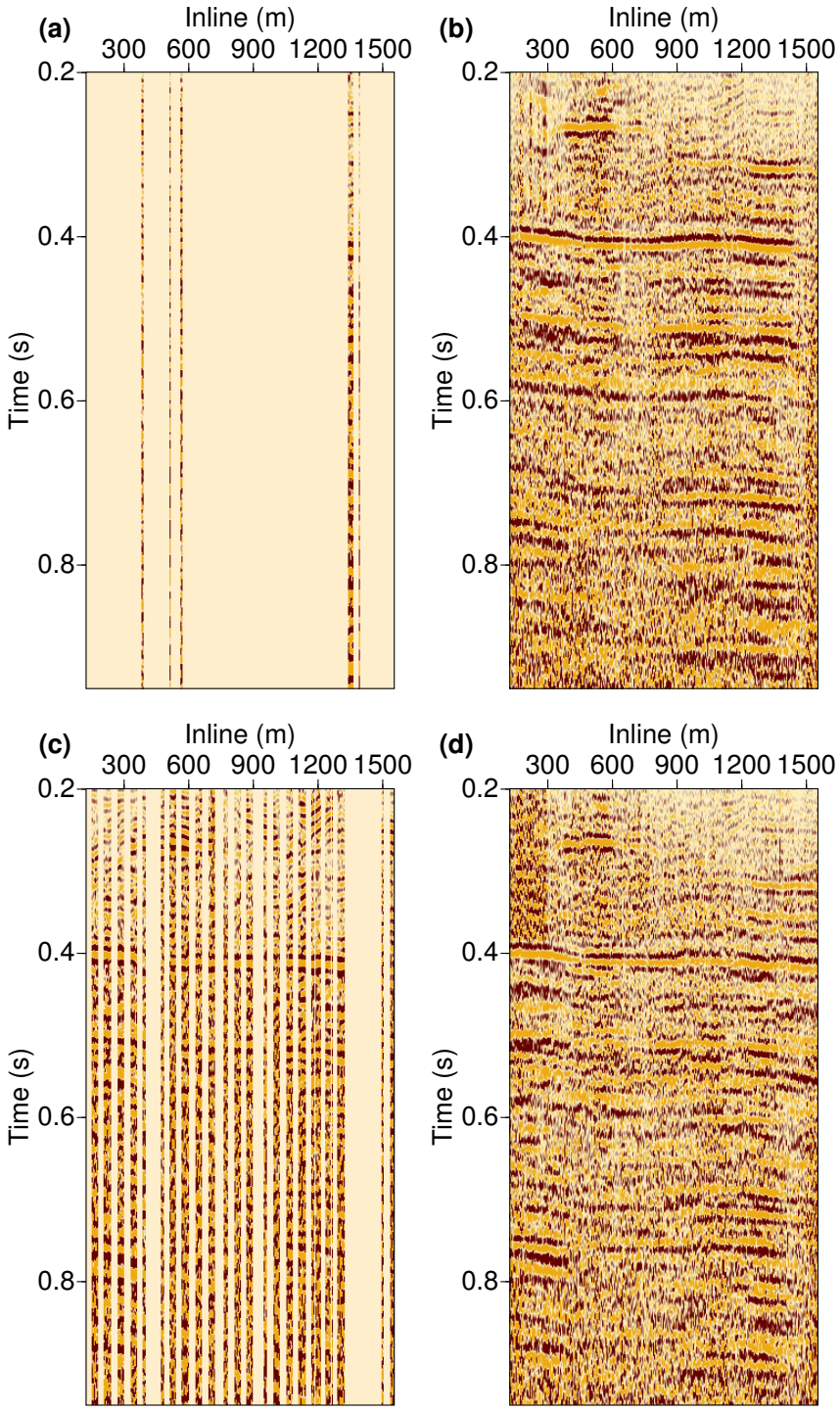



Figure 5.14: Near offset gathers (50 m) for a constant crossline. Panels (a) and (b) have a constant azimuth of 22.5° while (c) and (d) have a constant azimuth of 112.5°. (a)-(c) Before reconstruction. (b)-(d) After reconstruction.

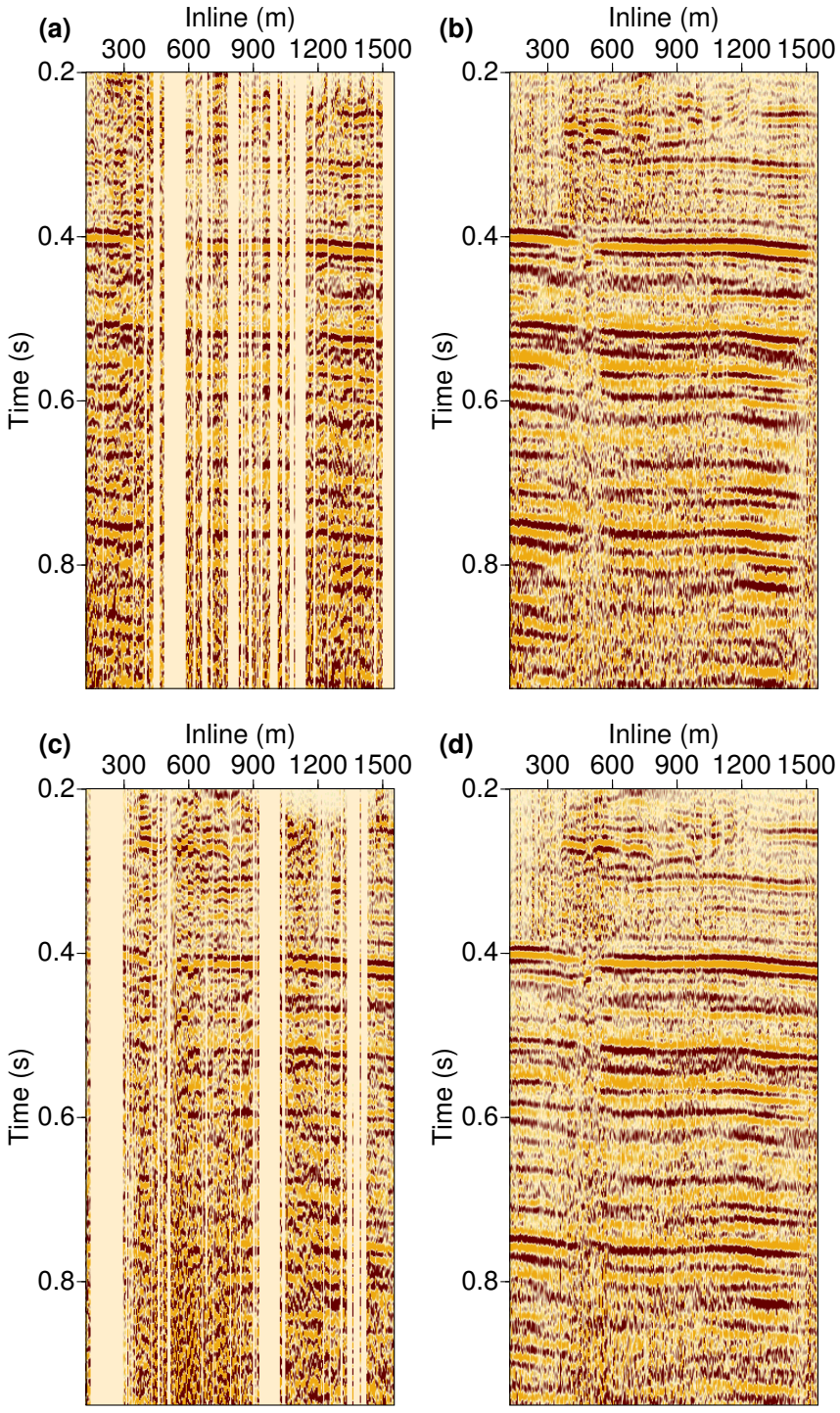



Figure 5.15: Mid-offset gathers (250 m) for a constant crossline. Panels (a) and (b) have a constant azimuth of 22.5° while (c) and (d) have a constant azimuth of 112.5°. (a)-(c) Before reconstruction. (b)-(d) After reconstruction.

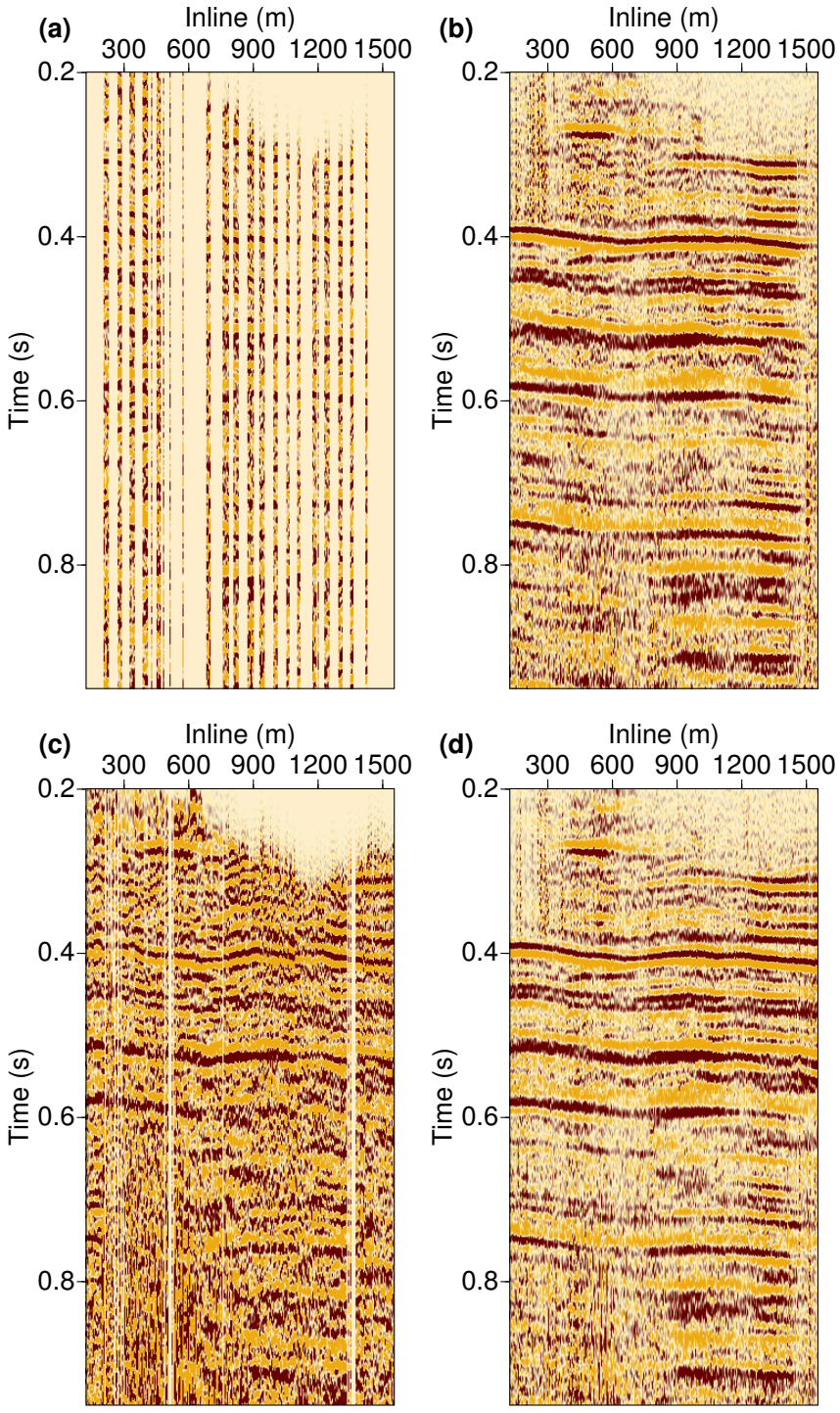



Figure 5.16: Far offset gathers (450 m) for a constant crossline. Panels (a) and (b) have a constant azimuth of 22.5° while (c) and (d) have a constant azimuth of 112.5°. (a)-(c) Before reconstruction. (b)-(d) After reconstruction.

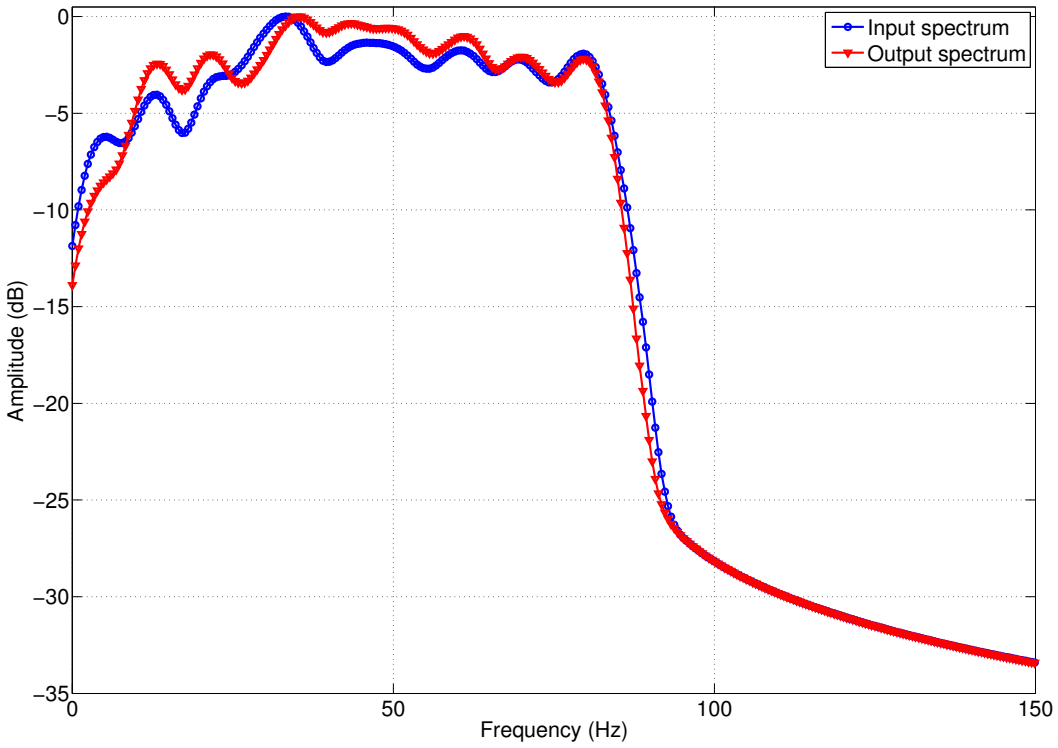



Figure 5.17: Power spectrum in dB before and after reconstruction for the gathers shown in Figure 5.15(a) and (b).

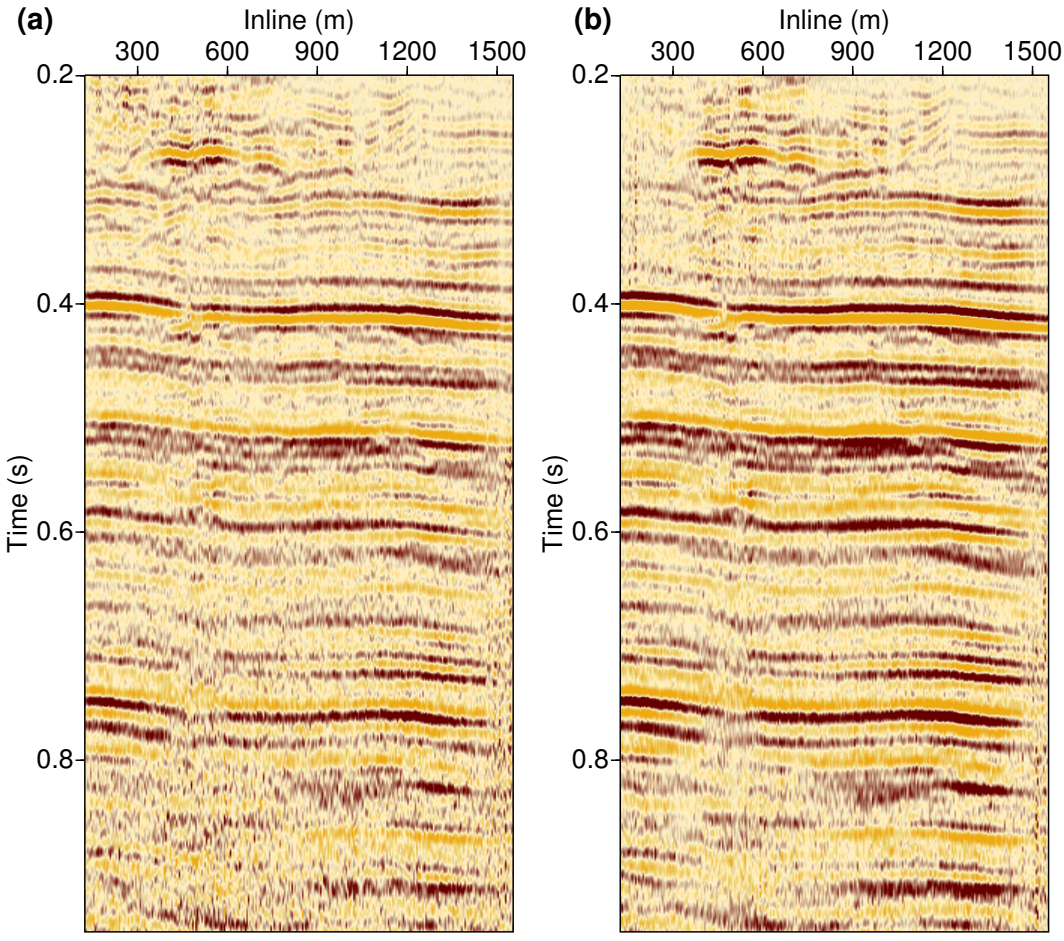



Figure 5.18: Stack of all 21 blocks for a constant crossline: (a) Without reconstruction. (b) With reconstruction.

# CHAPTER 6

# Comparison of methods

In the previous chapters we have covered three different ways of performing rank reduction on tensors for simultaneous reconstruction and noise attenuation. In our synthetic experiments and in the real data implementations all three methods gave satisfactory results. In this chapter we want to compare these three methods with Cadzow/MSSA (Trickett et al., 2010; Oropeza and Sacchi, 2011), which also belongs to the rank reduction class of reconstruction methods (operating on Hankel matrices). Furthermore, we will compare the methods presented in previous chapters with Minimum Weighted Norm Interpolation (MWNI) (Liu, 2004), a Fourier based reconstruction method which is a "classical" method commonly used in industrial applications. Our intention is to evaluate all methods with the same synthetic examples and real data example. Additionally, we will present a simulation on the binning parameters with the purpose of obtaining a suitable grid for reconstruction.

In this chapter we will adopt the following nomenclature for the methods we will compare:

**HOSVD**: the method presented in Chapter 3.

**SEQ-SVD**: the method presented in Chapter 4.

**ADMM**: the method presented in Chapter 5.

**CADZOW/MSSA**: First proposed by Trickett et al. (2010) and later expanded by Oropeza and Sacchi (2011) and Gao et al. (2013), this method is based on rank reduction of Hankel or Toeplitz matrices. For a 2D problem and one frequency $\omega$, we implement rank reduction on a Hankel/Toeplitz matrix formed with the $F - X$ slice, followed by the averaging along the antidiagonals, to obtain the rank-reduced data. An important property is that if the data is composed by a superposition of $r$ sinusoids, the rank of the Toeplitz matrix is $r$. This property is very useful at the time of implementing this method in synthetic examples (and

in real data, up to certain extent). For the multidimensional (5D) problem, each dimension forms a Hankel or Toeplitz matrix in levels, such that the final Topelitz matrix is a level-4 block Toeplitz matrix (Gao et al., 2013). This method utilizes Lanczos bidiagonalization for the low-rank approximation and circulant matrices for fast matrix-vector multiplications. Finally, the same modified imputation algorithm as in Chapter 3 and 4 performs the reconstruction. The codes from Gao et al. (2013) are used here for our simulations.

**MWNI**: Minimum weighted norm interpolation is a Fourier-based reconstruction method developed by Liu and Sacchi (2004). This method operates by minimizing a cost function that contains a misfit term and a model norm term, as follows

$$J = \|\mathbf{T}\mathbf{x} - \mathbf{y}\|_F^2 + \rho^2 \|\mathbf{x}\|_{\mathcal{W}}^2, \tag{6.1}$$

where $\rho$ is the trade-off parameter, $\mathbf{T}$ is the sampling operator, $\mathbf{x}$ is the data on a regular grid and $\mathbf{y}$ are the observations (also on a regular grid). The model norm is defined as

$$\|\mathbf{x}\|_{\mathcal{W}}^2 = \sum_{k \in \mathcal{K}} \frac{X_k^* X_k}{P_k^2}, \tag{6.2}$$

where $\mathcal{K}$ is the region of spectral support of the signal, the $^*$ sign denotes the conjugate and $P_k$ are spectral domain weights that resemble the ones from the signal to reconstruct. In practice, the weights are calculated with the smooth periodogram of the data.

## 6.1 Synthetic examples

Our first synthetic example contains three linear events in a 5D volume. The grid size is $12 \times 12 \times 12 \times 12$, the sampling rate is 2 ms and the volume contains 256 time samples. The wavelet's central frequency is 20 Hz and it is a Ricker wavelet. The sampling operator removes 70% of the traces randomly according to a uniform distribution. We present two different cases of this example, one with $SNR = 100$ and another with $SNR = 1$ (randomly distributed Gaussian noise). As a reminder, we measure the quality of the reconstruction in dB in the time domain as

$$Q = 10 \log \frac{\|\boldsymbol{\mathcal{D}}_{\text{true}}\|_F^2}{\|\boldsymbol{\mathcal{D}}_{\text{recovered}} - \boldsymbol{\mathcal{D}}_{\text{true}}\|_F^2}. \tag{6.3}$$

For all the examples in this chapter, we parametrized each of the methods using the following values:

**HOSVD**: A maximum of 20 iterations per frequency, core-tensor size $3 \times 3 \times 3 \times 3$, reinsertion parameter $a = 1$ when $SNR = 100$, and $a = 0.9$ when $SNR = 1$ (the real data example

used the latter value of $a$).

**SEQ-SVD**: A maximum of 20 iterations per frequency, rank $r = 3$ per unfolding, reinsertion parameter $a = 1$ when $SNR = 100$, and $a = 0.3$ when $SNR = 1$ (the real data example used the latter value of $a$).

**ADMM**: A maximum of 150 iterations per frequency, parameters $\beta = 15, \lambda = 2.5$.

**Cadzow/MSSA**: A maximum of 10 iterations per frequency, rank $r = 5$, reinsertion parameter $a = 1$ when $SNR = 100$, and $a = 0.8$ when $SNR = 1$ (the real data example used the latter value of $a$). These values were based on those from Gao et al. (2013).

**MWNI**: A maximum of 15 internal and 3 external iterations for the conjugate gradients solver. When $SNR = 1$, the trade-off parameter for MWNI is $\rho = 10$. We determined this value by the evaluation of $Q$ using different parameters. When $SNR = 100$, $\rho = 0$ and the number of iterations replaces this parameter, according to Liu and Sacchi (2004).

We do not show a figure of the results for the example in which $SNR = 100$ because all the values of $Q$ are higher than 15 dB, making it impossible to distinguish the results from one another. The running times on a single processor Intel Xeon(R) running at 3.07 Ghz using MATLAB and quality values for the reconstructions appear in Table 6.1. From the values in the first column of the table, we know that all methods perform the task efficiently, being HOSVD, SEQ-SVD, Cadzow/MSSA and MWNI the best ones. The result for ADMM has a $Q = 13.6$ dB. It is possible that this value is not larger because the parametrization is based on a noisy example (Chapter 5), instead of a noise-free one. The most costly algorithm is ADMM, for all the examples presented in this section, because of the SVD calculation on the large unfolded matrices (inside the shrink function).

We reconstruct the same synthetic volume with a $SNR = 1$ using all five methods. In this case there is a larger difference among the results, therefore we display them as CMP gathers (for a fixed CMP-$y$ and offset-$y$) in Figures 6.1 and 6.2. All panels in both figures have the same clip, so they are a fair visual comparison. The running times and quality for the reconstructions for this case appear in Table 6.1. Cadzow/MSSA gives the best results for this example, followed by tensor completion methods. Naturally, all running times are similar to the previous example.

Our third example is similar to the one presented before, with the exemption that the events contain curvature in all directions (the curvature is separable on all four spatial dimensions). The running times and quality for the reconstructions appear in Table 6.1, as the previous examples. The parameters used for the reconstruction are the same as those for the linear events example above with the exemption that now the reinsertion parameter is equal to 0.9 for HOSVD, 0.3 for SEQ-SVD and 0.8 for Cadzow/MSSA. As expected, tensor-related methods (presented in previous chapters) are the only ones that can achieve a satisfactory

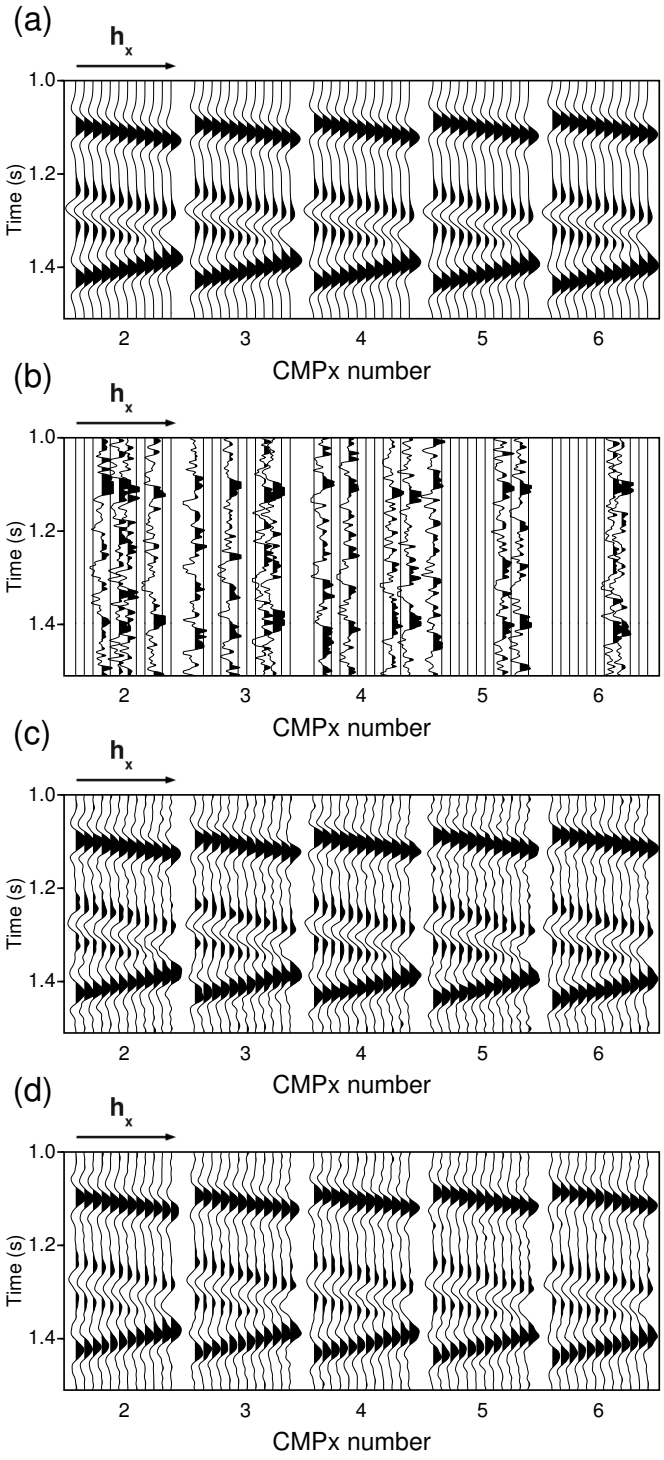



Figure 6.1: (a) Synthetic prestack volume with linear events. The data consist of $12 \times 12 \times 12 \times 12$ bins. A small subset of the data is portrayed. (b) 70% decimated data with $SNR = 1$. (c) Data recovered with HOSVD. (d) Data recovered with SEQ-SVD. Continues on Figure 6.2.

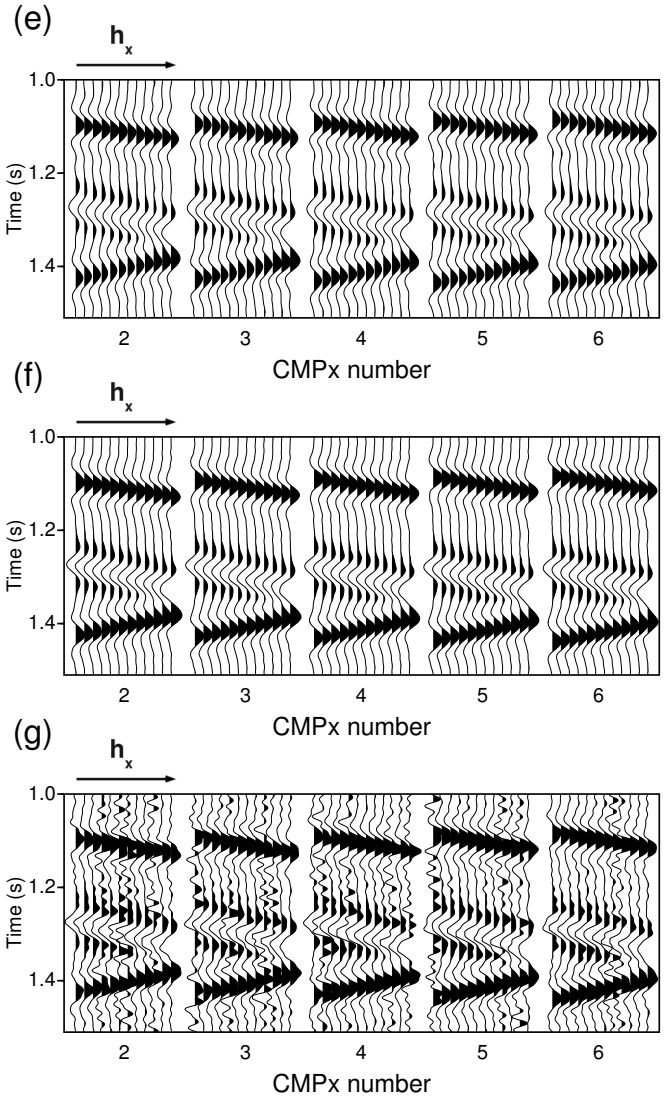



Figure 6.2: Continuation of Figure 6.1. (e) Data recovered with ADMM. (f) Data recovered with Cadzow/MSSA. (g) Data recovered with MWNI.

| Method | Linear $SNR = 100$ | Linear $SNR = 1$ | Curved $SNR = 1$ |
|---|---|---|---|
| HOSVD | 41.8 dB | 15.6 dB | 15.3 dB |
| | 1.3 min | 1.2 min | 1.2 min |
| SEQ-SVD | 42.1 dB | 14 dB | 13.7 dB |
| | 0.2 min | 0.2 min | 0.4 min |
| ADMM | 13.6 dB | 15.6 dB | 12.1 dB |
| | 15.6 min | 18.2 min | 18.1 min |
| Cadzow/MSSA | 30 dB | 21.2 dB | 4.17 dB |
| | 5 min | 5.4 min | 4.6 min |
| MWNI | 21.5 dB | 7.5 dB | 5.5 dB |
| | 0.3 min | 0.3 min | 0.3 min |

Table 6.1: Quality of the reconstruction (in dB) and running times (in minutes) for three different examples for all five methods.

$Q$ in the presence of curvature in the reflections. We notice that Cadzow/MSSA tends to "linearize" the events no matter which rank $r$ we choose. We tried a few combinations of higher values for $r$ and none of them gave a satisfactory result. MWNI seems to better preserve the curvature although it contains a different $SNR$ for original and reconstructed traces, an undesired characteristic in the reconstructed result. As we mentioned previously, MWNI does not contain a reinsertion formula as HOSVD, SEQ-SVD and Cadzow/MSSA algorithms. Furthermore, the conjugate gradients implementation of MWNI is not capable of handling this $SNR$ discrepancy in the same way as the other methods. However, in real data scenarios with a moderate level of noise MWNI provides satisfactory results (refer to Figure 6.10).

## 6.2 Simulation for binning parameters on real data

The choice of the binning parameters is fairly arbitrary in practice. A rule of thumb is to choose the size of the grid in all four dimensions according to the acquisition geometry. In an orthogonal acquisition geometry, the inline/crossline spacing are roughly $\Delta m_x = RI/2, \Delta m_y = SI/2$ (Cordsen et al., 2000), where $RI$ is the receiver interval and $SI$ is the

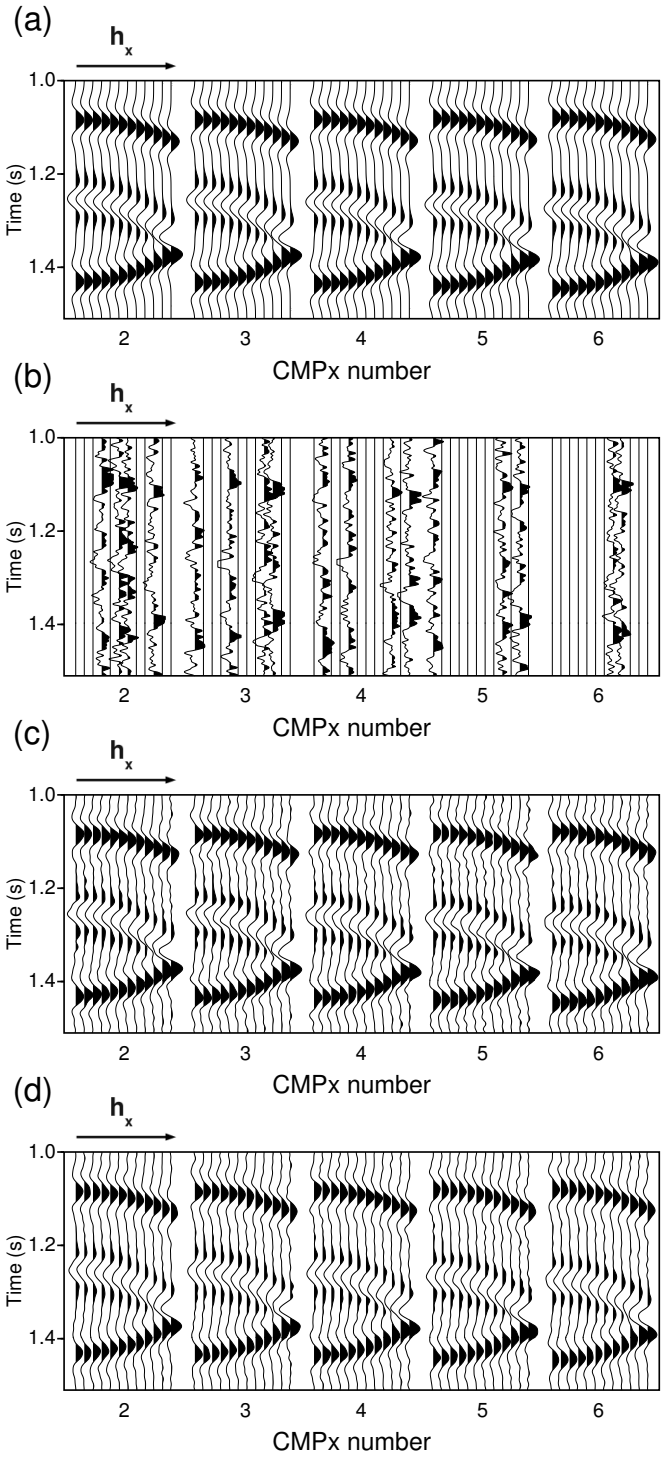



Figure 6.3: (a) Synthetic prestack volume with curved events. The data consist of $12 \times 12 \times 12 \times 12$ bins. A small subset of the data is portrayed. (b) 70% decimated data with $SNR = 1$. (c) Data recovered with HOSVD. (d) Data recovered with SEQ-SVD. Continues on Figure 6.4.

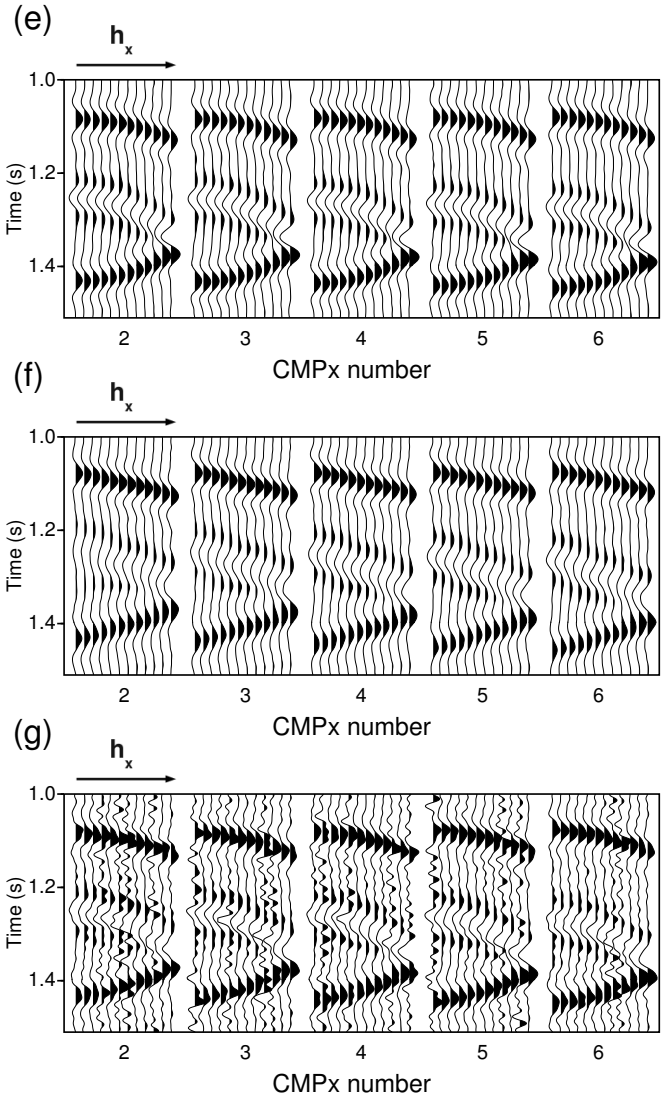



Figure 6.4: Continuation of Figure 6.3. (e) Data recovered with ADMM. (f) Data recovered with Cadzow/MSSA. (g) Data recovered with MWNI.

source interval (refer to Figure 6.5). The choice of $\Delta h$ and $\Delta az$ is not that well defined and seems to be more arbitrary.

We propose two ways to measure the suitability of a grid for reconstruction:

a) Redundancy of the grid: we calculate this as one minus the ratio of the amount of traces used in the binning to the amount of observed traces; it varies between 0 and 1.

b) Grid population: we calculate this as the ratio of the amount of traces used in the binning to the size of the grid; it varies between 0 and 1.

Given a certain grid, if more than one trace falls in the same bin, it is common practice to keep the one that is closer to the center of the bin (in 4D sense) or to average all of them. Ideally, a grid with as little redundancy as possible would be preferred (a bin size that does not discard too many traces when picking one to represent each bin); i.e. the closer to 0 is the ratio in a), the less redundant our grid will be. On the other hand, it would also be preferable to have a grid that has the least amount of empty bins so that reconstruction methods are easier to implement; i.e. the ratio in b) should be as close to 1 as possible. These two criteria are conflicting up to a point. It is natural to expect that the more populated the grid is, the larger the bin size needs to be and therefore the more traces will be discarded in the binning process. If we increase the bin size too much, we tend to smear details and make a larger error in binning and the continuity among nearby traces may be damaged. At the same time, if one wishes to preserve the original locations of the traces as much as possible, one needs a very fine grid that leads to a sampling operator with too many zeroes. It is clear that a compromise between both measures is required to achieve a favorable grid.

To test the aforementioned concepts, we implement a simple simulation on the land data presented in Chapter 3. For this survey, the shot line spacing is 780 m, the shot spacing is 120 m, the receiver line spacing is 500 m and the receiver spacing is 60 m. The grid is designed in the inline-crossline-offset-azimuth domain (different domain than the one used in Chapters 3 and 4) and assumes a standard bin size of 30 m for inline spacing, 50 m for crossline spacing, 100 m for offset (absolute offset) and 24° for the azimuth. These values are extracted according to the original acquisition geometry (Figure 3.13), while the offset and azimuth spacing are chosen to be small but reasonable values according to the acquisition.

After binning for various multiples of the standard bin size, the calculation of the measures a) and b) determines which combination offers a good compromise between them. Figures 6.6 and 6.7 show the inline, crossline, offset and azimuth spacings versus redundancy and grid population. The lines in the plots are densely sampled because every multiple of any coordinate is combined with the rest of the possible coordinates, resulting in a total of 40656 different grid scenarios. Figures 6.6 and 6.7 show the distribution of inline, crossline,

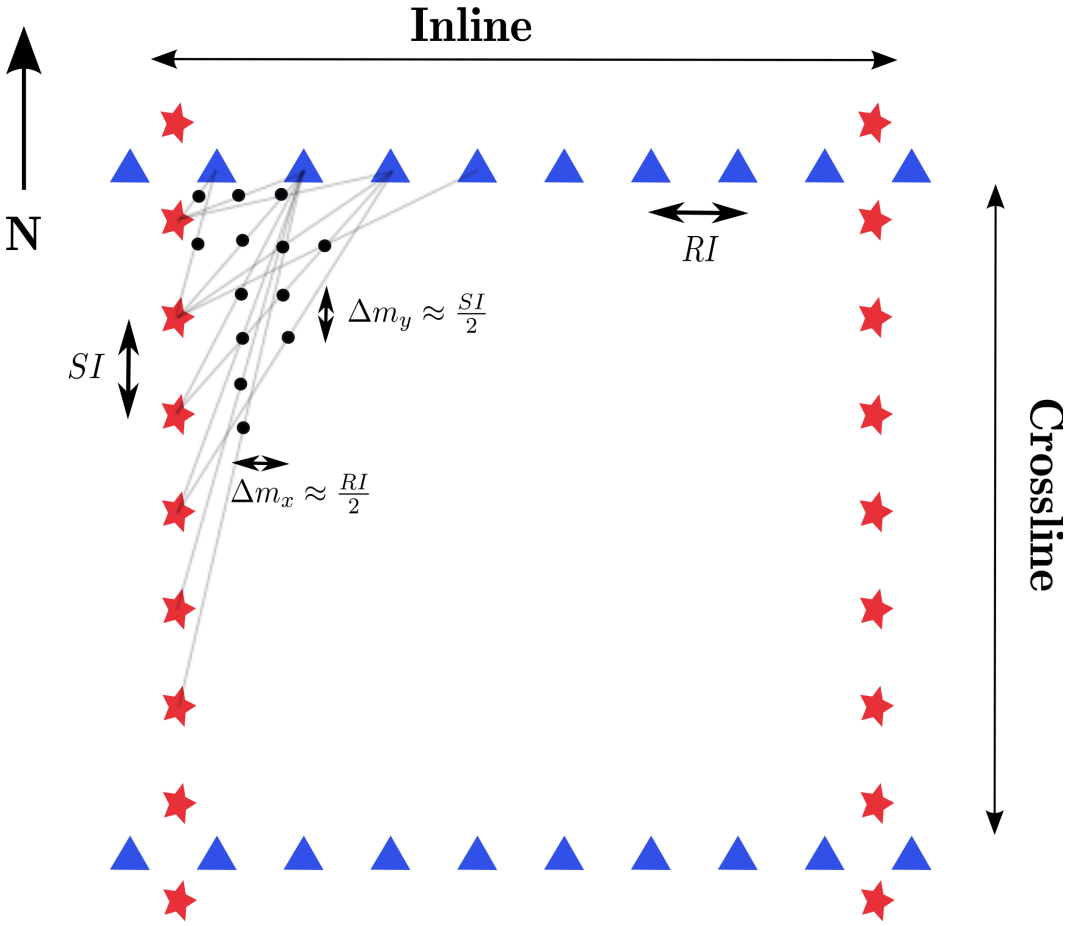
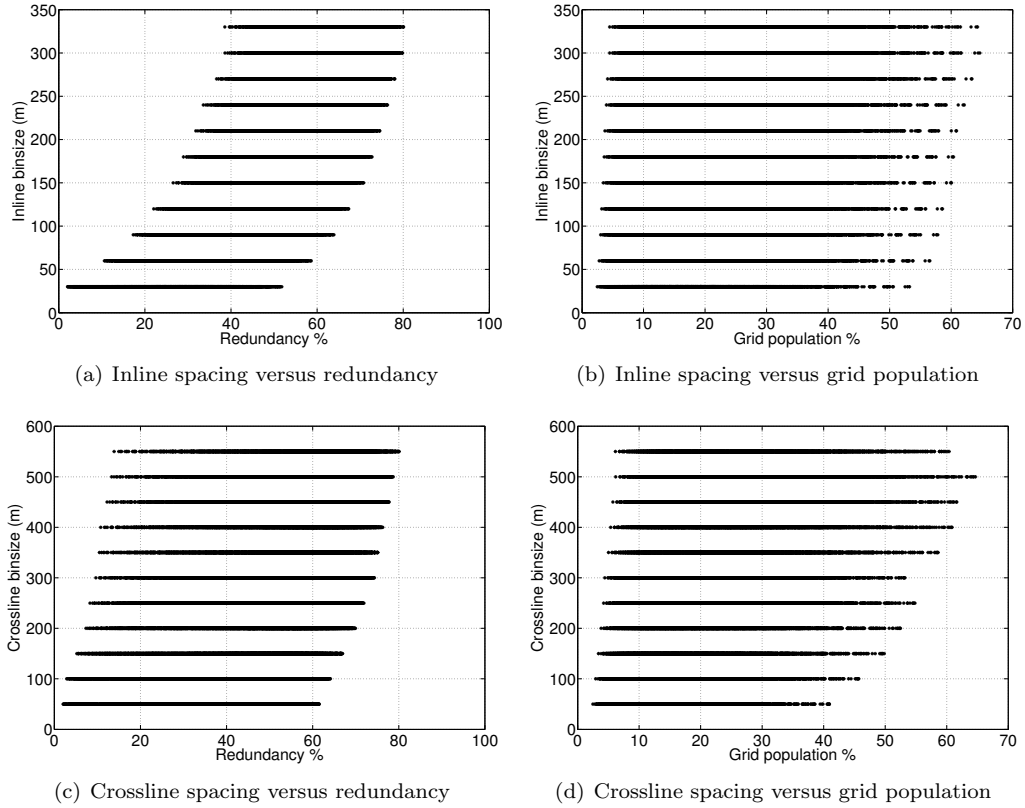



Figure 6.5: Standard bin size according to an orthogonal geometry. The grey shaded lines show the offset between different combinations of source-receivers and the black dots are their respective midpoints. The pattern of midpoints follows approximately the rule $\Delta m_x = RI/2, \Delta m_y = SI/2$, where $RI$ is the receiver interval and $SI$ is the source interval.




(a) Inline spacing versus redundancy

(b) Inline spacing versus grid population

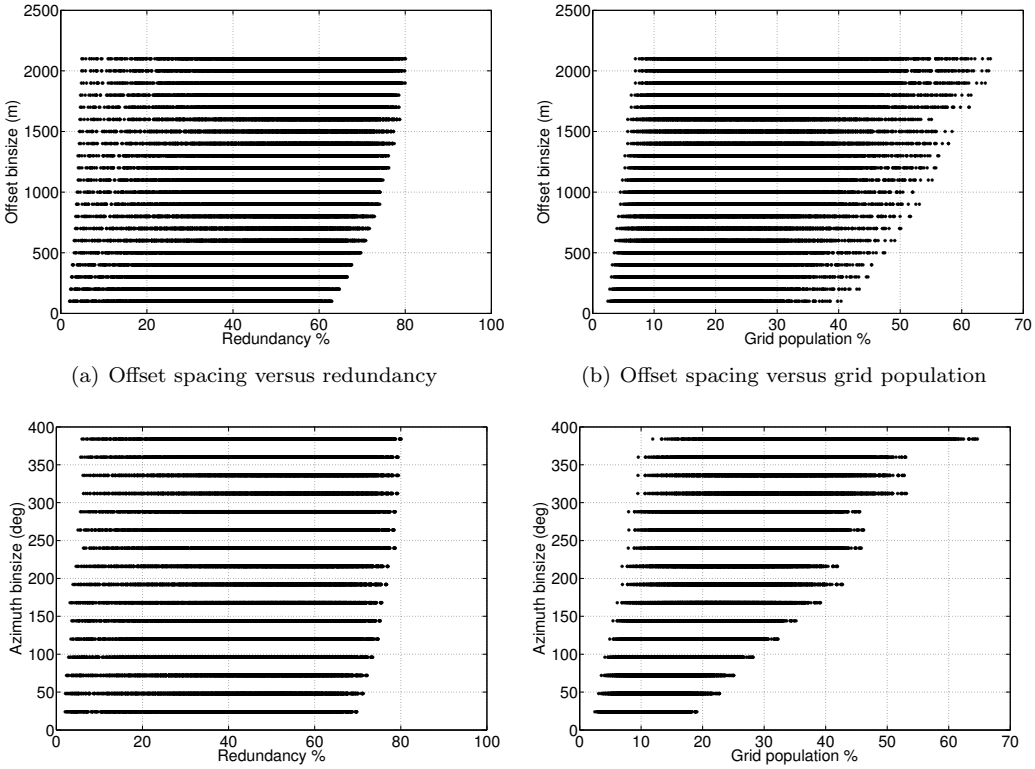
(c) Crossline spacing versus redundancy

(d) Crossline spacing versus grid population

Figure 6.6: Inline and crossline spacing versus redundancy and grid population.

offset and azimuth spacings independently from the values adopted by the other coordinates. Notice that prestack seismic data needs a 4D grid and the redundancy and grid population measures have taken into account all four dimensions in their calculation. However, we can extract some conclusions from these figures. Overall, only the smaller inline/crossline spacings can achieve a redundancy close to 0% which, at the same time, can also lead to a small number for the grid population. Naturally, this leads to a very fine grid in inline/crossline. The offset and azimuth coordinates have a similar behavior. Nevertheless, we can still observe some combinations outside the set of smallest bin sizes that achieve a similar effect on the grid.

We search for grids that simultaneously achieve a redundancy between 0-30% and a grid population larger than 15%. We consider that with a grid population of less than 10-15% reconstruction methods tend to be not too robust. From all the grids filtered by these limits, we apply the specified techniques on some of the grids and we present results from one combination that gave satisfactory results. This grid is of size $26 \times 16 \times 13 \times 8$, where




(a) Offset spacing versus redundancy

(b) Offset spacing versus grid population

(c) Azimuth spacing versus redundancy

(d) Azimuth spacing versus grid population

Figure 6.7: Offset and azimuth spacing versus redundancy and grid population.

$\Delta m_x = 30\,\text{m}$, $\Delta m_y = 100\,\text{m}$, $\Delta h = 200\,\text{m}$ and $\Delta az = 48°$. The fold map in this case (Figure 6.8) is different from the one presented in Figure 3.14, as we are using a different grid. For this grid choice, the grid population is 16% and 72% of the observed traces were used to fill the grid (redundancy is 28%). Figures 6.9-6.10 and Figures 6.11-6.12 contain common midpoint gathers and common offset gathers (for fixed crossline and azimuth) for this example, respectively. This example highlights the sensitivity of the reconstruction to the choice of the grid. We observe similar results in all gathers reconstructed by those methods presented in this thesis. Moreover, gathers reconstructed by tensor methods seem to retain more details than those reconstructed by Cadzow/MSSA and MWNI. As we mentioned in Chapters 3, 4 and 5, the methods based in tensor algebra are robust to curvature, which is not the case with Cadzow/MSSA and MWNI. Some of the reflections in the gathers reconstructed with tensor methods are not completely flat (for example, the reflection at 1.1 s), while we do not recognize the same in gathers e) and f) in Figure 6.10. A similar observation applies to Figures 6.11 and 6.12 (for example, notice those reflections slightly below 1.2 s). With real data we do not know the "true" answer, as we do with synthetic data. Therefore we cannot assess if the best solution arises from the tensor methods or the other methods. The flatness of the reflections given by Cadzow/MSSA seems unrealistic, considering that the input gathers in Figure 6.11 a) (offset number 10) do not show a completely flat signature. The tensor methods have preserved the characteristics of the sampled traces. The same can be said about MWNI up to some extent, while Cadzow/MSSA have changed them to appear flatter.

We show the offset and azimuth distribution for the grid used in the reconstruction in Figures 6.13 and 6.14, respectively. The sampling of the offsets increases with the bin number, except for the last bin. On the other hand, the sampling of the azimuths is more homogeneous than for the offsets, averaging approximately 900 traces for most azimuth bins.

## 6.3 Conclusions

In this chapter we tested the tensor methods introduced in this thesis versus Cadzow/MSSA and MWNI methods. We designed simple synthetic models to evaluate the behavior of the different methods in the presence of high and low $SNR$, sparsity and curvature of the reflections. All the methods showed similar results for the case in which the reflections are linear and contain a high $SNR$. When the $SNR$ is low, MWNI is the only method unable to correctly deal with the $SNR$ discrepancy among observed and reconstructed traces. When the reflections present curvature, we observed that only the tensor methods are capable of achieving a high quality reconstruction. We also evaluated the running times for the different algorithms and found that the most computationally expensive algorithm is ADMM.

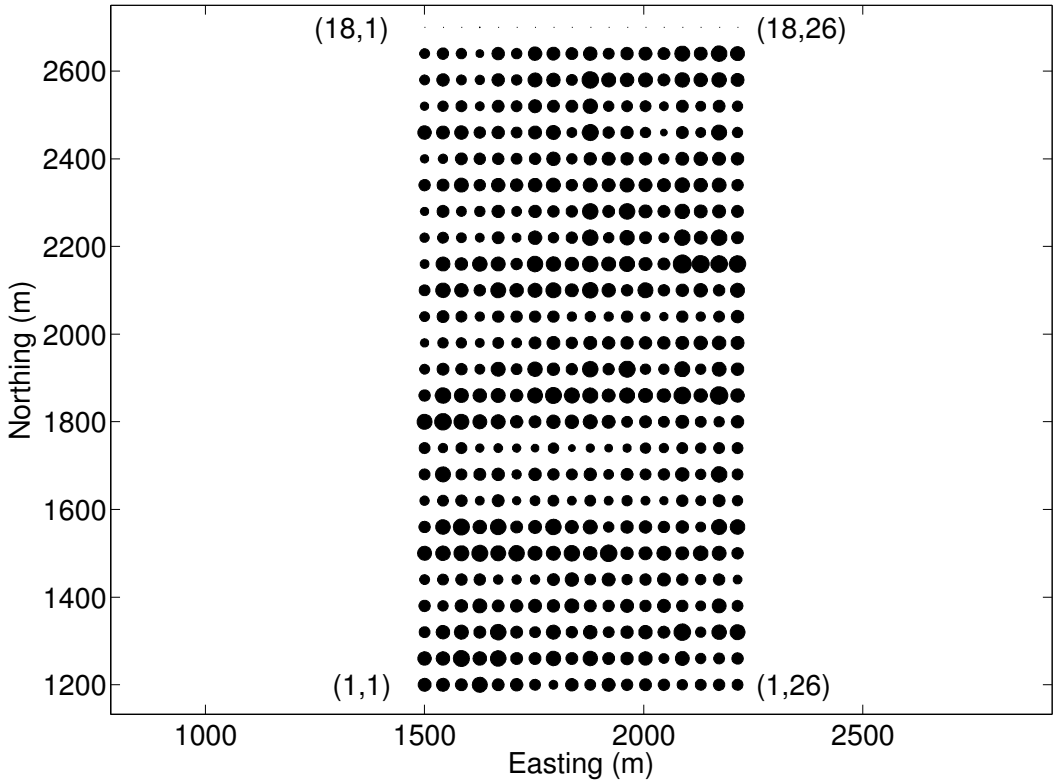



Figure 6.8: Fold map for the chosen grid. The radius of the circles are proportional to the fold. The maximum fold is 35 while the minimum fold is 0 (we observe the minimum in the 1st row of circles in the fold map).

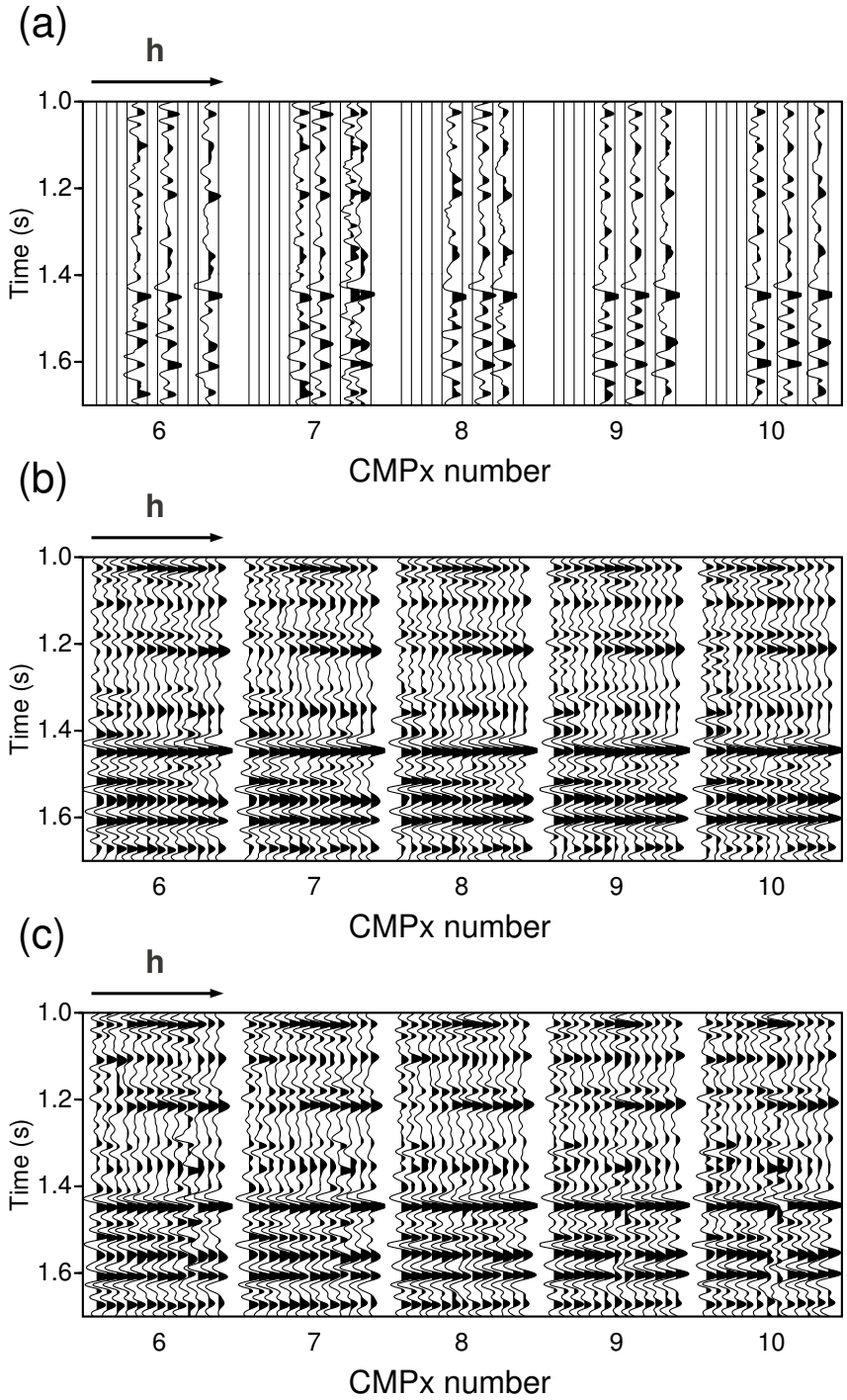



Figure 6.9: Common midpoint gathers for a grid populated 16% and a redundancy of 28%. a) Input gathers. b) Reconstructed gathers using HOSVD. c) Reconstructed gathers using SEQ-SVD. The figure continues in Figure 6.10.

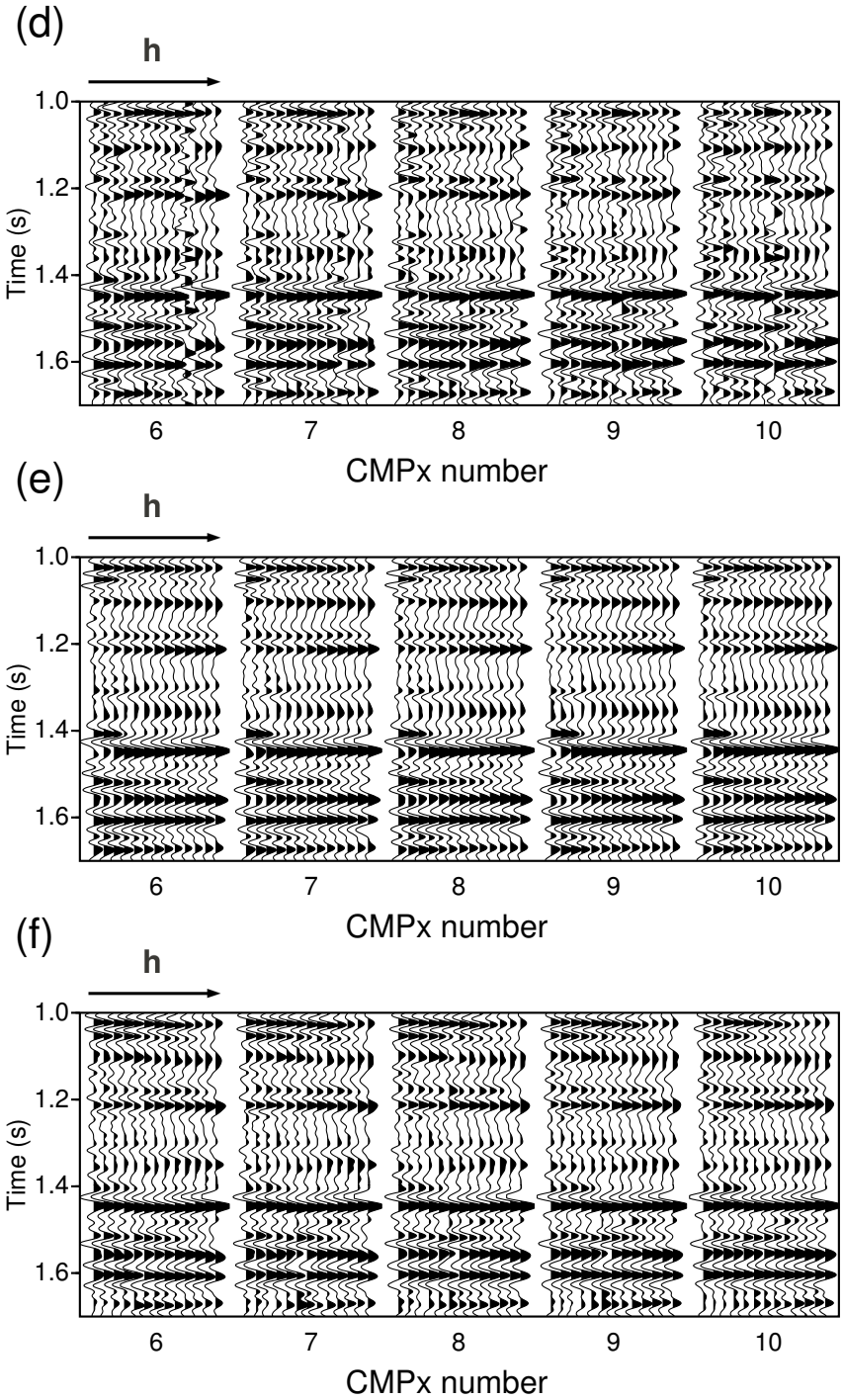



Figure 6.10: This figure is a continuation of Figure 6.9. Common midpoint gathers for a grid populated 16% and a redundancy of 28%. d) Reconstructed gathers using ADMM. e) Reconstructed gathers using Cadzow/MSSA. e) Reconstructed gathers using MWNI.

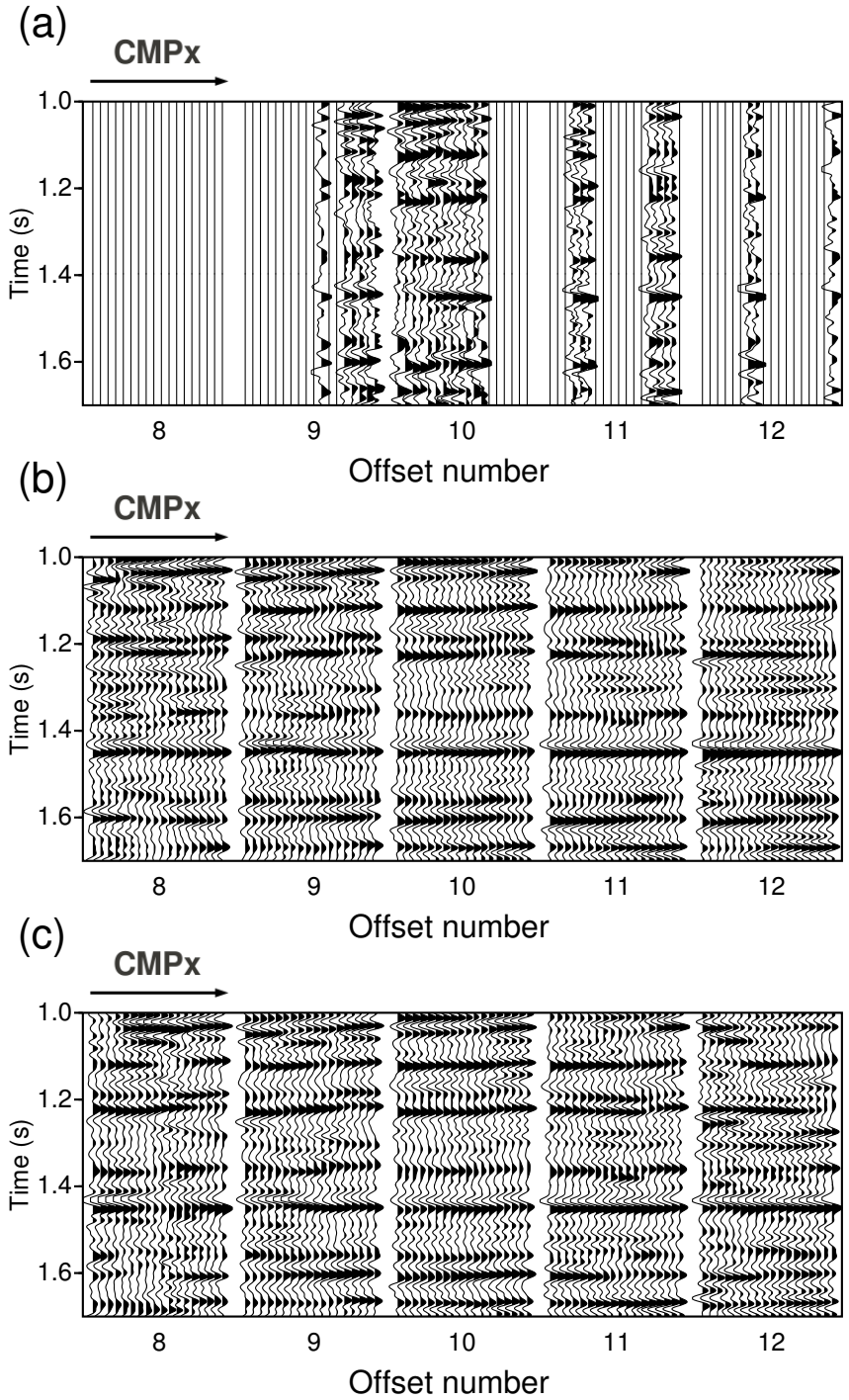



Figure 6.11: Common offset gathers for a grid populated 16% and a redundancy of 28%. a) Input gathers. b) Reconstructed gathers using HOSVD. c) Reconstructed gathers using SEQ-SVD. The figure continues in Figure 6.12.

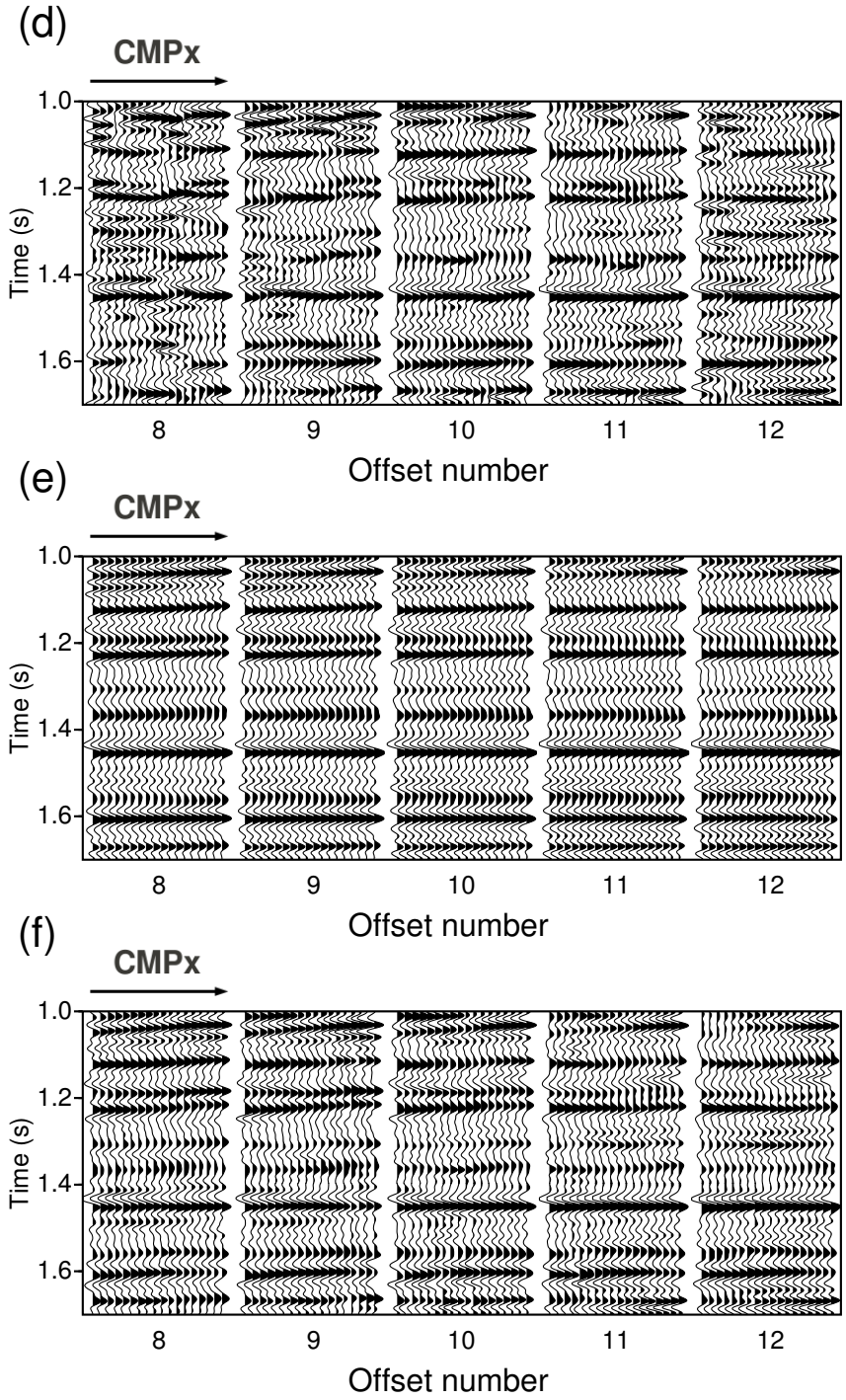



Figure 6.12: This figure is a continuation of Figure 6.11. Common offset gathers for a grid populated 16% and a redundancy of 28%. d) Reconstructed gathers using ADMM. e) Reconstructed gathers using Cadzow/MSSA. e) Reconstructed gathers using MWNI.

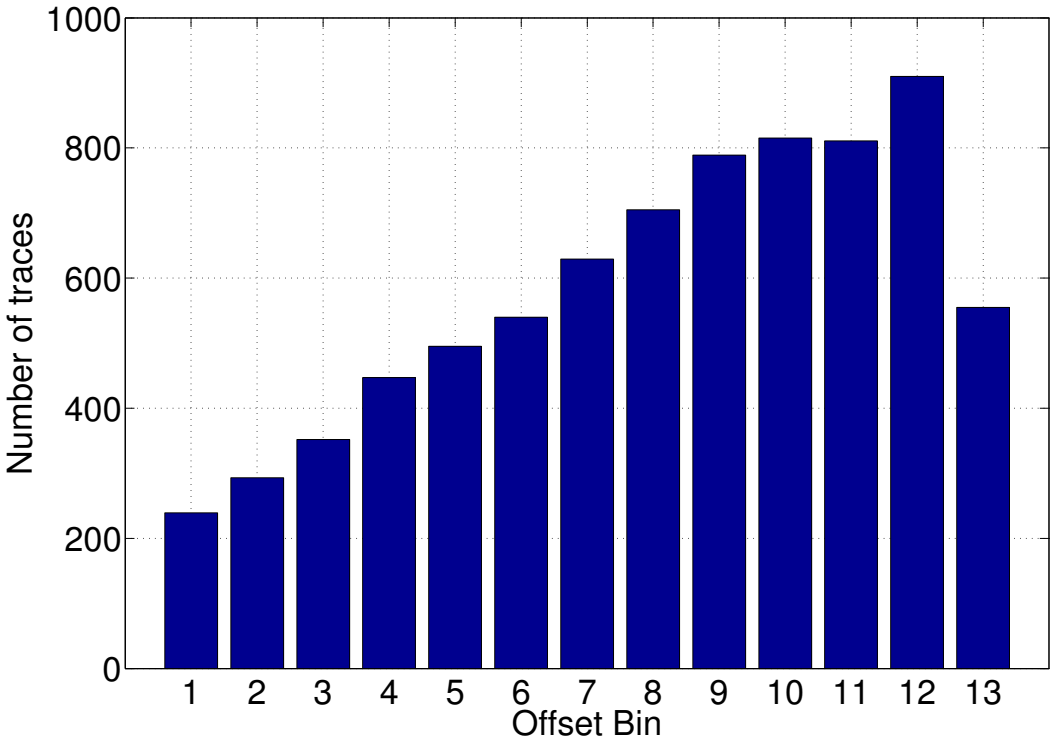



Figure 6.13: Offset distribution for the grid chosen to reconstruct the field data.

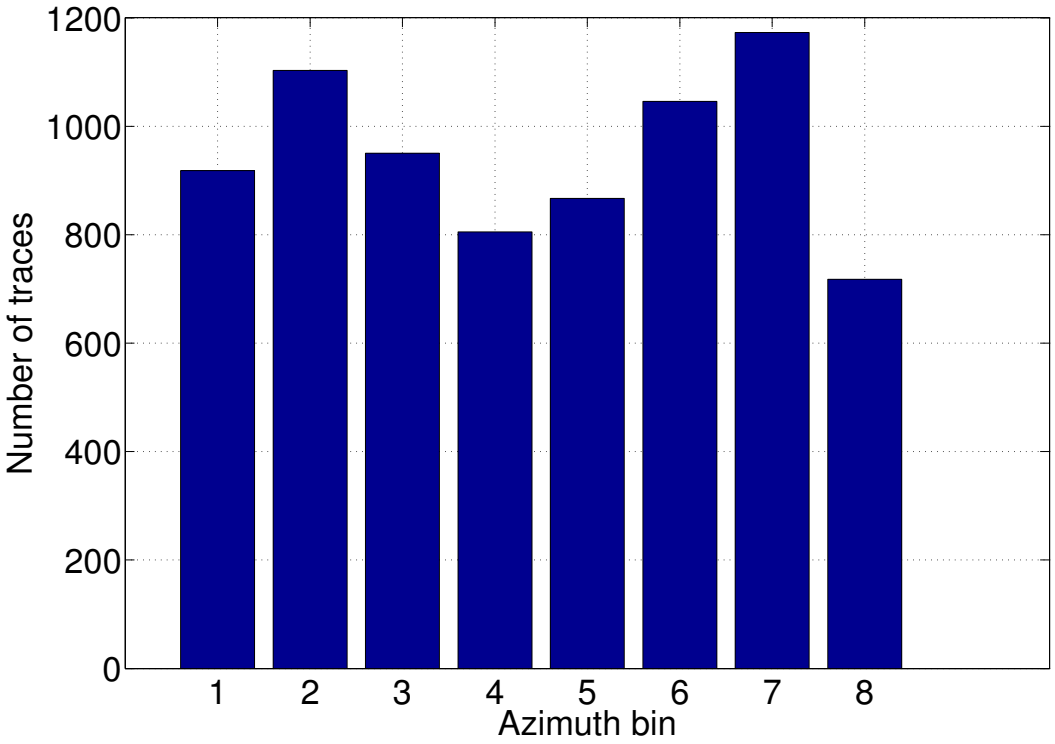



Figure 6.14: Azimuth distribution for the grid chosen to reconstruct the field data.

Reconstruction methods that require binning the data to a regular grid greatly depend on the choice of that grid. We believe that choosing a suitable grid plays a big role on the reconstruction process itself, as these methods are in general sensitive to the data distribution on the grid and the percentage of data that populates the grid. We ran a simulation to obtain well suited grids, with respect to the measures of sparsity and grid redundancy. We implemented reconstruction in one of the grids indicated by our simulation and the results were satisfactory. We noticed some differences between the tensor methods and Cadzow/MSSA and MWNI. In some of the gathers in which we had enough input traces to appreciate the effect, we observed that Cadzow/MSSA and MWNI modified the character of some of the reflections by flattening them (Cadzow/MSSA to a larger extent than MWNI). On the other hand, the tensor methods did not have this effect and preserved the signature of the reflections.

Although not presented in this chapter, we reconstructed the same patch of data on grids not that well suited (with respect to our measures) and confirmed our assumption about the grid sensitivity. This last statement seems to discourage the use of reconstruction methods that require binning (e.g. MWNI, Cadzow, etc.) and encourage the use of methods that can preserve the original trace locations (e.g. ALFT from Xu et al., 2005, and others). The binning process is inconvenient and often damaging for some features of the data that are affected by moving traces from their actual location, but it is necessary for some processes such as reconstruction and multichannel denoising (Yilmaz, 2001, p. 1019). Our simple experiment shows that, starting from the standard bin sizes given by the acquisition geometry, it is very simple and fast to implement a simulation to obtain a few grids that can be suitable for reconstruction.

# CHAPTER 7

# Conclusions

This thesis focuses on the application of tensor algebra techniques to the problem of multidimensional seismic data reconstruction. More precisely, we use tensor decomposition (HOSVD), sequential rank reduction (SEQ-SVD) on the unfoldings of the data tensor and tensor nuclear norm minimization (via ADMM). We have covered both aspects of the tensor multi-rank minimization problem:

- the SEQ-SVD algorithm is closely related to the minimization of the non-convex objective function containing the misfit and the $n$-rank of the tensor, while
- the nuclear norm minimization algorithm deals with the minimization of the objective function containing the closest convex approximation to the $n$-ranks.

The HOSVD algorithm is an iterative approach that exploits the existence of the core-tensor to reduce the rank of the data. We compared the performance of the proposed methods to another rank reduction method (Cadzow/MSSA) and a Fourier-based method (MWNI). All methods presented comparable results except in the presence of curved events, in which the tensor methods predominated. Another noticeable difference was that the tensor methods seemed to better preserve the details of the signal.

## 7.1 Contributions

The contributions of this dissertation are the following:

⋆ We introduce the use of tensors, or multidimensional arrays, in seismic signal processing. Although tensor analysis techniques are common in other areas (chemometrics,

bibliometrics, biomedical signal processing, etc.), only one article has been published in exploration geophysics that utilizes tensor analysis (Vrabie et al., 2006). We envision that tensor algebra can have other applications in seismic signal processing, such as coherent noise removal, denoising and imaging.

⋆ We propose the use of a multidimensional extension of the widely known Singular Value Decomposition (SVD) to tensors, called Higher-Order Singular Value Decomposition (HOSVD, Chapter 3). As the SVD contains a diagonal matrix with the singular values of the decomposed matrix in its diagonal, the HOSVD contains a core-tensor that plays a similar role. Although the core-tensor is not hyper-diagonal and is actually a full tensor, we can perform rank reduction on the decomposed tensor by truncating the core-tensor (as we would do with the diagonal matrix in the SVD). This procedure followed by a modified imputation algorithm is what we call HOSVD-based reconstruction. By implementing it iteratively, the imputation algorithm allows the reconstruction of the missing traces while balancing the $SNR$ of original and reconstructed traces. A novelty of this method is that it is quite robust to curvature and to low signal-to-noise ratios. Most reconstruction methods resort to the use of windows to avoid local changes of dip. With HOSVD this is unnecessary.

⋆ We suggest a rank reduction strategy different from the truncated HOSVD. The sequential-SVD (SEQ-SVD) uses the matrix unfoldings of the data tensor and applies a hard thresholding of the singular values of each matrix sequentially (one unfolding after the other). This algorithm uses the same modified imputation algorithm for the reconstruction as the HOSVD method. For the noise-free case, we found that SEQ-SVD is equivalent to solving the non-convex problem of tensor rank minimization one unfolding at a time. As HOSVD-based reconstruction, SEQ-SVD is quite robust to curvature and to low signal-to-noise ratios.

⋆ We use the nuclear norm to approximate the non-convex problem of tensor $n$-rank minimization (Fazel et al., 2001). The alternating direction method of multipliers (ADMM) is the algorithm used in the minimization of the cost function of the problem. The solution to the rank minimization problem is a low-rank solution that reveals the best rank possible subject to the constraints. Although more computationally expensive than the HOSVD approach, this method solves the inverse problem of tensor completion via optimization and a theorem guarantees its convergence. Similarly to the HOSVD and the SEQ-SVD methods, ADMM is robust to curvature of the reflections and to low signal-to-noise ratios.

⋆ We compared the tensor completion methods introduced in this thesis with Cadzow/MSSA and MWNI. To make the comparison as fair as possible, we used the

parameters suggested by Gao et al. (2013) for Cadzow/MSSA and we obtained parameters from synthetic tests for MWNI (we chose those parameters that gave the best results). However, we acknowledge that we lack industrial codes for this methods, which tend to be more robust than codes designed for academic purposes. We performed simple synthetic tests to evaluate the difference between these methods in the presence of linear events, curved events, sparsity and low/high $SNR$. We measured running times and quality of the reconstruction for all the examples. All methods, except MWNI, were robust to noise and only the tensor methods were robust to curvature. Additionally, we run a simple simulation over the binning parameters from a real dataset in the Western Canadian Basin (for a grid in the inline-crossline-offset-azimuth domain) to obtain a grid size that achieves a good compromise between redundancy and grid population. We chose one "optimal" grid (that balances the redundancy and grid population for the given grid) and reconstructed the data with all the methods. The tests with real data indicated a clear difference between the tensor methods and the other two, especially in the presence of not completely flat events. While Cadzow/MSSA resulted in flat reflections with a high $SNR$, the tensor methods seemed to better honor the sampled traces, verified where traces were available, and retained more details or noise.

## 7.2 Limitations and future work

**Missing fiber/slice**

The main drawback from using a tensor completion approach for the reconstruction of prestack data is that these strategies are unable to reconstruct missing tensor slices and missing fibers. Candès and Recht (2009) mention this limitation in the matrix completion case. They claim that a matrix cannot be recovered when a row or a column is completely missing and that we need at least one observation from a row or column to be able to recover it. Furthermore, they prove that reconstruction algorithms perform optimally when the samples are randomly missing, as it is the case in the theory of compressive sensing (Donoho, 2006). In general, matrix completion makes use of the nuclear norm, the "tightest convex relaxation" of the rank of a matrix (Fazel et al., 2001), to minimize using a convex optimization algorithm. Minimizing the nuclear norm of a matrix is equivalent to making the singular values of that matrix sparse. When the sampled observations of the matrix are such that one full column or row is missing, the minimum of the nuclear norm is not a sufficient assumption to recover them. The reason for this is that the singular value distribution from the incomplete matrix is already sparse (i.e. the matrix is already low-rank) in the presence of this missing row or column. Intuitively, we can think of "which is the matrix with lowest energy that can explain the missing row/column pattern?". The answer is "a matrix that

contains zeros in that same row/column". Obviously, any non-zero elements placed instead of zeros in that row/column increases the energy measured by the nuclear norm.

In the tensor completion case, we encounter the same problem in higher-order dimensions. Additionally, to make the problem worse, in our prestack data reconstruction application we rarely encounter randomly missing samples. The missing row/column problem for matrices manifests itself in the missing fibers and missing slices for tensors. Let us consider the midpoint-offset-azimuth domain. It is often the case, especially in land data acquisition, that a certain azimuth or offset is not sampled in the whole patch to be reconstructed (because of the regularity of the orthogonal geometry). This problem can be alleviated by using grids that are not too fine, although these grids can bring other problems. The reconstruction is still going to recover other parts of the tensor in which we have at least one sample in that dimension, but it will fail in recovering a completely missing azimuth and/or offset. Narita et al. (2011) addressed this problem by using additional information when using ALS in CP and Tucker decompositions. The additional information is added as an extra regularization term in the ALS objective function in the form of what they refer as "similarity matrices". These matrices have some measure of the relationship between different dimensions (for instance, smoothness, if adjacent dimensions behave similarly). Their experiments consider the missing fiber/slice problem and they show a significant uplift with respect to classical ALS methods, which ignore these problems, as well as in the cases when having extremely sparse observations. We believe that such a strategy could be implemented in our applications as well, although more research is needed to understand which property could be used as a similarity matrix.

**Spatial aliasing**

All the methods proposed in this thesis have not been designed as de-alias methods, compared to methods such as the ones by Spitz (1991) and Naghizadeh and Sacchi (2010c). We refer to the type of alias known as spatial alias, originating from regularly missing samples or regular decimation (e.g. inline/crossline infilling/upsampling problem). In the absence of any other constraint, the low-rank hypothesis is not enough to dealias seismic data because, in the $F - X$ domain, the aliased events have the same amplitude as the original events. Therefore, reconstruction via rank reduction will recover both original and aliased events. However, a similar modification to the one introduced by Naghizadeh and Sacchi (2013) for Cadzow/MSSA could be implemented in the tensor methods.

**Binning**

Binning is the process through which seismic data is mapped to a regular grid in a certain domain (for instance, inline-crossline-offset-azimuth). Another seismic processing method that requires binning is multichannel denoising (such as $F-X$ or $F-X-Y$ deconvolution). Binning is a far from ideal procedure, as it relocates the traces to a different location from

the original one. Once we choose the bin size (in 4D sense), we group the traces according to those bins and "create" a new dataset whose traces locations are now at the center of the bin. Because we cannot have more than one trace per bin, we generally choose the trace falling the closest to the bin center as the "representative" trace for that bin. This is the reason why choosing a grid that does not imply discarding too many original traces is ideal (refer to Chapter 6). Binning tends to distort the small features of the signal as it displaces the trace from its original location, although this can be minimized by choosing a grid size that is small enough (and this brings other problems such as large grid sparsity, see next paragraph) and by applying NMO-correction prior to binning. Reconstruction methods that use original trace locations, such as the antileakage Fourier transform (Xu et al., 2005) seem the only option to avoid binning. However, it is interesting to notice that MWNI, a well known reconstruction method widely used in industrial applications, is used intensively despite its need for binning. The simulation of binning parameters in Chapter 6 was a toy example attempt at minimizing the redundancy of the grid while keeping a reasonable portion of the grid populated. A similar exercise could be implemented for other real data examples. Incorporating an operator that can map more efficiently from original to binned trace locations is a missing task in this thesis.

**Extreme sampling sparsity**

The tensor based methods presented in this thesis will not work properly when trying to reconstruct seismic data in cases of extreme sparsity. Real data scenarios often lead to reconstruction with only 2-3% of the grid being populated. These "extreme" levels of sparsity imply that at least one fiber or slice will be completely missing in the 4D spatial tensor. Although a strategy such as the one proposed by Narita et al. (2011) might alleviate this problem, it embodies the biggest limitation of these methods.

**Acquisition designed for reconstruction**

The best scenario for a reconstruction method is given when the data sampling satisfies the method's optimal conditions. This condition is rarely satisfied. We could try to approximate the "ideal" conditions by having an acquisition that aims at satisfying them. For the tensor methods of this thesis, the ideal acquisition would need to be such that, for example, in the inline-crossline-offset-azimuth domain the traces are randomly sampled. However, random sampling in the inline-crossline-offset-azimuth domain would not lead to an "organized" (traditional) sampling in the source-receiver domain. Jittered sampling (Hennenfent and Herrmann, 2008) might offer a good alternative as well as the work from Sacchi et al. (2013), although we have not explored these possibilities in this thesis.

## 7.3 Final remarks

This thesis presents for the first time reconstruction algorithms for multidimensional seismic data that are based on multilinear algebra. Reconstruction algorithms based on waveform synthesis (MWNI, POCS, ALFT, curvelets) have been tested with different degrees of success. My contribution is to introduce tensor algebra to the problem of 5D seismic data reconstruction. Although more testing is needed to truly understand the limitations of these tools, I believe that tensor algebra can positively contribute to the field of seismic processing. These techniques belong to a newly opened and buoyant field in applied mathematics. I envision that tensor completion tools can benefit seismic exploration where multidimensional data processing is required.

# Bibliography


Abma, R., and N. Kabir, 2005, Comparison of interpolation algorithms: The Leading Edge, **24**, no. 10, 984–989.

———, 2006, 3D interpolation of irregular data with a POCS algorithm: Geophysics, **71**, no. 6, E91–E97.

Abma, R., C. Kelley, and J. Kaldy, 2007, Sources and treatments of migration-introduced artifacts and noise: SEG Technical Program Expanded Abstracts, 2349–2353.

Acar, E., and B. Yener, 2009, Unsupervised multiway data analysis: A literature survey: IEEE Transactions on Knowledge and Data Engineering, **21**, no. 1, 6 –20.

Andersson, C. A., and R. Bro, 1998, Improving the speed of multi-way algorithms: Part I. Tucker3: Chemometrics and Intelligent Laboratory Systems, **42**, 93 – 103.

Arrow, K. J., L. Hurwicz, and U. Uzawa, 1968, Studies in non-linear programming: Stanford University Press.

Bader, B. W., and T. G. Kolda, 2006, Algorithm 862: MATLAB tensor classes for fast algorithm prototyping: ACM TRans. Math. Softw., 635–653.

———, 2010, MATLAB Tensor Toolbox Version 2.4: http://csmr.ca.sandia.gov/ tgkolda/TensorToolbox/.

Baumstein, A., and M. T. Hadidi, 2004, 3D SRME: data reconstruction and application to field data: SEG Technical Program Expanded Abstracts, 1253–1256.

Bergqvist, G., and E. Larsson, 2010, The higher-order singular value decomposition: Theory and an application [lecture notes]: Signal Processing Magazine, IEEE, **27**, no. 3, 151 –154.

Bertsekas, D. P., 1996, Constrained optimization and Lagrange multiplier methods: Athena Scientific.

Bertsekas, D. P., and J. N. Tsitsiklis, 1989, Parallel and distributed computation: Numerical methods: Prentice Hall.

Biondi, B., 2001, Kirchhoff imaging beyond aliasing: Geophysics, **66**, 654–666.

Biondi, B., and G. Palacharla, 1996, 3-D prestack migration of common-azimuth data: Geophysics, **61**, 1822–1832.

Boyd, S., N. Parikh, E. Chu, B. Peleato, and J. Eckstein, 2011, Distributed optimization

and statistical learning via the alternating direction method of multipliers: Foundations and Trends in Machine Learning, **3**, no. 1, 1–122.

Burroughs, L., and S. Trickett, 2009, Prestack rank-reducing noise suppression: Practice: SEG, Expanded Abstracts, **28**, no. 1, 3337–3341.

Cadzow, J., 1988, Signal enhancement - a composite property mapping algorithm: IEEE Trans. on Acoustics, Speech and Signal Processing, **36**, 49–62.

Cai, J., S. Dong, M. Guo, S. Sen, J. Ji, B. Wang, Z. Li, and S. yong Suh, 2009a, Some aspects on data interpolation: Multiple prediction and imaging: SEG Technical Program Expanded Abstracts 2009, 3178–3182.

Cai, J., M. Guo, S. Sen, S. Dong, R. Camp, and B. Wang, 2009b, From SRME or wave-equation extrapolation to SRME and wave-equation extrapolation: SEG Technical Program Expanded Abstracts, 3133–3137.

Cai, J.-F., E. J. Candès, and Z. Shen, 2010, A singular value thresholding algorithm for matrix completion: SIAM Journal on Optimization, **20**, no. 4, 1956–1982.

Candès, E., and Y. Plan, 2010, Matrix completion with noise: Proceedings of the IEEE, **98**, no. 6, 925 –936.

Candès, E., and B. Recht, 2009, Exact matrix completion via convex optimization: Foundations of Computational Mathematics, **9**, no. 6, 717–772.

Carroll, J., and J.-J. Chang, 1970, Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition: Psychometrika, **35**, no. 3, 283–319.

Cary, P. W., 2011, Aliasing and 5D interpolation with the MWNI algorithm: SEG Technical Program Expanded Abstracts, 3080–3084.

Cattell, R. B., 1944, "Parallel proportional profiles" and other principles for determining the choice of factors by rotation: Psychometrika, **9**, no. 4, 267–283.

Claerbout, J., 1971, Toward a unified theory of reflector mapping: Geophysics, **36**, 467–481.

Comon, P., J. ten Berge, L. D. Lathauwer, and J. Castaing, 2009, Generic and typical ranks of multi-way arrays: Linear Algebra and its Applications, **430**, 2997 – 3007.

Cordsen, A., M. Galbraith, and J. Peirce, 2000, Planning land 3-D seismic surveys: Society of Exploration Geophysicists.

De Lathauwer, L., B. De Moor, and J. Vandewalle, 2000a, A multilinear singular value decomposition: SIAM J. Matrix Anal. Appl., **21**, no. 4, 1253–1278.

———, 2000b, On the best rank-1 and rank-$(R_1, R_2, ..., R_N)$ approximation of higher-order tensors: SIAM J. Matrix Anal. Appl., **21**, no. 4, 1324–1342.

De Silva, V., and L.-H. Lim, 2008, Tensor rank and the ill-posedness of the best low-rank approximation problem: SIAM J. Matrix Analysis Applications, **30**, no. 3, 1084–1127.

Deregowski, S. M., 1991, What is DMO?: Dip Moveout Processing, A1–A18.

Dologlou, I., J. Pesquet, and J. Skowronski, 1996, Projection-based rank reduction algor-

tihms for multichannel modelling and image compression: Signal Processing, **48**, 97–109.

Donoho, D. L., 2006, Compressed sensing: IEEE Transactions on Information Theory, **52**, no. 4, 1289–1306.

Duijndam, A., and M. Schonewille, 1999, Nonuniform fast fourier transform: Geophysics, **64**, 539–551.

Eckart, C., and G. Young, 1936, The approximation of one matrix by another of lower rank: Psychometrika, **1**, 211–218.

Eiken, O., G. Haugen, M. Schonewille, and A. Duijndam, 2003, A proven method for acquiring highly repeatable towed streamer seismic data: Geophysics, **68**, 1303–1309.

Fazel, M., H. Hindi, and S. Boyd, 2001, A rank minimization heuristic with application to minimum order system approximation: Proceedings of the 2001 American Control Conference, **6**, 4734 –4739.

Fomel, S., 2003, Seismic reflection data interpolation with differential offset and shot continuation: Geophysics, **68**, no. 2, 733–744.

———, 2011, Theory of 3-D angle gathers in wave-equation seismic imaging: Journal of Petroleum Exploration and Production Technology, **1**, no. 1, 11–16.

Fomel, S., L. Ying, and X. Song, 2010, Seismic wave extrapolation using lowrank symbol approximation: SEG Technical Program Expanded Abstracts, **29**, no. 1, 3092–3096.

Freire, S., and T. Ulrych, 1988, Application of singular value decomposition to vertical seismic profiling: Geophysics, **53**, no. 6, 778–785.

Gabay, D., and B. Mercier, 1976, A dual algorithm for the solution of nonlinear variational problems via finite element approximation: Computers & Mathematics with Applications, **2**, no. 1, 17 – 40.

Gandy, S., B. Recht, and I. Yamada, 2011, Tensor completion and low-n-rank tensor recovery via convex optimization: Inverse Problems, **27**, no. 2, 025010.

Gao, J., M. Sacchi, and X. Chen, 2013, A fast reduced-rank interpolation method for prestack seismic volumes that depend on four spatial dimensions: Geophysics, **78**, no. 1, V21–V30.

Garey, M. R., and D. S. Johnson, 1990, Computers and Intractability: A guide to the theory of NP-Completeness: W. H. Freeman & Co.

Gierse, G., D. Otto, A. Berhorst, H. Trappe, and J. Pruessmann, 2010, CRS technique for advanced prestack merging and regularisation of vintage 3D seismic data: SEG Technical Program Expanded Abstracts, 3624–3628.

Glowinski, R., and A. Marrocco, 1975, Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité, d'une classe de problèmes de Dirichlet non linéaires: Rev. Française Automat. Informat. Recherche Opérationnelle, **9**, no. R-2, 41–76.

Goldfarb, D., and S. Ma, 2011, Convergence of fixed-point continuation algorithms for matrix rank minimization: Foundations of Computational Mathematics, **11**, no. 2, 183–

210.

Golub, G., and C. Van Loan, 1996, Matrix computations: Johns Hopkins University Press. Johns Hopkins Studies in the Mathematical Sciences.

Gray, D., and S. Wang, 2009, Towards an optimal workflow for azimuthal AVO: CSPG CSEG CWLS Convention Expanded Abstracts, 1–4.

Gray, S. H., J. Etgen, J. Dellinger, and D. Whitmore, 2001, Seismic migration problems and solutions: Geophysics, **66**, no. 5, 1622–1640.

Grung, B., and R. Manne, 1998, Missing values in principal component analysis: Chemometrics and Intelligent Laboratory Systems, **42**, 125 – 139.

Gulunay, N., 2003, Seismic trace interpolation in the Fourier transform domain: Geophysics, **68**, no. 1, 355–369.

Hackbusch, W., 2012, Tensor spaces and numerical tensor calculus: Springer. Springer Series in Computational Mathematics, Vol 42.

Halko, N., P.-G. Martinsson, Y. Shkolnisky, and M. Tygert, 2011a, An algorithm for the principal component analysis of large data sets: SIAM Journal on Scientific Computing, **33**, no. 5, 2580–2594.

Halko, N., P. G. Martinsson, and J. A. Tropp, 2011b, Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions: SIAM Review, **53**, no. 2, 217–288.

Hanson, R., R. Jones, T. Langston, B. Sharp, J. Stein, J. Wang, J. Weigant, and D. Corrigan, 2006, A production implementation of data regularization and 3D SRME for complex multiple elimination: SEG Technical Program Expanded Abstracts, 2649–2653.

Harshman, R. A., 1970, Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis: UCLA Working Papers in Phonetics, **16**, no. 1, 84.

Hastie, T., R. Tibshirani, and J. H. Friedman, 2001, The elements of statistical learning: data mining, inference, and prediction: New York : Springer. Springer series in statistics.

He, B., H. Yang, and S. Wang, 2000, Alternating Direction Method with self-adaptive penalty parameters for monotone variational inequalities: Journal of Optimization Theory and Applications, **106**, no. 2, 337–356.

Hennenfent, G., L. Fenelon, and F. J. Herrmann, 2010, Nonequispaced curvelet transform for seismic data reconstruction: A sparsity-promoting approach: Geophysics, **75**, no. 6, WB203–WB210.

Hennenfent, G., and F. J. Herrmann, 2008, Simply denoise: Wavefield reconstruction via jittered undersampling: Geophysics, **73**, no. 3, V19–V28.

Hestenes, M. R., 1969, Multiplier and gradient methods: Journal of Optimization Theory and Applications, **4**, no. 5, 303–320.

Hestenes, M. R., and E. Stiefel, 1952, Methods of conjugate gradients for solving linear

systems: Journal of Research of the National Bureau of Standards, **49**, no. 6, 409–436.

Hitchcock, F. L., 1927, The expression of a tensor or a polyadic as a sum of products: Journal of Mathematics and Physics, **6**, 164–189.

Hunt, L., J. Downton, S. Reynolds, S. Hadley, D. Trad, and M. Hadleyi, 2010, The effect of interpolation on imaging and AVO: A Viking case study: Geophysics, **75**, no. 6, WB265–WB274.

Jain, P., R. Meka, and I. S. Dhillon, 2010, Guaranteed Rank Minimization via Singular Value Projection: Proceedings of the Neural Information Processing Systems Conference (NIPS), 937–945.

Kabir, M. N., and D. Verschuur, 1995, Restoration of missing offsets by parabolic Radon transform: Geophysical Prospecting, **43**, 347–368.

Kamnitzer, J., 2011, Tensor products, notes from MAT247, University of Toronto.

Kaplan, S. T., M. Naghizadeh, and M. D. Sacchi, 2010, Data reconstruction with shot-profile least-squares migration: Geophysics, **75**, WB121–WB136.

Karush, W., 1939, Minima of functions of several variables with inequalities as side constraints: Master's thesis, Dept. of Mathematics, Univ. of Chicago.

Kiers, H. A. L., 2000, Towards a standardized notation and terminology in multiway analysis: Journal of Chemometrics, **14**, 105–122.

Kolda, T., 2001, Orthogonal tensor decompositions: SIAM Journal on Matrix Analysis and Applications, **23**, 243–255.

Kolda, T. G., 2006, Multilinear operators for higher-order decompositions: Technical report, Sandia National Laboratories, Albuquerque, NM and Livermore, CA.

Kolda, T. G., and B. W. Bader, 2009, Tensor decompositions and applications: SIAM Review, **51**, no. 3, 455–500.

Kreimer, N., and M. D. Sacchi, 2012a, Rank reduction of unfolded tensors for pre-stack de-noising and reconstruction: The Recorder, **37**, no. 9, 24–27.

———, 2012b, Reconstruction of seismic data via tensor completion: Statistical Signal Processing Workshop (SSP), 2012 IEEE, 29–32.

———, 2012c, Tensor completion via nuclear norm minimization for 5D seismic data reconstruction: SEG Technical Program Expanded Abstracts, 1–5.

———, 2012d, A tensor higher-order singular value decomposition for prestack seismic data noise reduction and interpolation: Geophysics, **77**, no. 3, V113–V122.

———, 2013, Nuclear norm minimization and tensor completion in exploration seismology: Presented at the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

Kruskal, J. B., 1977, Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics: Linear Algebra and its Applications, **18**, 95 – 138.

———, 1983, Statement of some current results about three-way arrays: manuscript, AT&T Bell Laboratories, Murray Hill, NJ. (Available at http://three-mode.leidenuniv.nl/pdf/k/kruskal1983.pdf).

———, 1989, Rank, decomposition, and uniqueness for 3-way and $N$-way arrays, *in* Multiway data analysis: North-Holland, 7–18.

Kuhn, H. W., and A. W. Tucker, 1951, Nonlinear programming: Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, 1950, University of California Press, 481–492.

Laub, A., 2004, Matrix analysis for scientists and engineers: Society for Industrial and Applied Mathematics.

Lay, T., and T. Wallace, 1995, Modern global seismology: Elsevier Science. International Geophysics.

Leggott, R. J., R. G. Williams, and M. Skinner, 1999, Co-location of 4D seismic data in the presence of navigational and timing errors: SEG Technical Program Expanded Abstracts, 1699–1702.

Levin, F., 1971, Apparent velocity from dipping interface reflections: Geophysics, **36**, 510–516.

Levin, J., 1963, Three-mode factor analysis: PhD thesis, University of Illinois, Urbana.

Little, R. J. A., and D. B. Rubin, 1987, Statistical analysis with missing data: John Wiley and Sons, New York, NY, USA. Wiley series in probability and mathematical statistics.

Liu, B., 2004, Multi-dimensional reconstruction of seismic data: PhD thesis, University of Alberta.

Liu, B., and M. D. Sacchi, 2004, Minimum weighted norm interpolation of seismic records: Geophysics, **69**, no. 6, 1560–1568.

Liu, J., P. Musialski, P. Wonka, and J. Ye, 2009, Tensor completion for estimating missing values in visual data: IEEE 12th International Conference on Computer Vision, 2114 – 2121.

Malcolm, A. E., M. V. de Hoop, and J. H. LeRousseau, 2005, The applicability of dip moveout/azimuth moveout in the presence of caustics: Geophysics, **70**, no. 1, S1–S17.

Mayne, W., 1962, Common reflection point horizontal data stacking techniques: Geophysics, **27**, no. 6, 927–938.

McLachlan, G. J., and T. Krishnan, 1997, The EM algorithm and extensions: John Wiley and Sons, New York, NY, USA. Wiley series in probability and statistics. Applied probability and statistics.

M'Kendrick, A. G., 1925, Applications of mathematics to medical problems: Proceedings of the Edinburgh Mathematical Society, **44**, 98–130.

Morice, S., S. Ronen, P. Canter, K. Welker, and D. Clark, 2000, The impact of positioning differences on 4D repeatability: SEG Technical Program Expanded Abstracts, 1611–1614.

Naghizadeh, M., 2012, Seismic data interpolation and denoising in the frequency-wavenumber domain: Geophysics, **77**, V71–V80.

Naghizadeh, M., and M. Sacchi, 2013, Multidimensional de-aliased Cadzow reconstruction of seismic records: Geophysics, **78**, no. 1, A1–A5.

Naghizadeh, M., and M. D. Sacchi, 2007, Multistep autoregressive reconstruction of seismic records: Geophysics, **72**, no. 6, V111–V118.

———, 2009, *F-X* adaptive seismic-trace interpolation: Geophysics, **74**, no. 1, V9–V16.

———, 2010a, Beyond alias hierarchical scale curvelet interpolation of regularly and irregularly sampled seismic data: Geophysics, **75**, no. 6, WB189–WB202.

———, 2010b, On sampling functions and Fourier reconstruction methods: Geophysics, **76**, WB137 – WB151.

———, 2010c, Seismic data reconstruction using multidimensional prediction filters: Geophysical Prospecting, **58**, no. 2, 157–173.

Narita, A., K. Hayashi, R. Tomioka, and H. Kashima, 2011, Tensor factorization using auxiliary information, *in* Machine Learning and Knowledge Discovery in Databases: Springer Berlin Heidelberg, volume **6912** *of* Lecture Notes in Computer Science, 501–516.

Navasca, C., L. De Lathauwer, and S. Kindermann, 2008, Swamp reducing technique for tensor decomposition: Presented at the Proc. 16th European Signal Processing Conference (EUSIPCO 2008).

Nelson, P. R., P. A. Taylor, and J. F. MacGregor, 1996, Missing data methods in PCA and PLS: Score calculations with incomplete observations: Chemometrics and Intelligent Laboratory Systems, **35**, 45 – 65.

Nemeth, T., C. Wu, and G. T. Schuster, 1999, Least-squares migration of incomplete reflection data: Geophysics, **64**, 208–221.

Nocedal, J., and S. J. Wright, 2006, Numerical optimization, 2nd ed.: Springer.

Omberg, L., G. H. Golub, and O. Alter, 2007, A tensor higer-order singular value decomposition for integrative analysis of DNS microarray data from different studies: PNAS, **104**, no. 47, 18371–18376.

Oropeza, V., 2010, The Singular Spectrum Analysis method and its application to seismic data denoising and reconstruction: MSc Thesis, Department of Physics, University of Alberta, Edmonton, Canada.

Oropeza, V., and M. Sacchi, 2011, Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis: Geophysics, **76**, no. 3, V25–V32.

Oropeza, V. E., and M. D. Sacchi, 2009, Multifrequency Singular Spectrum Analysis: SEG Expanded Abstracts, **28**, 3193–3197.

———, 2010, A randomized SVD for Multichannel Singular Spectrum Analysis (MSSA) noise attenuation: SEG Expanded Abstracts, **29**, 3539–3544.

Porsani, M., 1999, Seismic trace interpolation using half-step prediction filters: Geophysics,

**64**, no. 5, 1461–1467.

Powell, M. J. D., 1969, A method for nonlinear constraints in minimization problems *in* Optimization ed. by R. Fletcher: Academic Press.

Rajih, M., and P. Comon, 2005, Enhanced line search: A novel method to accelerate Parafac: in Eusipco05, 4–8.

Reiter, E., and C. Johnson, 2012, Limits of computation: An introduction to the undecidable and the intractable: Taylor & Francis Group.

Ronen, J., 1987, Wave-equation trace interpolation: Geophysics, **52**, no. 7, 973–984.

Sacchi, M., 2009, *FX* Singular Spectrum Analysis: CSPG CSEG CWLS Convention, 392–395.

Sacchi, M. D., M. M. I. Bhuiyan, and N. Kazemi, 2013, Maximum entropy acquisition design and optimal interpolation: CSEG Geoconvention Expanded Abstracts, 1–6.

Sacchi, M. D., and B. Liu, 2005, Minimum weighted norm wavefield reconstruction for AVA imaging: Geophysical Prospecting, **53**, no. 6, 787–801.

Sacchi, M. D., T. J. Ulrych, and C. J. Walker, 1998, Interpolation and Extrapolation Using a High-Resolution Discrete Fourier Transform: IEEE Transaction on Signal Processing, **46**, no. 1, 31–38.

Schonewille, M., 2003, A modeling study on seismic data regularization for timelapse applications: SEG Technical Program Expanded Abstracts, 1537–1540.

Sheriff, R. E., 2002, Encyclopedic dictionary of applied geophysics, fourth ed.: Society of Exploration Geophysicists.

Sidiropoulos, N. D., and R. Bro, 2000, On the uniqueness of multilinear decomposition of N-way arrays: Journal of Chemometrics, **14**, 229–239.

Signoretto, M., L. De Lathauwer, and J. A. K. Suykens, 2010, Nuclear norms for tensors and their use for convex multilinear estimation: Technical report, ESAT-SISTA, K.U.Leuven (Leuven, Belgium).

Slotnick, M. M., 1959, Lessons in seismic computing: Society of Exploration Geophysicists (Edited by R. A. Geyer).

Spitz, S., 1991, Seismic trace interpolation in the *F-X* domain: Geophysics, **56**, no. 6, 785–794.

Stanton, A., N. Kreimer, D. Bonar, M. Naghizadeh, and M. Sacchi, 2012, A comparison of 5D reconstruction methods: SEG Technical Program Expanded Abstracts, 1–5.

Stanton, A., and M. Sacchi, 2013a, Vector reconstruction of multicomponent seismic data: Geophysics, **78**, V131–V145.

Stanton, A., and M. D. Sacchi, 2013b, Shear wave splitting parameter estimation using a regular distribution of azimuths: CSEG GeoConvention Expanded Abstracts, 1–4.

Stolt, R. H., 2002, Seismic data mapping and reconstruction: Geophysics, **67**, no. 3, 890–908.

Strang, G., 2006, Linear algebra and its applications: Thomson, Brooks/Cole.

Tanner, J., and K. Wei, 2012, Normalized iterative hard thresholding for matrix completion: preprint, 1–23.

Tomasi, G., 2006, Practical and computational aspects in chemometric data analysis: Ph. D. dissertation: Department of Food Science, Royal Veterinary and Agricultural University.

Tomasi, G., and R. Bro, 2005, PARAFAC and missing values: Chemometrics and Intelligent Laboratory Systems, **75**, no. 2, 163 – 180.

———, 2006, A comparison of algorithms for fitting the PARAFAC model: Computational Statistics & Data Analysis, **50**, 1700 – 1734.

Tomioka, R., K. Hayashi, and H. Kashima, 2011, Estimation of low-rank tensors via convex optimization: Arxiv Preprint arXiv:1010.0789v2 [stat.ML].

Tourlakis, G., 2012, Theory of computation: Wiley.

Trad, D., 2008, Five dimensional seismic data interpolation: 78th Annual International Meeting, SEG, Expanded Abstracts, 978–981.

———, 2009, Five-dimensional interpolation: Recovering from acquisition constraints: Geophysics, **74**, no. 6, V123–V132.

Trad, D., T. J. Ulrych, and M. D. Sacchi, 2002, Accurate interpolation with high-resolution time-variant Radon transforms: Geophysics, **67**, no. 2, 644–656.

Trickett, S., 2008, *F-XY* Cadzow noise suppression: SEG, Expanded Abstracts, **27**, no. 1, 2586–2590.

Trickett, S., and L. Burroughs, 2009, Prestack rank-reducing noise suppression: Theory: SEG, Expanded Abstracts, **28**, no. 1, 3332–3336.

———, 2013, Interpolation using Hankel tensor completion: CSEG GeoConvention Expanded Abstracts, 1–4.

Trickett, S., L. Burroughs, A. Milton, L. Walton, and R. Dack, 2010, Rank-reduction-based trace interpolation: SEG, Expanded Abstracts, **29**, no. 1, 3829–3833.

Trickett, S. R., 2003, *F-XY* eigenimage noise suppression: Geophysics, **68**, no. 2, 751–759.

Tseng, P., 1991, Applications of a splitting algorithm to decomposition in convex programming and variational inequalities: SIAM Journal on Control and Optimization, **29**, no. 1, 119–138.

Tucker, L., 1966, Some mathematical notes on three-mode factor analysis: Psychometrika, **31**, no. 3, 279–311.

Tucker, L. R., 1963, Implications of factor analysis of three-way matrices for measurement of change: In C. W. Harris (Ed.), Problems in measuring change, 122–137.

———, 1964, The extension of factor analysis to three-dimensional matrices, *in* Contributions to mathematical psychology.: Holt, Rinehart and Winston, 110–127.

Van Borselen, R., M. Schonewille, and R. Hegge, 2004, 3D SRME: Acquisition & processing solutions: SEG Technical Program Expanded Abstracts, 1241–1244.

Van Dedem, E., and D. Verschuur, 2005, 3D surface-related multiple prediction: A sparse inversion approach: Geophysics, **70**, V31–V43.

Van Dedem, E. J., and D. J. Verschuur, 1998, 3D surface-related multiple elimination and interpolation: SEG Technical Program Expanded Abstracts, 1321–1324.

Verschuur, D., A. Berkhout, and C. Wapenaar, 1992, Adaptive surface-related multiple elimination: Geophysics, **57**, 1166–1177.

Vrabie, V., N. Le Bihan, and J. Mars, 2006, Multicomponent wave separation using HOSVD/unimodal-ICA subspace method: Geophysics, **71**, no. 5, V133–V143.

Walczak, B., and D. Massart, 2001, Dealing with missing data: Part I: Chemometrics and Intelligent Laboratory Systems, **58**, 15 – 27.

Xu, S., Y. Zhang, D. Pham, and G. Lambare, 2005, Antileakage Fourier transform for seismic data regularization: Geophysics, **70**, no. 4, V87–V95.

Yilmaz, Ö., 2001, Seismic data analysis: Society of Exploration Geophysicists.

Zwartjes, P., and A. Gisolf, 2007, Fourier reconstruction with sparse inversion: Geophysical Prospecting, **55**, no. 2, 199–221.

Zwartjes, P., and M. D. Sacchi, 2007, Fourier reconstruction of nonuniformly sampled, aliased seismic data: Geophysics, **72**, no. 1, V21–V32.

Zwartjes, P. M., and C. O. H. Hindriks, 2001, Regularising 3D data using Fourier reconstruction and sparse inversion: SEG Technical Program Expanded Abstracts, **20**, no. 1, 1906–1909.

# APPENDIX A

# Rank of a tensor composed of separable events

## A.1 Model containing one linear event

In the $F-X$ domain, a seismic volume with three spatial coordinates can be represented by a third-order tensor. A seismic volume with one linear event can be modeled as

$$s(t,x,y,z) = w(t-px-qy-rz), \tag{A.1}$$

where $p,q,r$ are the ray parameters in $x,y,z$ respectively. Taking the Fourier transform with respect to time and using the shift theorem, equation A.1 becomes a plane-wave with the following expression

$$S(\omega,x,y,z) = W(\omega)\exp^{-j\omega px}\exp^{-j\omega qy}\exp^{-j\omega rz}, \tag{A.2}$$

where $W(\omega)$ is the frequency dependent amplitude of the reflection, $j^2=-1$ is the imaginary unit and $\omega$ is the frequency. Assuming regularly sampled data $x=i_1\Delta x, y=i_2\Delta y$ and $z=i_3\Delta z$. To make our notation more compact, we define $\alpha=p\Delta x, \beta=q\Delta y$ and $\gamma=r\Delta z$. Therefore, the elements of the cube in $F-X$ are

$$S_{i_1i_2i_3} = W(\omega)\exp^{-j\omega\alpha i_1}\exp^{-j\omega\alpha i_2}\exp^{-j\omega\beta i_3}, \tag{A.3}$$

As a simplification, this model can be rewritten as

$$S_{i_1i_2i_3} = a_{i_1}\circ b_{i_2}\circ c_{i_3}, \tag{A.4}$$

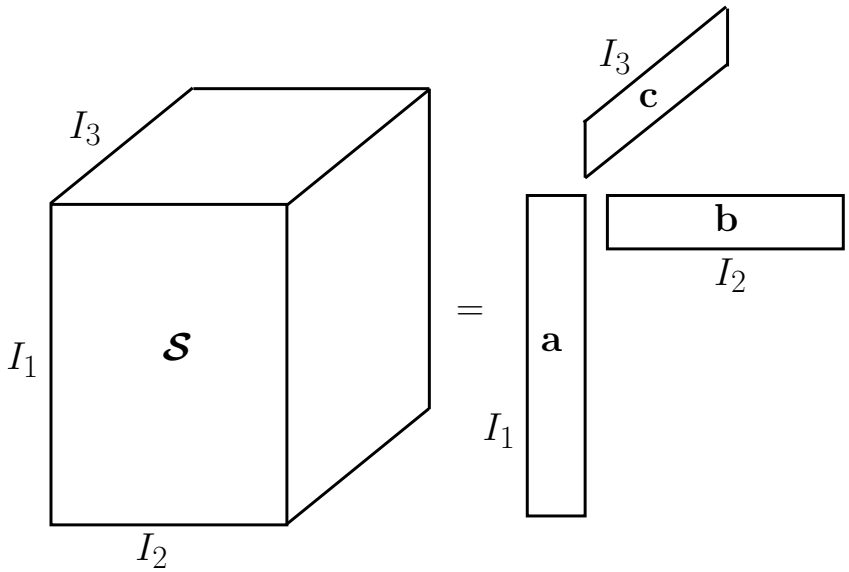



Figure A.1: Model of seismic volume with 3 spatial coordinates and 1 linear event, for a fixed frequency.

where we have adopted $W(\omega) = 1$ for simplicity. The vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$ represent the vectors $\exp^{-i\omega\alpha i_1}, i_1 = 1 \cdots I_1$, $\exp^{-i\omega\alpha i_2}, i_2 = 1 \cdots I_2$ and $\exp^{-i\omega\alpha i_3}, i_3 = 1 \cdots I_3$, respectively. Figure A.1 shows a sketch of this model for one frequency.

Equation A.4 is equivalent to a CP decomposition with one term (Carroll and Chang, 1970; Harshman, 1970) (refer to Chapter 2). This expression can be rewritten in matricial form for each of the unfoldings of $\boldsymbol{\mathcal{S}}$ (Kolda and Bader, 2009):

$$\begin{aligned} \mathbf{S}^{(1)} &= \mathbf{a}(\mathbf{c} \odot \mathbf{b})^{\mathrm{T}} \\ \mathbf{S}^{(2)} &= \mathbf{b}(\mathbf{c} \odot \mathbf{a})^{\mathrm{T}} \\ \mathbf{S}^{(3)} &= \mathbf{c}(\mathbf{b} \odot \mathbf{a})^{\mathrm{T}}. \end{aligned} \tag{A.5}$$

We intend to prove that the unfoldings of tensor $\boldsymbol{\mathcal{S}}$ are low-rank. For this purpose, we will use the properties specified in Chapter 2. Let us focus on the unfolding in mode-1 $\mathbf{S}^{(1)}$. Notice that because $\mathbf{a}, \mathbf{b}, \mathbf{c}$ are vectors, we can replace the Khatri-Rao product by the Kronecker product. Therefore

$$\mathbf{S}^{(1)} \;=\; \mathbf{a}(\mathbf{c} \odot \mathbf{b})^{\mathrm{T}} \;=\; \mathbf{a}(\mathbf{c} \otimes \mathbf{b})^{\mathrm{T}}. \tag{A.6}$$

Next, we estimate the rank of the preceding expression. Following equation 2.9 and equation

2.10,

$$\text{rank}((\mathbf{c} \otimes \mathbf{b})^{\text{T}}) \;=\; \text{rank}(\mathbf{c} \otimes \mathbf{b}) \;=\; \text{rank}(\mathbf{c})\,\text{rank}(\mathbf{b}). \tag{A.7}$$

Finally,

$$\text{rank}(\mathbf{S}^{(1)}) \;=\; \mathbf{a}(\mathbf{c} \otimes \mathbf{b})^{\text{T}} \;\leq\; \min(\text{rank}(\mathbf{a}), \text{rank}(\mathbf{c})\,\text{rank}(\mathbf{b})), \tag{A.8}$$

using the property 2.11. Because $\mathbf{a}, \mathbf{b}, \mathbf{c}$ are vectors, their ranks are equal to one. Therefore,

$$\begin{aligned} \text{rank}(\mathbf{S}^{(1)}) &\leq \min(\text{rank}(\mathbf{a}), \text{rank}(\mathbf{c})\,\text{rank}(\mathbf{b})) \\ &= \min(1, 1 \;\centerdot\; 1) = 1. \end{aligned} \tag{A.9}$$

The rank of the unfolding in mode-1 of $\boldsymbol{\mathcal{S}}$ cannot be less than one. In consequence, its rank is one. For the other two unfoldings, the proof is analogous with a change of order of $\mathbf{a}, \mathbf{b}, \mathbf{c}$. In conclusion, for a 3D model with one event, the tensor $\boldsymbol{\mathcal{S}}$ is low-rank. More specifically, $\boldsymbol{\mathcal{S}}$ has $n$-ranks equal to one when containing one linear event.

For four spatial coordinates, the proof is very similar. For this case, we have another vector added to the outer product, e.g. a vector $\mathbf{d}$. The unfolding in mode-1 of tensor $\boldsymbol{\mathcal{S}}$ will be

$$\mathbf{S}^{(1)} = \mathbf{a}(\mathbf{d} \odot \mathbf{c} \odot \mathbf{b})^{\text{T}}.$$

Because again $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$ are vectors, the Khatri-Rao product can be replaced by the Kronecker product. The properties that we used in the proof are all applicable to the four-dimensional (4D) case. For this reason, we reach the same conclusion as for the 3D case that the tensor $\boldsymbol{\mathcal{S}}$ is low-rank (and its $n$-ranks are equal to one).

### A.1.1 Extension to separable events

Let us assume now that our model consists of one event with variable dip. In addition, we will assume that the seismic volume can be modeled as

$$s(t, x, y, z) = w(t - \Phi_1(x) - \Phi_2(y) - \Phi_3(z)). \tag{A.10}$$

Again, expression A.10 can be written as follows

$$S(\omega, x, y, z) = W(\omega) \exp^{-j\,\Phi_1(x)} \exp^{-j\,\Phi_2(y)} \exp^{-j\,\Phi_3(z)}, \tag{A.11}$$

where the functions $\Phi_i$ can have either the form for linear events $j\omega px$, for parabolic events $j\omega px^2$, or other form. Notice that all the derivations from the previous section are still valid

for model A.11. The important fact is that $s(t, x, y, z)$ can be decomposed as a product of separable components of $x, y, z$. As a result, the tensor $\boldsymbol{\mathcal{S}}$ is low-rank also in the presence of curved events, if the event travel-time is separable in the coordinates $x, y$ and $z$.

## A.2 Model containing two linear events

Following the model proposed in the previous section, a seismic volume with two linear events and three spatial coordinates can be modeled as

$$s(t, x, y, z) = w(t - p_1 x - q_1 y - r_1 z) + w(t - p_2 x - q_2 y - r_2 z), \tag{A.12}$$

where $p_i, q_i, r_i, i = 1, 2$ are the ray parameters in $x, y, z$ respectively for events 1 and 2, respectively. In the $F - X$ domain, the model becomes

$$\begin{aligned} S(\omega, x, y, z) = W(\omega) \exp^{-j\omega p_1 x} \exp^{-j\omega q_1 y} \exp^{-j\omega r_1 z} \\ + W(\omega) \exp^{-j\omega p_2 x} \exp^{-j\omega q_2 y} \exp^{-j\omega r_2 z}, \end{aligned} \tag{A.13}$$

where $W(\omega)$ is the frequency dependent amplitude of the reflections, $j^2 = -1$ is the imaginary unit and $\omega$ is the frequency. Assuming the same compact notation as in the previous section, the elements of the cube in $F - X$ are

$$S_{i_1 i_2 i_3} = a_{1 i_1} \circ b_{1 i_2} \circ c_{1 i_3} + a_{2 i_1} \circ b_{2 i_2} \circ c_{2 i_3}, \tag{A.14}$$

where we have adopted $W(\omega) = 1$ for simplicity. Figure A.2 shows a sketch of this model for one frequency.

Like in the previous section, equation A.14 resembles the CP decomposition with two terms. To make the analysis more simple, we can form matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ with column vectors $\mathbf{a_1}, \mathbf{a_2}, \mathbf{b_1}, \mathbf{b_2}$ and $\mathbf{c_1}, \mathbf{c_2}$ respectively, as explained in Chapter 2, section 2.4.1. The unfoldings of $\boldsymbol{\mathcal{S}}$ are (Kolda and Bader, 2009):

$$\begin{aligned} \mathbf{S}^{(1)} &= \mathbf{A}(\mathbf{C} \odot \mathbf{B})^{\mathrm{T}} \\ \mathbf{S}^{(2)} &= \mathbf{B}(\mathbf{C} \odot \mathbf{A})^{\mathrm{T}} \\ \mathbf{S}^{(3)} &= \mathbf{C}(\mathbf{B} \odot \mathbf{A})^{\mathrm{T}}. \end{aligned} \tag{A.15}$$

As with the case containing one event, we want to prove that the tensor $\boldsymbol{\mathcal{S}}$ is low-rank.

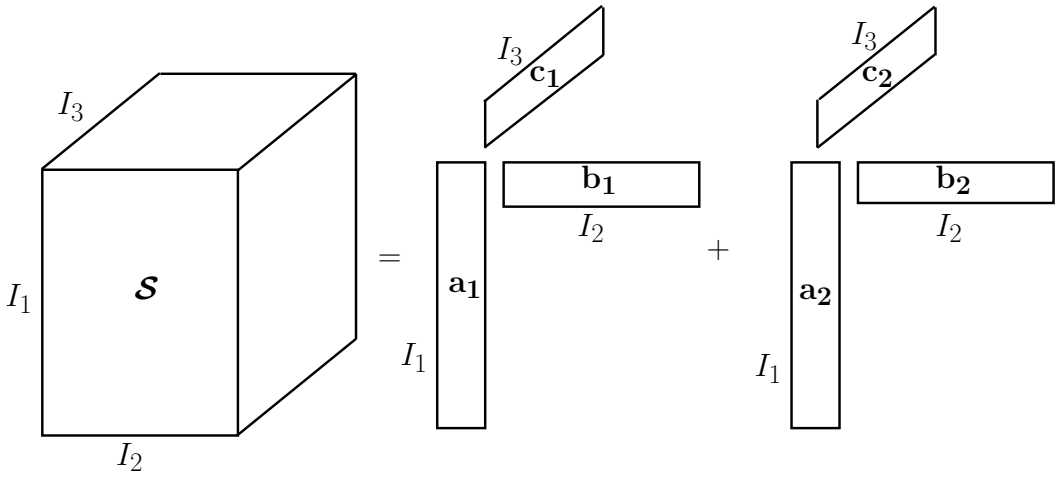



Figure A.2: Model of seismic volume with three spatial coordinates and one linear event, for a fixed frequency.

Focusing in the unfolding in mode-1 and using property 2.12, the following applies

$$\begin{aligned}\mathbf{S}^{(1)} &= \mathbf{A}(\mathbf{C} \odot \mathbf{B})^{\mathrm{T}} \\ &= \mathbf{A}([\mathbf{c_1} \otimes \mathbf{b_1} \quad \mathbf{c_2} \otimes \mathbf{b_2}])^{\mathrm{T}}. \end{aligned} \tag{A.16}$$

The Kronecker product $\mathbf{c_1} \otimes \mathbf{b_1}$ is a vector, from property 2.7. Therefore, it has rank one. The same applies to the product $\mathbf{c_2} \otimes \mathbf{b_2}$. The components $\mathbf{a_i}, \mathbf{b_i}$ and $\mathbf{c_i}$ represent coordinates of the event $i$. If we assume that both events are not related, i.e. they contain different dips, then the product $\mathbf{c_1} \otimes \mathbf{b_1}$ is linearly independent from $\mathbf{c_2} \otimes \mathbf{b_2}$. The rank of the matrix $[\mathbf{c_1} \otimes \mathbf{b_1} \quad \mathbf{c_2} \otimes \mathbf{b_2}]^{\mathrm{T}}$ with this assumption is two (or one if the dips are not different in each coordinate). In the same manner, the matrix $\mathbf{A}$ is rank two because it contains vectors representing one coordinate for two different events. Using property 2.11, the rank of the unfolding $\mathbf{S}^{(1)}$ becomes

$$\begin{aligned}\mathrm{rank}(\mathbf{S}^{(1)}) &\leq \min(\mathrm{rank}(\mathbf{A}), \mathrm{rank}([\mathbf{c_1} \otimes \mathbf{b_1} \quad \mathbf{c_2} \otimes \mathbf{b_2}])) \\ &= \min(2, 2) = 2. \end{aligned} \tag{A.17}$$

In conclusion, we have proved that the unfolding in mode-1 of tensor $\boldsymbol{\mathcal{S}}$ is low-rank for three spatial coordinates and two events. The proof for the other two unfoldings is irrelevant, as it only implies a change in the ordering of the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$.

The expansion to four spatial coordinates is, like in the preceding section, trivial. Instead of having three vectors in the model representation, we will have four. The product $(\mathbf{C} \odot \mathbf{B})$ becomes $(\mathbf{D} \odot \mathbf{C} \odot \mathbf{B})$ instead. This Khatri-Rao product of matrices is equivalent to the

matrix $[\mathbf{d_1} \otimes \mathbf{c_1} \otimes \mathbf{b_1} \quad \mathbf{d_2} \otimes \mathbf{c_2} \otimes \mathbf{b_2}]$. The same reasoning as the one made before applies for this case as well, such that the tensor $\boldsymbol{\mathcal{S}}$ for the 4D case is also low-rank. As before, we also proved that the $n$-ranks of $\boldsymbol{\mathcal{S}}$ are less or equal to two for this case.

### A.2.1 Extension to separable events

As we did in section A.1.1, the seismic volume can be modeled as

$$s(t,x,y,z) = w(t - \Phi_1(x) - \Phi_2(y) - \Phi_3(z)) + w(t - \Phi_4(x) - \Phi_5(y) - \Phi_6(z)), \tag{A.18}$$

which in $F - X$ is equivalent to

$$\begin{aligned} S(\omega,x,y,z) = W(\omega) \exp^{-j\,\Phi_1(x)} \exp^{-j\,\Phi_2(y)} \exp^{-j\,\Phi_3(z)} \\ + W(\omega) \exp^{-j\,\Phi_4(x)} \exp^{-j\,\Phi_5(y)} \exp^{-j\,\Phi_6(z)}, \end{aligned} \tag{A.19}$$

where $\Phi_i$ can be the expression for a linear event, a parabolic event or other expression. The derivations from the previous section are still valid with model A.19. Therefore, the tensor $\boldsymbol{\mathcal{S}}$ is low-rank in this case as well, as long as it can be decomposed into a sum of two outer products of separable components in $x, y$ and $z$.

## A.3 Model containing $n$ linear events

If the seismic volume contains $n$ events that are not related, i.e. contain distinct dips, similar arguments as those presented above will apply. The number $n$ has to be small enough compared to the sizes of the matrix unfoldings such that the matrices can be considered to be low-rank. In the case of two events, we observed that the model leads to a sum of two outer products of three vectors (or four for the 4D case). For $n$ events, we will have a sum of $n$ outer products of three (or four) vectors. The matrices that we used for compacting the notation will therefore have $n$ columns instead. The relationship $\mathbf{C} \odot \mathbf{B}$ will be equal to $[\mathbf{c_1} \otimes \mathbf{b_1} \quad \mathbf{c_2} \otimes \mathbf{b_2} \cdots \mathbf{c_n} \otimes \mathbf{b_n}]$ for the 3D case. It is not difficult to observe that the same properties as those used for the case of two events can be used for the case of $n$ events. Therefore, the $n$-ranks of $\boldsymbol{\mathcal{S}}$ for the case of $n$ linear events will be less or equal than $n$. Furthermore, following A.1.1 and A.2.1, $n - \text{rank}(\boldsymbol{\mathcal{S}}) \leq n$ when the model can be decomposed as a sum of $n$ outer product of separable components, where these components can represent, for example, linear or parabolic events.

## A.4 Validity of the separability assumption

Throughout this appendix, we have assumed that the waveforms can be approximated by a product of separate functions of the spatial coordinates. In other words, we have assumed that the following model

$$s(t, x, y, z) = w(t - \Phi(x, y, z)), \tag{A.20}$$

which in the $F - X$ domain is represented as

$$S(\omega, x, y, z) = W(\omega) \exp^{-j\omega\, \Phi(x,y,z)}, \tag{A.21}$$

is such that $\Phi(x, y, z) = \phi_1(x) + \phi_2(y) + \phi_3(z)$. With this assumption, the exponential in equation A.21 is separated into the product of three exponentials and the developments from sections A.1 and A.2 follow. However, this assumption is not valid in some situations and we will analyze this using a general travel-time expression for a dipping plane in 3D.

We will utilize the formulation in the article by Fomel (2011). The 3D dipping reflector model in Figure A.3 will be used in the following analysis (after the article by Levin (1971)). The general equation for the plane is

$$x \cos\alpha + y \cos\beta + z \cos\gamma = d, \tag{A.22}$$

where $\alpha, \beta$ and $\gamma$ are the normal angles between the $x, y$ and $z$ axes, respectively, and the normal to the dipping plane ($\gamma$ is the dip angle of the reflector as well). These angles also satisfy $\cos^2\alpha + \cos^2\beta + cos^2\gamma = 1$. The travel-time equation for the geometry of Figure A.3, where the source and receiver are located on a horizontal plane, is given by (Slotnick, 1959; Levin, 1963)

$$t(m_x, m_y, h_x, h_y) = \frac{2}{v}\sqrt{D^2 + h_x^2 + h_y^2 - (h_x \cos\alpha + h_y \cos\beta)^2}, \quad \text{where} \tag{A.23}$$

$$D = d - m_x \cos\alpha - m_y \cos\beta. \tag{A.24}$$

$D$ is the length of the normal to the reflector in the midpoint location, whereas $d$ is the constant in the plane's equation (equation A.22). The variables $m_x, m_y, h_x$ and $h_y$ are the inline, crossline, inline offset and crossline offset, respectively. The constant $v$ is the local propagation velocity. For the particular case where the source is located at $(0, 0, 0)$, $d$ is also the length of the normal to the reflector in the source location. Figure A.3 assumes the source is located at $(0, 0, 0)$, but the former equation applies also to the more generic case where the source is positioned at an arbitrary location.

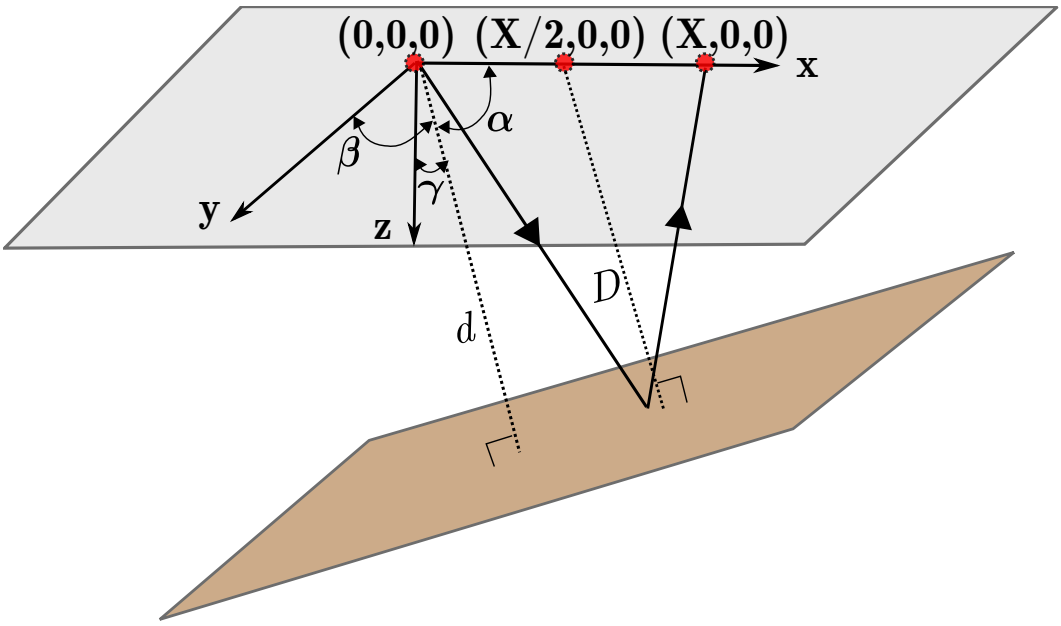



Figure A.3: 3D dipping reflector model, after Levin (1971).

The travel-time equation A.24 is clearly not separable as a sum of functions dependent of $m_x, m_y, h_x$ and $h_y$. However, this formula can be approximated by a second order Taylor series. The second order Taylor expansion at the point $(0,0,0,0) = \mathbf{0}$ is given by:

$$t(m_x,m_y,h_x,h_y) \approx t(\mathbf{0}) + m_x\frac{\partial t}{\partial m_x}(\mathbf{0}) + m_y\frac{\partial t}{\partial m_y}(\mathbf{0}) + h_x\frac{\partial t}{\partial h_x}(\mathbf{0}) + h_y\frac{\partial t}{\partial h_y}(\mathbf{0}) +$$
$$+\frac{1}{2}\left[m_x^2\frac{\partial^2 t}{\partial m_x^2}(\mathbf{0}) + 2m_xm_y\frac{\partial^2 t}{\partial m_x\partial m_y}(\mathbf{0}) + m_y^2\frac{\partial^2 t}{\partial m_y^2}(\mathbf{0}) + 2m_xh_x\frac{\partial^2 t}{\partial m_x\partial h_x}(\mathbf{0}) + h_x^2\frac{\partial^2 t}{\partial h_x^2}(\mathbf{0}) +\right.$$
$$\left.+2m_xh_y\frac{\partial^2 t}{\partial m_x\partial h_y}(\mathbf{0}) + h_y^2\frac{\partial^2 t}{\partial h_y^2}(\mathbf{0}) + 2m_yh_x\frac{\partial^2 t}{\partial m_y\partial h_x}(\mathbf{0}) + 2m_yh_y\frac{\partial^2 t}{\partial m_y\partial h_y}(\mathbf{0})\right]. \qquad \text{(A.25)}$$

Replacing all derivatives evaluated at $\mathbf{0}$ in expression A.25, the second order Taylor approximation becomes

$$t(m_x,m_y,h_x,h_y) \approx \frac{2d}{v} - \frac{2}{v}m_x\cos\alpha - \frac{2}{v}m_y\cos\beta + \frac{1}{vd}h_x^2\sin^2\alpha +$$
$$+\frac{1}{vd}h_y^2\sin^2\beta - \frac{2}{vd}h_xh_y\cos\alpha\cos\beta \qquad \text{(A.26)}$$

This equation is also not separable with respect to $h_x, h_y$. However, let us analyze two particular cases:

- If $\alpha = 90^\circ$ and $\beta = 90^\circ$ (flat reflector case):

$$t(m_x, m_y, h_x, h_y) \approx \frac{2d}{v} + \frac{1}{vd}\, h_x^2 + \frac{1}{vd}\, h_y^2, \tag{A.27}$$

  which is separable with respect to $h_x, h_y$ and is the expression of a paraboloid.

- If $\beta = 90^\circ$ (source-receiver direction aligned to the dip direction):

$$t(m_x, m_y, h_x, h_y) \approx \frac{2d}{v} - \frac{2}{v}\, m_x\, \cos\alpha + \frac{1}{vd}\, h_x^2\, \sin^2\alpha + \frac{1}{vd}\, h_y^2, \tag{A.28}$$

  which is also separable with respect to $h_x, h_y$ and is the expression of a paraboloid.

- If $\alpha = 90^\circ$ (source-receiver direction perpendicular to the dip direction):

$$t(m_x, m_y, h_x, h_y) \approx \frac{2d}{v} - \frac{2}{v}\, m_y\, \cos\beta + \frac{1}{vd}\, h_x^2 + \frac{1}{vd}\, h_y^2\, \sin^2\beta, \tag{A.29}$$

  which is also separable with respect to $h_x, h_y$ and is the expression of a paraboloid.

In conclusion, the second-order Taylor approximation to the travel-time equation A.26 is only separable in the particular cases in which the angles $\alpha = 90^\circ$, $\beta = 90^\circ$ and when both angles are equal to $90^\circ$. These are the cases in which the source-receiver direction is perpendicular to the dip direction, the case in which the source-receiver direction is aligned to the dip direction and the case of a flat reflector, respectively. Any other combination of $\alpha$ and $\beta$ will violate the separability assumption. However, the examples presented in Chapter 5 are generated using $\alpha$ and $\beta$ angles which are not equal to $90^\circ$ and the reconstruction is successful. More research is needed to understand the impact of violating the separability assumption.