

IDEA League

MASTER OF SCIENCE IN APPLIED GEOPHYSICS
RESEARCH THESIS

Data-driven 3D Deghosting Using Multisensor Marine Measurements

Erik Franciscus Maria Koene

August 12, 2016

Data-driven 3D Deghosting Using Multisensor Marine Measurements

MASTER OF SCIENCE THESIS

for the degree of Master of Science in Applied Geophysics at
Delft University of Technology
ETH Zürich
RWTH Aachen University
by

Erik Franciscus Maria Koene

August 12, 2016

Department of Geoscience & Engineering	.	Delft University of Technology
Department of Earth Sciences	.	ETH Zürich
Faculty of Georesources and Material Engineering	.	RWTH Aachen University

Copyright © 2016 by IDEA League Joint Master's in Applied Geophysics:

ETH Zürich

All rights reserved.

No part of the material protected by this copyright notice may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying or by any information storage and retrieval system, without permission from this publisher.

Printed in Switzerland

IDEA LEAGUE
JOINT MASTER'S IN APPLIED GEOPHYSICS

Delft University of Technology, The Netherlands
ETH Zürich, Switzerland
RWTH Aachen, Germany

Dated: *August 12, 2016*

Supervisor(s):

Dr. Yousif Izzeldin Kamil (Schlumberger)

Dr. Dirk-Jan van Manen (ETH Zürich)

Committee Members:

Dr. Yousif Izzeldin Kamil (Schlumberger)

Dr. Dirk-Jan van Manen (ETH Zürich)

Dr. Ranajit Ghose (TU Delft)

Abstract

Marine seismic acquisition tows submerged streamers to record pressure waves from the subsurface. The recording, however, contains both the desired upgoing wavefield and its (immediate) reflection off the sea-surface, causing a downgoing wavefield known as the seismic ghost. Interference between the up-and downgoing waves causes periodic low signal to noise ratio (S/N) ‘ghost notches’ in the recorded spectrum. To restore the broadband upgoing signal, we must remove the ghost (‘*deghosting*’). Deghosting using solely pressure measurements fails to restore the signal in the low S/N notches of the data. Current acquisition techniques acquire signals with different ghost notches, such that their proper combination recovers the broadband signal. This thesis uses multisensor acquisition: measurements of the pressure and particle velocity vector. The ghost notches on the pressure and vertical particle velocity are offset by half a period, such that their combination may provide good S/N at all frequencies. Current multisensor deghosting techniques make deterministic assumptions on the data and ghost model (such as a known streamer depth, or assuming energy propagating only along the streamer). If the assumptions do not correspond to the data, the deghosting fails to restore the true broadband signal. We propose two novel data-driven deghosting techniques, which estimate an adequate deghosting filter based on the data itself. The first method estimates the 3D propagation of energy using measurements of the pressure and crossline particle velocity along a single streamer. The 3D incidence angle is used to sum the pressure wave with vertical particle velocity such that only the upgoing wave is recovered. The second method estimates the filter parameters that explain the recorded ghosted data by minimizing a multisensor least-squares deghosting cost function. The cost function is analytically shown to outperform similar single sensor adaptive deghosting techniques in terms of sensitivity to the true ghost model. The obtained filter parameters may then be used to construct an inverse filter that restores the upgoing wavefield. We found that both methods produce encouraging results on real data, outperforming the existing deterministic multisensor deghosting methods.

Acknowledgements

This thesis project was the first in my life where each answered question spurred a hundred more. I cannot thank my supervisor Yousif Izzeldin Kamil enough for having the patience to remind me (again and again) to focus on the important ones. This thesis would not exist without you and your input, in such a way that I cannot help but think of this thesis as anything else than a team effort. Thanks for the laughs, conversations, and making me a better geophysicist – even though you don't consider yourself wise enough to be called one. I think you're wrong about that!

The team work extends further out to Philippe Caprioli and Massimiliano Vassallo at WesternGeco and Dirk-Jan van Manen at the ETH Zürich, who also supervised me. Your input was always much appreciated, and more than once you managed to make immediate sense of results I worked hard on but didn't understand at all. I hope you will keep on taking interns, even in these difficult industry conditions.

Further thanks to Pradeep Loganathan for truly getting me up and running in the office, and the good times afterwards. This extends to all the people at WesternGeco - LTC, thanks for making it a good time.

I must thank my fellow interns, classmates and London housemates Fabian and Julian. We joked upfront we would surely hate each other at the end of this process, being around each other 24/7. This could hardly be further away from reality, I had great fun living with you guys (and Gabriela, Julian's girlfriend). I will miss it.

Lastly, I want to extend my greatest gratitude to my family. The constant traveling for this Master's has not always been easy on you and me. I couldn't be happier and more proud that you have always been there for me. Thanks for letting me follow my passions.

Table of Contents

Abstract	v
Acknowledgements	vii
Background	xvii
Conventions	xvii
Plane wave decomposition	xx
1 The ghost problem	1
1-1 Broadband seismics & the receiver ghost	1
1-2 The ghost as filter	2
1-2-1 The ghost as frequency filter	2
1-2-2 The ghost as frequency-wavenumber and frequency-slowness filter	3
1-3 Deterministic deghosting	5
1-3-1 Inverse filter	5
1-3-2 Drawbacks of deterministic deghosting	6
1-4 Acquiring more ghosts	7
1-4-1 Deghosting by acquisition strategy	7
1-4-2 The ghost on multisensor measurements	7
1-5 Thesis aim	10
2 Literature review	11
2-1 Single sensor adaptive deghosting	11
2-1-1 Single Sensor Adaptive Deghosting (SSAD)	11
2-1-2 Energy Minimization	13
2-2 Multisensor deghosting	17
2-2-1 Data-independent – Single streamer	17

PZSUM	17
Posthumus & Optimal Deghosting (ODG)	19
2-2-2 Data-dependent – Multistreamer	20
Generalized Matching Pursuit (GMP)	20
3 Data-driven multisensor PZSUM: PYZSUM	23
3-1 Motivation	23
3-1-1 Pitfalls of PZSUM	23
3-1-2 3D angle PZSUM	23
3-2 PYZSUM	26
3-2-1 Time Gradient Ratio	26
The single event requirement	27
Median filtering to improve S/N	27
3-2-2 Implementation	28
3-2-3 Illustration on synthetic data	33
Quality check of the deghosting results	37
QC of the synthetic results	37
3-3 Application to real data	43
3-3-1 Real data 1	43
3-3-2 Real data 2	48
4 Data-driven multisensor adaptive deghosting	53
4-1 Motivation	53
4-1-1 Pitfalls of Energy Minimization	53
4-1-2 Multisensor cost functions	53
4-2 Two and three component Cross Ghost	56
4-2-1 Two and three component Cross Ghost	56
4-2-2 Sensitivity analysis	57
The two components cross ghost	57
Sensitivity of the two and three component cross ghost	57
4-2-3 Implementation	60
4-2-4 Illustration on synthetic data	62
4-3 Application to real data	69
4-3-1 Real data 1	69
4-3-2 Real data 2	72
5 Discussion & Conclusion	75

A Derivations	83
A-1 Inverse ghost filter	83
A-2 Complex frequency technique	85
A-3 Mirror-sum & Posthumus	85
A-4 Least squares solution	86
A-4-1 Posthumus	86
A-4-2 Optimal Deghosting (ODG)	87
A-4-3 Robust optimal deghosting (RODG)	87
A-4-4 Cross ghost cost function	88
A-5 Sensitivity analysis of the cross ghost	88
A-5-1 Unknown time delay	89
A-5-2 Unknown reflection coefficient	91
A-5-3 Unknown incidence angle	93
B Additional images	97
B-1 Full deghosted shots	97
B-1-1 Real data 1	97
B-1-2 Real data 2	100
B-2 Cost functions at other time delays	102

List of Figures

1	Illustration of spatial aliasing	xix
1-1	2D illustration of the seismic ghost	2
1-2	Schematical illustration of the ghost filter.	2
1-3	The pressure ghost in the time and the frequency domain.	3
1-4	The ghost model in the plane wave domain	4
1-5	Schematical illustration of the inverse ghost filter.	5
1-6	Example of a stable and an unstable deghosting inverse filter.	6
1-7	Different acquisition methods with their ghost responses	8
1-8	Illustration of the ghost on the vertical particle velocity V_z	9
1-9	The multisensor ghost response	9
2-1	Illustration of ringing artefacts at the use of a wrong ghost delay	12
2-2	The intertwined relation between upgoing wavefield and ghost model	13
2-3	An illustration of the unconstrained energy minimization cost function	15
2-4	An illustration of the constrained energy minimization	16
2-5	Example of PZSUM and mirror-sum	18
3-1	Graphical explanation of the scaling applied by the 2D PZSUM.	24
3-2	Example of the 2D PZSUM failing to take into account the crossline propagation	25
3-3	The regularized scalar for the PZSUM in the $\tau-p_x$ domain.	29
3-4	Illustration of the PZSUM scalar in 2D.	29
3-5	The different TGR differentiation methods lead to different noise levels	31
3-6	Two different median filters applied to the TGR result	32
3-7	The total ‘3D’ scalar for the PYZSUM.	33
3-8	The synthetic data created using buried sources at locations specified in Table 3-1.	35
3-9	The $\tau-p_x$ panel of synthetic data, and the PYZSUM scalar derived from this data	36

3-10 Comparison between the PZSUM and PYZSUM on synthetic data	38
3-11 Comparison between the PZSUM and PYZSUM using moving windows, applied on synthetic data	39
3-12 Comparison between the PZSUM and PYZSUM by comparing the zero offset traces	40
3-13 Power spectral density plot for the PZSUM and PYZSUM on synthetic data	41
3-14 The autocorrelation function for the PZSUM and PYZSUM on synthetic data	42
3-15 The acquisition geometry of the real data set 1	43
3-16 The multisensor measurements of real data set 1	44
3-17 The real data set 1 deghosted with PZSUM, ODG and PYZSUM	45
3-18 Power spectral density plot for the real data set 1 for the PZSUM, ODG and PYZSUM	46
3-19 The autocorrelation function for the PZSUM and PYZSUM on the real data set 1	47
3-20 The multisensor measurements of real data set 2	48
3-21 The real data set 2 deghosted with PZSUM, ODG and PYZSUM	49
3-22 Power spectral density plot for the real data set 2 for the PZSUM, ODG and PYZSUM	50
3-23 The autocorrelation function for the PZSUM and PYZSUM on the real data set 2	51
4-1 Evaluation of multisensor cost functions to detect a time delay of 2 ms	55
4-2 Sensitivity of the 2 and 3 component cross ghost for a large time delay	58
4-3 Sensitivity of the 2 and 3 component cross ghost for a large time delay	58
4-4 Sensitivity of the cross ghost for a small time delay	59
4-5 Sensitivity of the 2 and 3 component cross ghost for a large time delay	59
4-6 Evaluating influence of the reflection coefficient on the cross ghost	60
4-7 The effect of window size on the $\tau-p_x$ transform result	61
4-8 The picked time delay and vertical slowness as picked by minimizing the cross ghost function	63
4-9 Comparison between the deterministic ODG and multisensor adaptive deghosting on synthetic data	65
4-10 Comparison between the deterministic ODG and multisensor adaptive deghosting by comparing the zero offset traces	66
4-11 Power spectral density plot for the ODG and multisensor adaptive deghosting on synthetic data	67
4-12 The autocorrelation function for the deterministic ODG and multisensor adaptive deghosting on synthetic data	68
4-13 The real data set 1 deghosted with PZSUM, ODG and multisensor adaptive deghosting	69
4-14 Power spectral density plot for the real data set 1 for the PZSUM, ODG and multisensor adaptive deghosting	70
4-15 The autocorrelation function for the deterministic ODG and the multisensor adaptive deghosting on the real data set 1	71
4-16 The real data set 2 deghosted with PZSUM, ODG and multisensor adaptive deghosting	72
4-17 Power spectral density plot for the real data set 2 for the PZSUM, ODG and multisensor adaptive deghosting	73

4-18	The autocorrelation function for the deterministic ODG and the multisensor adaptive deghosting on the real data set 2	74
A-1	An impulse response illustration of the forward and inverse ghost filter	84
A-2	The varying upper bound for the cross ghost cost function at varying time delays	90
A-3	The varying upper bound for the cross ghost cost function for varying $ r_0 $ and t_Δ	92
A-4	Numerical evaluation of the cross ghost cost function for varying t_Δ and p_z	96
B-1	Full original and deghosted shot for PYZSUM for real data set 1, with gain of t applied.	98
B-2	Full original and deghosted shot for the 3 component cross ghost adaptive deghosting for real data set 1, with gain of t applied.	99
B-3	Full original and deghosted shot for PYZSUM for real data set 2, with gain of t applied.	100
B-4	Full original and deghosted shot for PYZSUM for real data set 2, with gain of t applied.	101
B-5	The cost functions at a time delay of 25 ms, with a maximum frequency of 100 Hz.	102
B-6	The cost functions at a time delay of 5 ms, with a maximum frequency of 100 Hz.	103

Background

Conventions

The analysis of functions in this thesis is done on the L_I^2 linear space, I being the interval considered. We define the L_I^2 inner product between functions f and g as:

$$\langle f, g \rangle = \int_I f g^* dx, \quad (1)$$

where $*$ is the complex conjugate operator (i.e. $(a + bi)^* = (a - bi)$, with $i^2 = -1$).

We can increase the interval I to $\pm\infty$. The **L² norm** is then defined as:

$$\|g(t)\|_2 = \sqrt{\langle g(t), g(t) \rangle} = \sqrt{\int_{-\infty}^{\infty} gg^* dt} = \sqrt{\int_{-\infty}^{\infty} |g(t)|^2 dt}. \quad (2)$$

The resulting quantity is called ‘**energy**’ in signal processing.

Define the **Fourier Transform**, a basis transform of a finite energy function, say from time to frequency:

$$\hat{g}(f) = \mathcal{F}(g(t)) = \langle g(t), e^{2\pi ift} \rangle = \int_{-\infty}^{\infty} g(t) e^{-2\pi ift} dt. \quad (3)$$

The inverse operation is the **Inverse Fourier Transform**, an inner product with the conjugate exponential:

$$g(t) = \mathcal{F}^{-1}(\hat{g}(f)) = \langle \hat{g}(f), e^{-2\pi ift} \rangle = \int_{-\infty}^{\infty} \hat{g}(f) e^{2\pi ift} df. \quad (4)$$

Parseval’s theorem proves that the signal’s energy can also be found in the frequency domain:

$$\|f(t)\|_2 = \sqrt{\langle g(t), g(t) \rangle} = \sqrt{\langle \hat{g}(f), \hat{g}(f) \rangle} = \sqrt{\int_{-\infty}^{\infty} |g(t)|^2 dt} = \sqrt{\int_{-\infty}^{\infty} |\hat{g}(f)|^2 df}. \quad (5)$$

An important signal is the **delta function**, $\delta(t)$, which has an infinite spike of zero width when the argument $t = 0$. It is defined as a distribution with $\int_{-\infty}^{\infty} f(x)\delta(a - x) dx = f(a)$, so it can be considered as a way to **sample** another function at instance a . The Fourier transform of the delta function is:

$$\widehat{\delta(a-t)}(f) = \int_{-\infty}^{\infty} \delta(a-t)e^{-2\pi ift} dt = e^{-2\pi ifa}. \quad (6)$$

The **frequency spectrum** is defined as the modulus of the Fourier transformed function, here $|e^{-2\pi ifa}| = 1$. This means that the frequency spectrum of the delta function is 1 for all frequencies.

Geophysics deals with two signals: the earth its impulse response is probed by an input signal, and their combined response is sampled at intervals in time and/or space at the surface. In a **Linear Time Invariant system**, the full relation between input and recorded output is elegantly described, and we will assume this system is valid in this thesis. We break down the two signals below.

1. Probing the earth its impulse response $r(t)$ by an input signal $s(t)$ gives output $g(t)$. The operation is written as a **convolution product** in the time domain:

$$g(t) = (s * r)(t) = \int_{-\infty}^{\infty} s(n)r(t-n) dn. \quad (7)$$

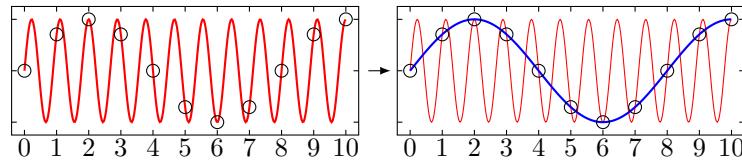
Notice that an input signal $\delta(t)$ would provide an ‘undisturbed’ recording of the earth its response ($(\delta * r)(t) = r(t)$). This is the desired result of a geophysical experiment.

The **convolution theorem** tells us that a convolution in the time domain is equal to multiplication in the frequency domain and vice versa. As a result, we can find the frequency domain expression of the output signal:

$$\hat{g}(f) = \hat{s}(f)\hat{r}(f). \quad (8)$$

Recall that the Fourier transform of $\delta(t)$ equals to $\hat{\delta}(f) = 1$, thus $\hat{g} = \hat{\delta}(f)\hat{r}(f) = \hat{r}(f)$. Again we see how the delta function would give an undisturbed recording of the earth its response. Any physically realizable wavelet, however, has a limited **bandwidth**: a limited frequency content. This means that frequencies not present in the signal (where $\hat{s}(f) = 0$) will not be in the output signal, i.e. not measured. The above is a ‘convoluted’ way to describe that the input signal is imprinted on the output signal.

2. The recording step affects the signal again. We cannot digitally record a continuous signal, but only sample the signal at discrete intervals of Δt in time and Δx in space. This operation may cause an effect called **aliasing**: the signal reconstructed from the samples differs from the original continuous input signal. Consider measuring the red sinusoid in the left picture below at every second (denoted by the circles). The most obvious reconstruction of the signal from the samples is the blue sinusoid on the right picture. Taking the discrete Fourier transform on aliased data will also show information at these aliased frequencies, not originally present in the continuous function. To prevent this ‘fake’ data, we must be careful with processing steps beyond the aliased frequencies.



The frequency up to which no aliasing is present in the time domain is called the Nyquist frequency, characterized by two samples per frequency:

$$f_{\text{nyquist}} = 0.5 f_{\text{sampling}} = \frac{1}{2\Delta t}. \quad (9)$$

A similar effect plays in the space-time domain, where it is not a lower frequency which can also explain the data, but an opposite dip which can also explain the data. The function describing the frequency up to which the data is not spatially aliased is:

$$f_{\max} = \frac{1}{2\Delta x} \frac{v_p}{\sin(\theta)}, \quad (10)$$

where Δx represents the interval between recording stations, v_p the acoustic wave velocity, θ the apparent dip. See Figure 1.

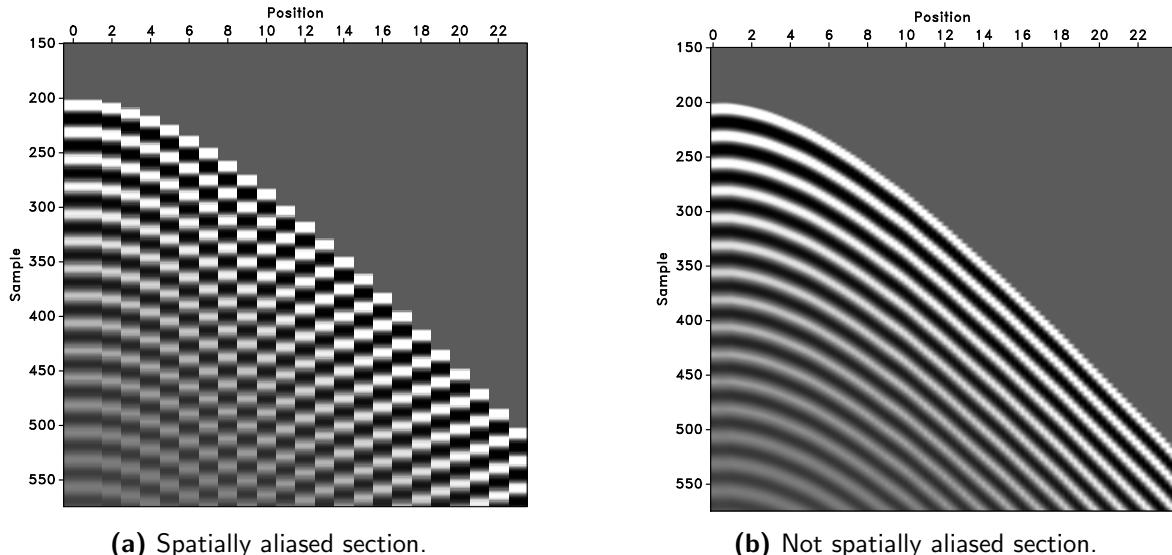


Figure 1: Illustration of spatial aliasing with hyperbola's dipping down towards the right. **(a)** Beyond a certain angle, one can also see the signal dipping in an opposite direction: an effect of aliasing. **(b)** A smaller Δx sampling captures the correct dip of the signal. Images from [Liu & Fomel \(2011\)](#).

The sampling of the earth its response function thus inherently alters the desired signal.

Plane wave decomposition

Consider the one-way hyperbolic linear (i.e. ‘plane’) wave equation ([LeVeque, 1992](#)):

$$\frac{\partial P}{\partial \vec{x}} + \vec{p} \frac{\partial P}{\partial t} = 0, \quad (11)$$

with \vec{p} the slowness of propagation of a wave P in terms of incidence angle θ and azimuth ϕ and velocity v_p :

$$\vec{p} = \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} = \begin{pmatrix} \sin \theta \cos \phi \\ \sin \theta \sin \phi \\ \cos \theta \end{pmatrix} \frac{1}{v_p}. \quad (12)$$

The three components are related through $p_x^2 + p_y^2 + p_z^2 = \frac{1}{v_p^2}$. The general solution is

$$P(x, t) = P_0(t - \vec{p} \cdot \vec{x}) = P_0(t - xp_x - yp_y - zp_z), \quad (13)$$

where P_0 is the input wavelet. This corresponds to simply translating the source wavelet through the space-time domain with velocity v_p . One potential choice of wavelet is the ‘monochromatic’ (e.g. constant frequency cosine) wave:

$$P(x, y, z, t) = \text{Real} \left[A(f) e^{i2\pi f(t - xp_x - yp_y - zp_z)} \right], \quad (14)$$

where $A(f)$ is a scalar describing the amplitude of the wavelet at specific frequency.

The plane wave is a useful concept, because it tells us exactly from which direction the energy is received. A standard seismic section does not immediately give us this information, but simply records the energy along time and space. To interpret and process the data in terms of their incidence angles, it helps to transform the data into the plane wave domain. The process of going from the space-time domain towards a domain dependent on the slowness \vec{p} is called a **plane wave decomposition**. Two examples are given below.

1. **FK transform:** Consider a real-valued function in the space-time domain $h(x, y, z, t)$. We can apply the Fourier transform over time and the horizontal (x and y) directions to obtain the f - k transform:

$$\tilde{h}(k_x, k_y, z, f) = \iiint_{-\infty}^{\infty} h(x, y, z, t) e^{-i2\pi(ft - k_x x - k_y y)} dt dx dy. \quad (15)$$

The variables k_x and k_y are called ‘wavenumbers’, and decompose the data into ‘spatial frequencies’. We can correspondingly transform back to the original space-time domain with three inverse Fourier transforms. The Fourier transform for real data is conjugate symmetric: the information for negative frequencies is the conjugate of the information in the positive frequencies. We can thus take twice the ‘real’ part of the inverse Fourier transform over time to get the same output ([Wapenaar & Berkhouwt, 1989](#)):

$$h(x, y, z, t) = 2 \text{Real} \left[\int_0^{\infty} \left(\iint_{-\infty}^{\infty} \tilde{h}(k_x, k_y, z, f) e^{i2\pi(ft - k_x x - k_y y)} dk_x dk_y \right) df \right]. \quad (16)$$

Note the similarity between Equations (14) and (16). Apparently, for a fixed z , the inverse Fourier transform sums a set of monochromatic plane waves with amplitude \tilde{h} , and we can see the wavenumber represents:

$$k_x = p_x f, \quad (17)$$

and

$$k_y = p_y f. \quad (18)$$

For this reason, the f - k_x - k_y (short: f - k) domain can be considered a plane wave domain, separating events with different slownesses. This allows us to filter events on the basis of their incidence and azimuthal angle. The great property of this domain is that the forward and inverse operation are rapid transformations using the Fast Fourier Transform, and the operation is fully invertible. The downside, at least in the context of this thesis, is that we do not get access to p_x and p_y but the slowness multiplied by the frequency.

- 2. Tau P transform:** The τ - p transform is another plane wave decomposition method. This transform gives immediate access to the actual p_x and p_y , and separates them in time. The method performs a summation of the data along a straight lines of constant p_x and p_y , and maps these values to the τ - p_x - p_y domain. The intercept time τ corresponds to the time in the data from which the varying lines of p_x and p_y are drawn. Consider again a real-valued function $h(x, y, z, t)$. Notice the similarity to Equation (13), in describing the operation mathematically (Zhou & Greenhalgh, 1994):

$$\bar{h}(\tau, p_x, p_y, z) = \iint_{-\infty}^{\infty} h(x, y, z, \tau - xp_x - yp_y) dx dy, \quad (19)$$

$$= \iint_{-\infty}^{\infty} \underbrace{\left(\int_{-\infty}^{\infty} \hat{h}(x, y, z, f) e^{2\pi i f(\tau - xp_x - yp_y)} df \right)}_{\text{forward and inverse Fourier transform over time}} dx dy, \quad (20)$$

$$= \int_{-\infty}^{\infty} \left(\iint_{-\infty}^{\infty} \hat{h}(x, y, z, f) e^{-2\pi i f(xp_x + yp_y)} dx dy \right) e^{2\pi i f\tau} df. \quad (21)$$

In practice, this means we can achieve a τ - p_x - p_y transform by taking the Fourier transform over time of our data, multiply and sum with the slowness exponential, and then take the inverse Fourier transform. An inverse τ - p transform is realized by summing along $\bar{h}(p_x, p_y, z, \tau + xp_x + yp_y)$, but this does not recover the initial data completely. Higher frequencies have been relatively attenuated in the τ - p panel and are not recovered in the inverse transform. One way to make the τ - p panel of higher resolution (preserving the high frequencies) is to perform deconvolution in the forward transform – essentially a step to make sure that the conjugate operation would give back the input data. Even then, the τ - p transform is not fully invertible. In my experience, errors on the order of 1% of the maximum amplitude occur in the inverse transform. This is fundamentally different from the f - k transform.

Lastly, to recover as much as possible from the signal, we need to record all the slowness values in the data. To protect against aliasing, this means we must take a slowness step Δp dependent on the maximum frequency and offset in our data (Turner, 1990):

$$\Delta p \leq \frac{1}{f_{\max} x_{\max}} \leq \frac{2\Delta t}{x_{\max}}. \quad (22)$$

The latter expression replaces f_{\max} by the Nyquist frequency.

The τ - p transform has the big advantage of separating events of different slowness values, while retaining the time axis. The disadvantage is that the inverse operation does not fully reproduce the input data, and the method is several orders of magnitude slower than the f - k transform.

Chapter 1

The ghost problem

1-1 Broadband seismics & the receiver ghost

Reflection seismology images the subsurface by recording the earth its response to pressure waves. To get a good signal-to-noise ratio (S/N) in this image, one has to record narrow and clean wavelets. In the frequency domain this translates to getting a powerful, broadband, minimum phase recording ([Yilmaz, 2001](#)). The broad spectrum enables deep structure imaging with the low frequencies, as well as good temporal resolution through the high frequencies. The seismic acquisition and processing workflow must preserve this broadband signal to allow accurate geological interpretations. The seismic ghost is a phenomena which hampers this goal in marine seismics by degrading the high S/N broadband signal ([Day et al., 2013](#)).

The interface between sea-water and air is a near-perfect reflector ($r_0 \approx -1$). Thus, an upgoing seismic wave reflects off the sea-surface and propagates downward with reversed polarity. Marine seismic recordings thus contain both the desired upgoing wave and its sea-surface reflection, called the ‘receiver ghost’. The receiver ghost interferes with the upgoing wave, causing periodic low S/N notches in the recorded spectrum. The notches reduce the usable bandwidth and degrade the resolution of the seismic data, causing a blurring effect to the seismic images (Figure 1-1). A similar phenomenon is the ‘source ghost’, a duplication of the source wavelet as it reflects off the sea surface after emission. In this thesis we focus on the receiver side ghost.

The ghost thus degrades the broadband marine seismic signal. We want to remove the ghost wavefield from the data to restore the original broadband nature of the data. This is key for high resolution seismic imaging, as well as a requirement for several standard processing steps such as multiple removal and velocity analysis ([Mayhan & Weglein, 2013](#)).

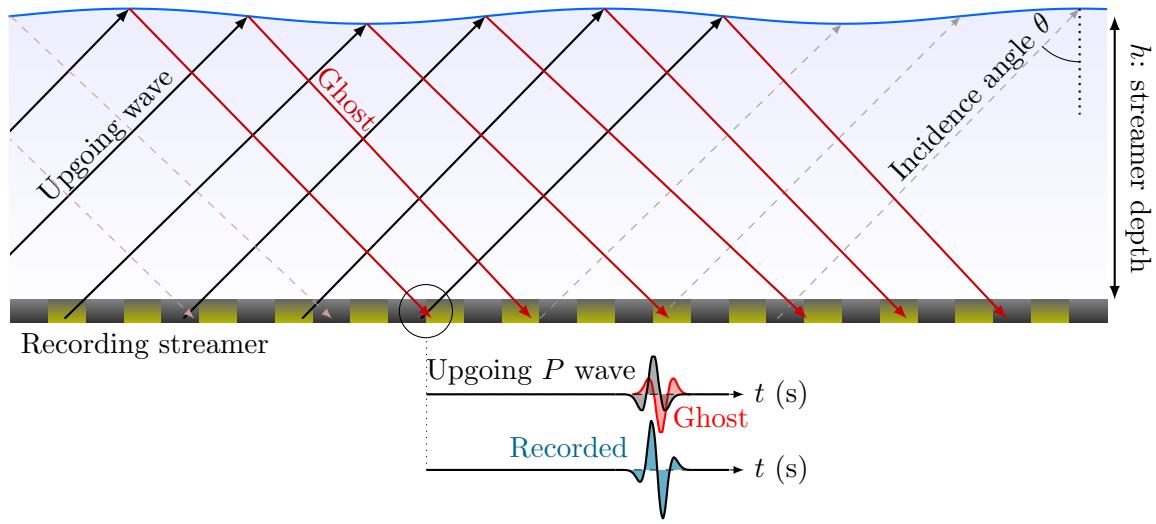


Figure 1-1: 2D illustration of the seismic ghost. An upgoing acoustic plane wave (black parallel rays) reflects off the sea surface to produce a downgoing wavefield: the ghost (red). The combination of the upgoing and ghost wavefields results in the recorded wavelet (blue), which is a distorted version of the desired upgoing wavelet: its phase and frequency content have changed. The ghost may not be a pure plane wave anymore due to reflection from the wavy sea surface, illustrated by the non-parallel red rays.

1-2 The ghost as filter

We want to recover the information in the low S/N notches of the spectrum, to restore the broadband signal at all frequencies. We will first study the ghost as a filter on the upgoing wavefield, and then remove the effect of the filter. Assume the ‘ghost filter’ acts on the upgoing pressure wave P_{up} to produce the recorded wavelet P_{total} .

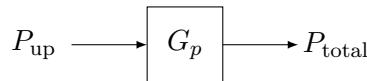


Figure 1-2: Schematic illustration of the ghost filter. The desired upgoing signal P_{up} is filtered by the ghost filter G_p , resulting in the ghosted recording P_{total} .

1-2-1 The ghost as frequency filter

Assume a flat sea surface and upgoing plane wave. The time delay between the upgoing wave and its ghost at the streamer is then:

$$t_{\Delta} = \frac{2h}{v_p} \cos \theta, \quad (1-1)$$

where h is the streamer depth, $v_p \approx 1480$ m/s the acoustic wave speed in sea water, and θ the incidence angle of the wave. The time delay is largest at vertical incidence ($\theta = 0^\circ$).

The recorded wavefield in the time (t) domain is the sum of the up- and downgoing wavefields. Assuming a flat sea surface, the downgoing signal is a delayed and polarity reversed copy of the upgoing signal. The total recorded wavefield is thus:

$$P_{\text{total}}(t, t_\Delta) = P_{\text{up}}(t) - |r_0|P_{\text{up}}(t - t_\Delta) = \left(\delta(t) - |r_0|\delta(t - t_\Delta) \right) * P_{\text{up}}(t), \quad (1-2)$$

with the latter expression a convolution product between P and the ‘ghost model’. Observe that this filter is a simple finite impulse response (FIR) filter, with only two non-zero elements. The frequency (f) domain expression follows from the convolution theorem,

$$\hat{P}_{\text{total}}(f, t_\Delta) = \underbrace{\left(1 - |r_0|e^{-2\pi i f t_\Delta} \right)}_{\hat{G}_p(f, t_\Delta)} \cdot \hat{P}_{\text{up}}(f) = \hat{G}_p(f, t_\Delta) \hat{P}_{\text{up}}(f), \quad (1-3)$$

where $(\hat{\cdot})$ represents a frequency domain function. Assuming a reflection coefficient $r_0 = -1$, we note that $|\hat{G}_p| = 2|\sin(\pi f t_\Delta)|$, i.e. its spectrum periodically varies between 0 and 2. The ghost model $\hat{G}_p(f)$ thus acts as a frequency filter on the upgoing wavefield, periodically attenuating and boosting certain frequencies (Figure 1-3).

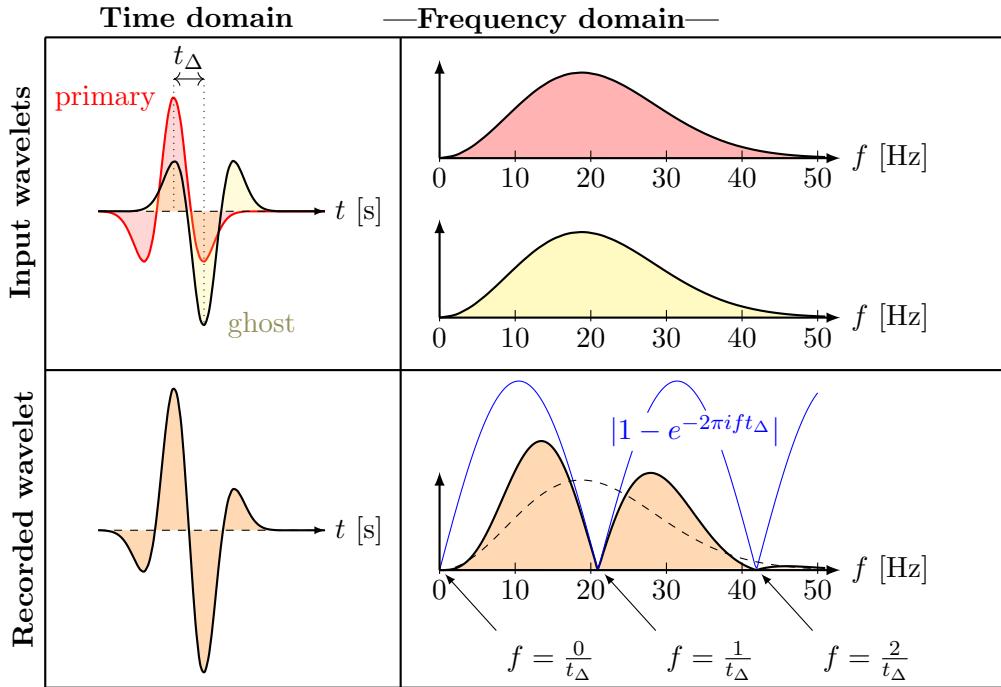


Figure 1-3: Effect of the pressure ghost on a Ricker wavelet displayed in the time domain (left: the sum of the primary and the ghost signal) and frequency domain (right: the multiplication of the primary spectrum (dashed) with the ghost model (blue)). The recorded spectrum is a distorted version of the desired spectrum.

1-2-2 The ghost as frequency-wavenumber and frequency-slowness filter

Describing the ghost for events with varying incidence angles is tedious in the offset-time (x - y - t) domain. Furthermore, describing the ghost for overlapping events is not straightforward.

A natural way to treat the ghost for all events on a seismic section is to apply plane wave decomposition (PWD). This decomposition groups events of constant incidence angle θ in the offset-time domain to a new domain. Two standard PWD domains are the frequency-wavenumber (f - k) and the frequency-slowness (f - p) domains. Assuming measurements in the inline x (along the streamer) and crossline y (across the streamer) direction, we rewrite the ghost delay:

- Frequency-wavenumber (f - k):** Substitute the vertical wavenumber $k_z = \frac{f \cos \theta}{v_p} \rightarrow t_\Delta = \frac{2h}{v_p} \cos \theta = \frac{2hk_z}{f}$. We know $k_z(k_x, k_y) = \sqrt{\frac{f^2}{v_p^2} - k_x^2 - k_y^2}$, with k_x and k_y the inline and crossline wavenumbers respectively. The ghost in the f - k_x - k_y (̄) domain:

$$\tilde{P}_{\text{total}}(f, k_x, k_y) = \underbrace{\left(1 - |r_0| e^{-2\pi i 2hk_z}\right)}_{\tilde{G}_p(k_z)} \cdot \tilde{P}_{\text{up}}(f, k_x, k_y) = \tilde{G}_p(k_z) \tilde{P}_{\text{up}}(f, k_x, k_y). \quad (1-4)$$

- Frequency-slowness (f - p):** Rewrite t_Δ using the vertical slowness $p_z = \frac{k_z}{f} = \frac{\cos \theta}{v_p} \rightarrow t_\Delta = 2hp_z$. Similarly, knowing $p_z(p_x, p_y) = \sqrt{\frac{1}{v_p^2} - p_x^2 - p_y^2}$, with inline and crossline slowness p_x and p_y , we consider the ghost in the f - p_x - p_y (̄) domain:

$$\bar{P}_{\text{total}}(f, p_x, p_y) = \underbrace{\left(1 - |r_0| e^{-2\pi i 2hfp_z}\right)}_{\bar{G}_p(f, p_z)} \cdot \bar{P}_{\text{up}}(f, p_x, p_y) = \bar{G}_p(f, p_z) \bar{P}_{\text{up}}(f, p_x, p_y). \quad (1-5)$$

The two plane wave ghost models are shown for $h = 10$ m in Figure 1-4.

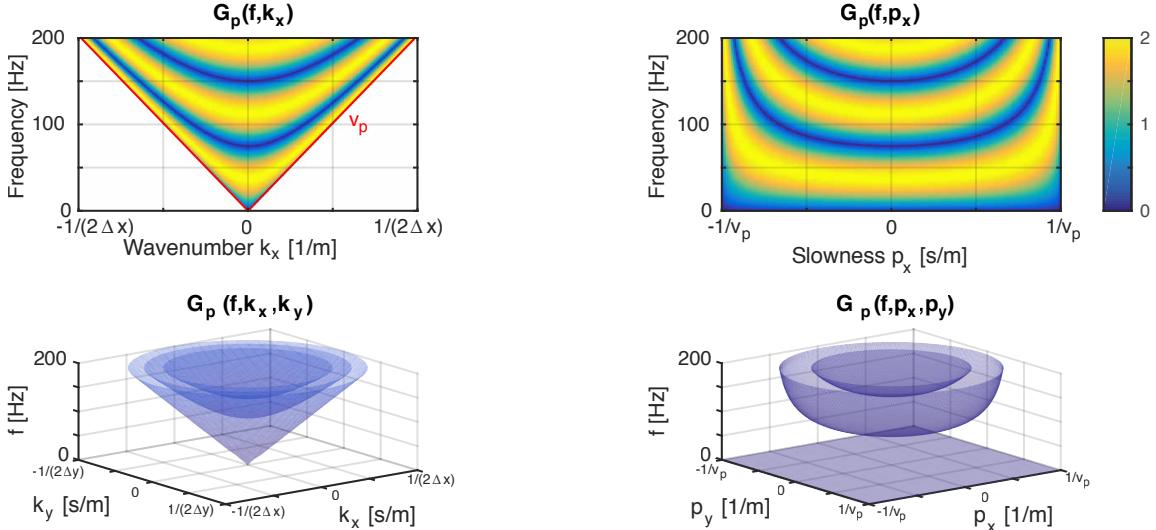


Figure 1-4: The ghost model in the plane-wave domains f - k and f - p , showing periodic peaks and notches as a function of incidence angle θ and frequency f . The ghost model for fixed conditions is radially symmetric, the bottom figure shows only the notches of the full 3D ghost model. Δx and Δy represent the inline and crossline receiver spacing respectively.

1-3 Deterministic deghosting

1-3-1 Inverse filter

To remove the ghost (hereafter referred to as *deghosting*), we must thus invert the ghost filter acting on the data (Figure 1-5). The ghost in the time domain is a convolution between an

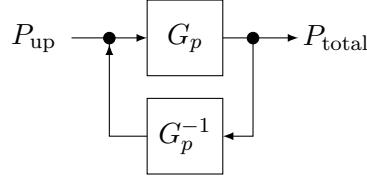


Figure 1-5: Schematic illustration of the inverse ghost filter, which undoes the ghost operation and returns the upgoing signal P_{up} from the ghosted wavefield P_{total} .

FIR filter and the upgoing wavefield. Its inverse, the deconvolution filter, is thus an infinite impulse response (IIR) filter, which may be unstable. In the frequency domain, this translates to a division of the data by the ghost model. Assuming complete reflection, $r_0 = -1$:

$$\hat{P}_{\text{total}}(f) = \underbrace{\left(1 - e^{-2\pi if t_\Delta}\right)}_{\hat{G}_p} \hat{P}_{\text{up}} \implies \hat{P}_{\text{up}}(f) = \left(\frac{1}{\hat{G}_p(f)}\right) \hat{P}_{\text{total}}(f). \quad (1-6)$$

This inverse filter \hat{G}_p^{-1} has a pole ($\frac{1}{0}$) at the ghost notches. To mitigate this pole, we can:

1. Multiply the conjugate ghost with the numerator and denominator to preserve the phase effect, and stabilize the amplitude with a small number in the denominator (see Appendix A-1).

$$\hat{P}_{\text{up}}(f) \approx \left(\frac{\hat{G}_p^*(f)}{\hat{G}_p(f)\hat{G}_p^*(f) + \varepsilon}\right) \hat{P}_{\text{total}}(f), \quad (1-7)$$

where ε is a small number such as 0.01 (compare, the other term $|G_p|^2$ varies between 0 and 4). The data at the notch frequencies are not recovered, as G_p^* in the numerator has value 0 at the ghost notch. We call this technique regularization.

2. Stabilize the filter by artificially decaying the data (the complex frequency technique: Phinney (1965), or Appendix A-2). This prevents the pole in the filter:

$$\begin{aligned} P_{\text{up}}(t) &\approx \mathcal{F}^{-1} \left(\frac{\mathcal{F}(P_{\text{total}}e^{-at})}{\hat{G}_p(f - \frac{ai}{2\pi})} \right) e^{at}, \\ &\approx \mathcal{F}^{-1} \left(\frac{\mathcal{F}(P_{\text{total}}e^{-at})}{1 - e^{-2\pi i(f - \frac{ai}{2\pi})t_\Delta}} \right) e^{at} = \mathcal{F}^{-1} \left(\frac{\mathcal{F}(P_{\text{total}}e^{-at})}{1 - e^{-at_\Delta}e^{-2\pi if t_\Delta}} \right) e^{at}. \end{aligned} \quad (1-8)$$

The latter notation clarifies how the approach, essentially, adds a constant reflection coefficient of $|r_0| = e^{-at_\Delta} < 1$ to the data. In other words, the upgoing wave is always stronger than the ghost due to the decaying function. The pole is thus removed, allowing the inverse filter. However, if noise is present in frequencies close to the ghost notches, the noise will be amplified and degrade the S/N.

Assuming a white noise baseline, the S/N will be worst at the ghost notches. Amplifying frequencies around the ghost notch thus amplifies more ‘noise’ than ‘signal’. The standard regularization is thus the best candidate to deghost data with noise, by not amplifying the low S/N frequencies. See Figure 1-6. We establish that a stable deghosting operation, robust against noise, does not recover the true broadband signal at and near the notches.

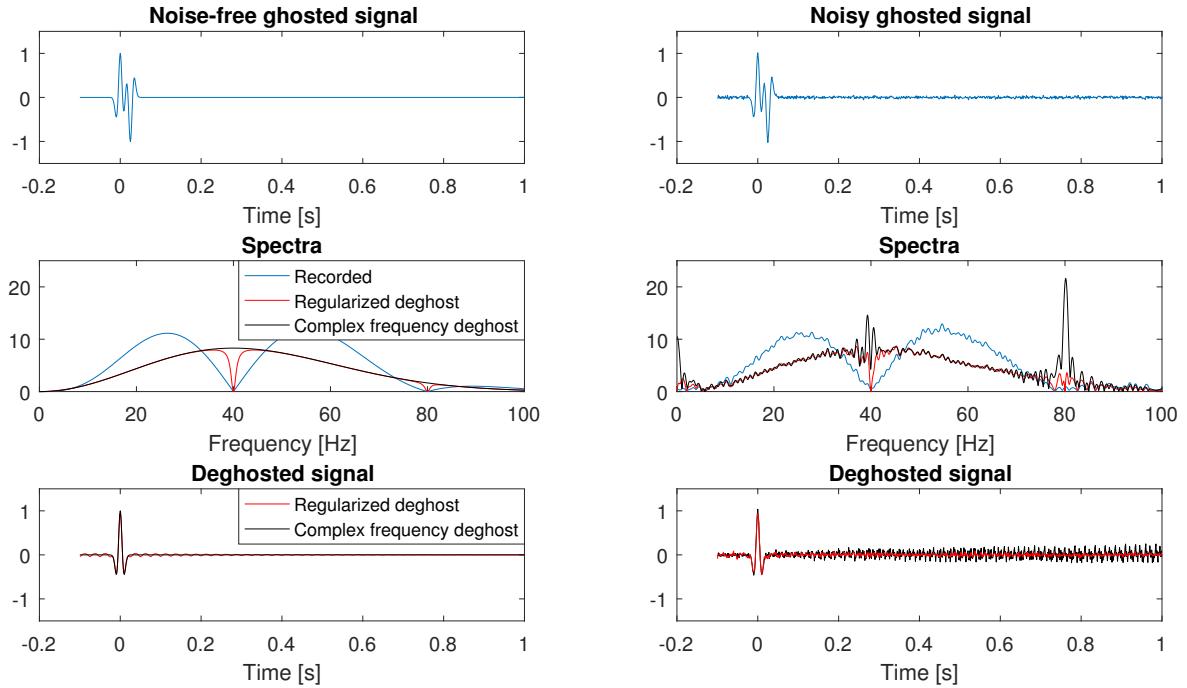


Figure 1-6: Example of a stable and an unstable deghosting inverse filter. The regularization technique does not recover the notch frequency, presenting as ringing with a period of the ghost notch – but the technique retains the original S/N characteristics in case of noise. Conversely, the complex frequency technique does recover the notch frequency, but degrades the S/N by amplifying those frequencies of poor S/N.

1-3-2 Drawbacks of deterministic deghosting

Figure 1-5 illustrates that the inverse filter G_p^{-1} must correspond to the forward ghost filter G_p , to recover the true broadband seismic signal. If the applied inverse filter differs from the ghost filter, we first of all do not recover the true upgoing wave, and may also generate artefacts noted as ‘ringing’. The frequency of these periodic artefacts correspond to the notch frequencies of the incorrect inverse ghost filter, boosting the local signal amplitudes.

If the ghost model ($t_\Delta = \frac{2h}{v_p} \cos \theta$ and $|r_0|$) is assumed to be known upfront, the deghosting is called *deterministic deghosting*. This assumption is not always met in realistic survey settings, as the parameters vary along with the conditions of the sea and subsurface. The streamer depth h may vary through undercurrents. Moreover, considerable wave action varies the true streamer depth through time. The acoustic velocity in water v_p varies with pressure,

temperature and salinity. Sharp contrasts in water temperature – thermoclines – and salinity – haloclines – thus vary the ‘effective’ velocity along the survey. The incidence angle θ is calculated by transforming the data to a PWD domain. In case of an unruly sea surface, the upgoing incidence angle of the wave differs from the downgoing incidence angle of the wave, in which case the time delay does not follow the simple model $t_\Delta = \frac{2h}{v_p} \cos \theta$ anymore. Moreover, a 3D PWD of the data is not always possible. Whereas the inline receiver spacing along a streamer is usually small ($\Delta x \approx 3\text{-}12$ m), the crossline spacing may reach up to 125 m. Furthermore, the cross-line recording aperture is limited – usually 12 data points, compared to several hundred recorders along the inline direction. In practice, the PWD thus tends to be restricted to the inline direction only, as the wavefield is severely aliased in the crossline direction. A 2D PWD does not provide the *true* incidence angle, but merely the apparent *inline* incidence angle. Crossline propagation of energy is thus not taken into account. Lastly, the acoustic impedance (ρv_p , or, density times acoustic wave speed) contrast between water and air may vary along with changes in temperature and salinity. With a changing contrast, the reflection coefficient may vary along time and space.

A deterministic deghosting approach fails if the assumed ghost model does not correspond to the actual ghost in the data. The incorrect inverse ghost filter may generate artefacts rather than restore the true signal bandwidth. This may seem like a handicap of the inverse filter, but is in fact exploited by some *adaptive* deghosting methods, which estimate an adequate ghost model by finding the ghost model that minimizes the ringing in the data. Adaptive methods can account for the varying conditions in a realistic survey setting, thus aid the recovery of the true upgoing wavefield.

1-4 Acquiring more ghosts

1-4-1 Deghosting by acquisition strategy

Surveys used to be designed to put the ghost notches outside of the desired bandwidth for a particular target, requiring little to no processing to still image the target. However, this led to compromises where multiple targets were not all illuminated optimally. New acquisition methods try not to lessen the ghost, but measure different ghost characteristics. The combination of the different ghosts then provides increased S/N at a wide range of frequencies (Day et al., 2013). Typical industry examples are shown in Figure 1-7.

1-4-2 The ghost on multisensor measurements

We will focus on the multicomponent, or, multisensor measurements. The particle velocity V_z is related to the pressure P with the equation of motion,

$$\nabla P = -\rho \frac{\partial \vec{V}}{\partial t}, \quad (1-9)$$

with ρ the density in kg/m^3 and $\vec{V} = [V_x, V_y, V_z]^T$ the components of the particle velocity in m/s . The ghost is thus not only present on the pressure measurements, but also on the particle velocity measurements. Whereas the inline and crossline particle velocities (V_x and V_y

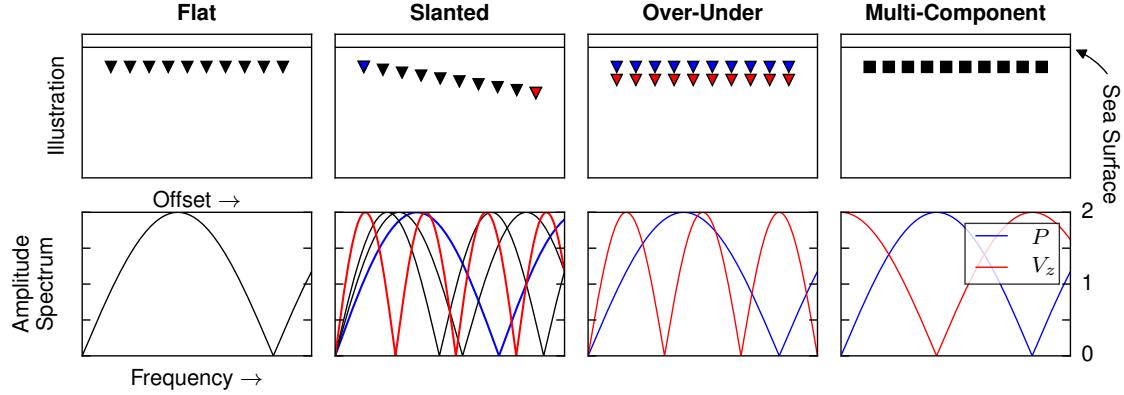


Figure 1-7: The traditional seismic survey (flat, left) has a single ghost response that degrades the signal its frequency spectrum. Advanced acquisition methods try to work around this effect by recording data with a several ghost responses which provide a good S/N at broader frequency range when combined. Figure from Schuberth (2015).

respectively) have the same ghost response as the pressure wave – that is, reversing polarity between the upgoing and downgoing wave – the response for the vertical particle velocity is different. Assuming horizontal homogeneity, we can write the equation of motion in the $f-k_x-k_y-z$ domain:

$$i2\pi k_z \tilde{P} = i2\pi f \rho \tilde{V}_z \rightarrow \tilde{P} = \frac{f\rho}{k_z} \tilde{V}_z = \frac{\rho}{p_z} \tilde{V}_z. \quad (1-10)$$

Observe that the sign on \tilde{P} and \tilde{V}_z is equal for the upgoing wave ($p_z = \frac{\cos\theta}{v_p} > 0$) but opposite for the downgoing wave ($p_z = \frac{\cos\theta}{v_p} < 0$). This relation shows that, as the pressure data reverses its polarity between upgoing and downgoing wave, we observe twice the same polarity on the vertical particle velocity. The ghost model is thus $1 + |r_0|e^{-2\pi ift_\Delta}$ (Figure 1-8).

The complementary nature of the ghost on P_{total} and $V_{z,\text{total}}$ opens up new possibilities for deghosting. We deal with a single upgoing wavefield causing up to four measurements (P and the three components of \vec{V}), i.e. Figure 1-9. The system in the $f-k_x-k_y$ domain can be described as the following relation:

$$\begin{pmatrix} \tilde{P}_{\text{total}} \\ \tilde{V}_{x,\text{total}} \\ \tilde{V}_{y,\text{total}} \\ \tilde{V}_{z,\text{total}} \end{pmatrix} = \begin{pmatrix} \tilde{G}_p \\ \tilde{G}_x \\ \tilde{G}_y \\ \tilde{G}_z \end{pmatrix} \tilde{P}_{\text{up}} = \begin{pmatrix} 1 - |r_0|e^{-i\omega 2hk_z} \\ \left(\frac{k_x}{\omega\rho}\right)(1 - |r_0|e^{-i\omega 2hk_z}) \\ \left(\frac{k_y}{\omega\rho}\right)(1 - |r_0|e^{-i\omega 2hk_z}) \\ \left(\frac{k_z}{\omega\rho}\right)(1 + |r_0|e^{-i\omega 2hk_z}) \end{pmatrix} \tilde{P}_{\text{up}}. \quad (1-11)$$

The scaling factors in front of the velocity ghost models originate from the equation of motion (1-10), thus provide the scaling between particle velocities and pressure.

The complementary nature of the ghost notches on P and V_z allows us to restore the broadband nature of the upgoing wavefield: the part of the spectrum with lowest S/N of the pressure is the highest S/N part of the spectrum of \tilde{V}_z and vice versa.

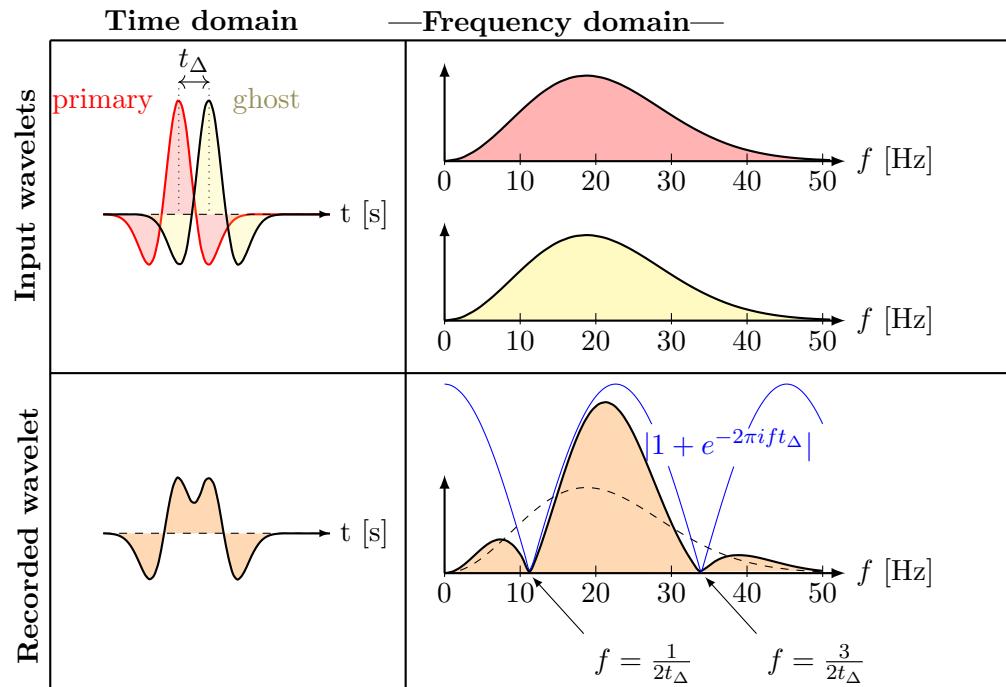


Figure 1-8: The ghost on the vertical particle velocity V_z has twice the same polarity, thus is additive rather than subtractive as the ghost on P , and therefore has a different frequency notch.

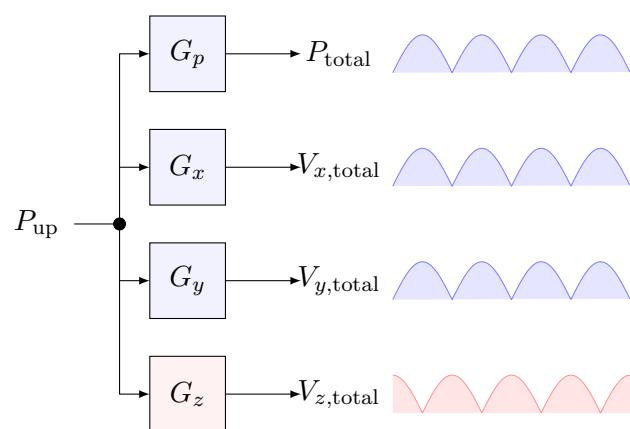


Figure 1-9: The multisensor ghost model may be considered as a filter acting on a single upgoing wavefield which results in four ghosted measurements. The ghost notches for the pressure and horizontal particle velocities are equal, whereas the vertical particle velocity has its notches shifted with half a period.

1-5 Thesis aim

Various methods of multisensor deghosting exist. However, these methods all make deterministic assumptions on the ghost model. If such assumptions are invalid, the deghosting will not recover the true broadband nature of the seismic data. The aim of this thesis is to provide data-adaptive multisensor deghosting techniques for single streamer data. The data-adaptive part means that the exact deghosting filter depends on the data itself. The single streamer part means we limit ourselves against spatial aliasing – we will use the crossline particle velocity V_y measurements to estimate any cross-line energy propagation if needed.

Chapter 2 covers the theory for single sensor data-adaptive, and multisensor deterministic, deghosting methods. In chapter 3 we will propose a novel multisensor, fully data-driven technique to deghost the data taking into account the full 3D propagation of the wavefield. This method will overcome standard single-streamer deghosting methods which assume inline propagation of energy only – suggesting the biggest advantage of the method must be expected for strong crossline energy propagation. The method will be tested on synthetic and real data. In chapter 4 we will propose another novel data-driven technique to deghost multisensor data, by estimating the ghost model from the data itself. This method overcomes limitations of adaptive single sensor deghosting techniques, which for example have a sensitivity limited to the signal bandwidth. The proposed multisensor adaptive deghosting method is analyzed and shown to be accurate, regardless of the signal bandwidth. The implementation details are discussed, after which the method is shown to work on synthetic and real data. A conclusion and discussion will follow in Chapter 5.

Chapter 2

Literature review

2-1 Single sensor adaptive deghosting

Inverse filtering is sensitive to the ghost model. Figure 2-1 illustrates the inverse filter with various time delays and reflection coefficients. The deghosting result ‘rings’ when using a wrong delay time, visible in the form of periodic artefacts. Deghosting the data using wrong the reflection coefficient also creates artefacts, albeit less severe. This motivates the need for an algorithm that adaptively estimates the ghost model parameters from the data. Single sensor adaptive methods generally estimate the ghost model by minimizing the ringing after deghosting.

2-1-1 Single Sensor Adaptive Deghosting (SSAD)

Current adaptive deghosting methods ([Schuberth, 2015](#); [Rickett et al., 2014](#); [Zhang et al., 2015](#)) assume that the true upgoing wavefield contains little to no ringing. They try a series of potential deghosting operators, and pick a solution that minimizes a cost function. The state-of-the-art is the Single Sensor Adaptive Deghosting (SSAD, [Rickett et al., 2014](#)), in its simplest form:

$$\text{SSAD} : \min_{t_\Delta, \hat{P}_{\text{up}}} \left\| \hat{P}_{\text{data}} - \hat{G}_p(t_\Delta) \hat{P}_{\text{up}} \right\|_2^2 + \lambda \| P_{\text{up}} \|_1, \quad (2-1)$$

with the L2 norm describing the residual energy between the recorded data P_{data} and model, and the L1 norm describing the sparsity of the synthetic upgoing wavefield. Parameter λ weighs the two objectives. The system finds an upgoing wavefield P_{up} by finding an optimal solution to the objective function, rather than with an inverse filter. Regardless, minimizing the first term in the objective function still generates a $\hat{P}_{\text{up}} \approx \frac{1}{\hat{G}_p(t_\Delta)} \hat{P}_{\text{data}}$. Written in this form, the first term in the objective function becomes not only trivial, but highlights the connection between the upgoing wavefield and the ghost:

$$\left\| \hat{P}_{\text{data}} - \hat{G}_p(t_\Delta) \hat{P}_{\text{up}} \right\|_2^2 \approx \left\| \hat{P}_{\text{data}} - G_p(t_\Delta) \frac{1}{G_p(t_\Delta)} \hat{P}_{\text{data}} \right\|_2^2 = 0. \quad (2-2)$$

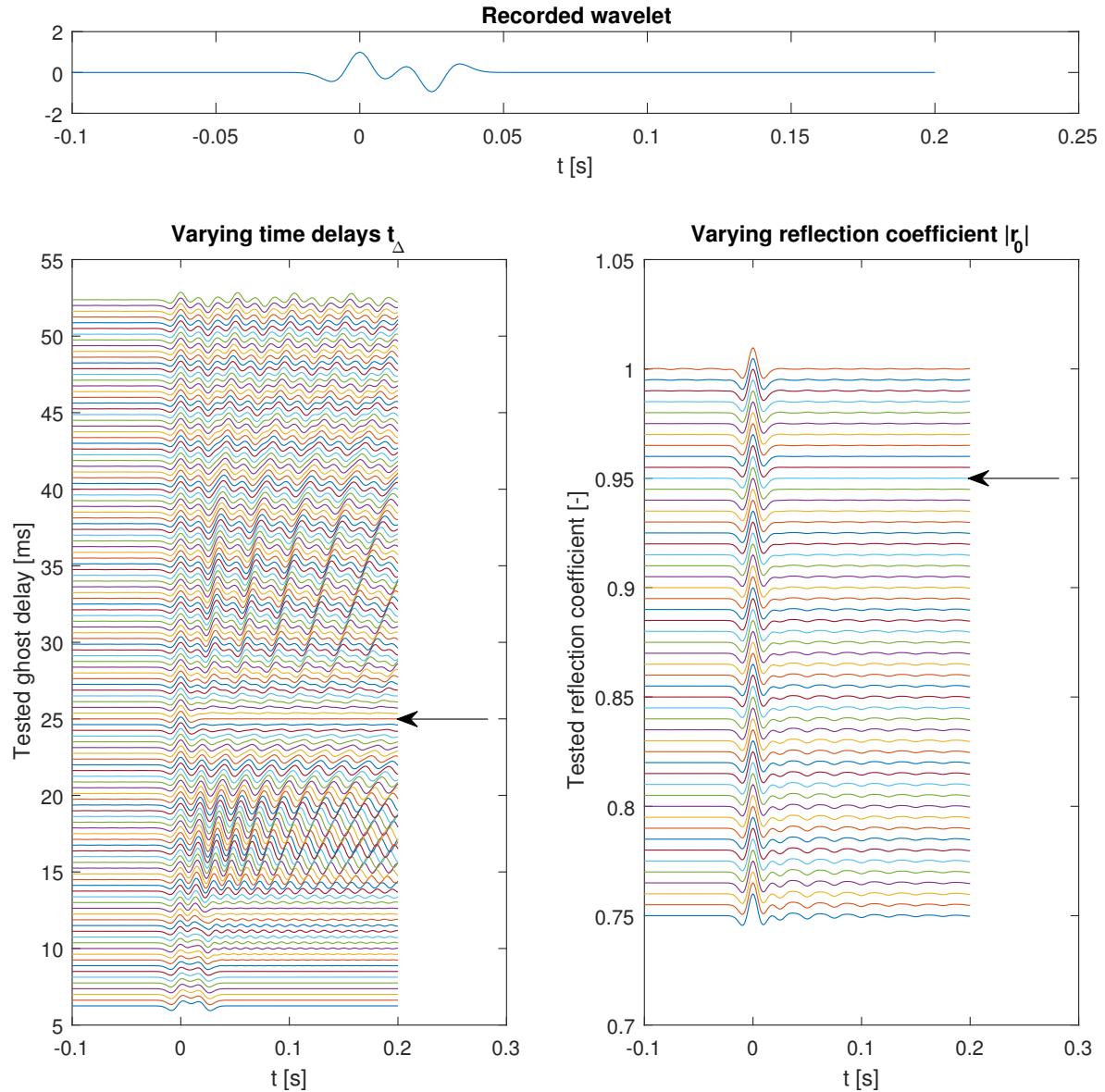


Figure 2-1: Deghosting a 40 Hz central frequency Ricker wavelet using the wrong ghost model leads to ringing artefacts. The 'true' values are a time delay $t_\Delta = 25$ ms and a reflection coefficient $|r_0| = 0.95$, the correct solution is pointed at with an arrow. While varying the time delay, the true reflection coefficient was used and vice versa.

This equation shows that *any* arbitrary ghost model $G_p(t_\Delta)$ has a corresponding upgoing wavefield that recreates the input data P_{data} . Any proper combination of $G_p(t_\Delta)$ and P_{up} can thus bring the first term in the SSAD to zero (Figure 2-2). This is why the SSAD penalizes non-sparse solutions in its second term, because a very ‘ringy’ P_{up} can still minimize the first term.

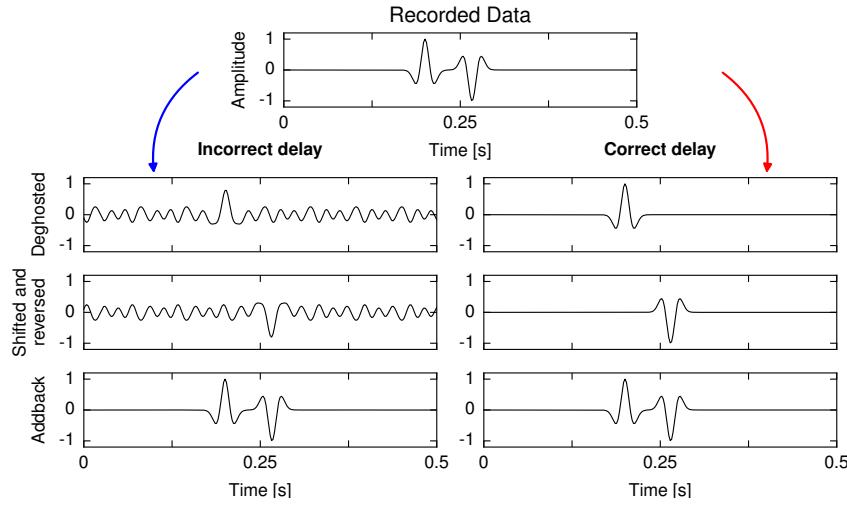


Figure 2-2: Two different ghost models create two different upgoing ('deghosted') wavefields that both explain the input data. The two upgoing signals differ in amount of ringing. Figure from Schuberth (2015). The assumption in adaptive deghosting methods, is that the wavefield with the least ringing is the true wavefield.

2-1-2 Energy Minimization

Schuberth (2015) argues that the intertwined relation between P_{up} and t_Δ means we can simplify the cost function to merely the second term: find a ghost delay that minimizes the ringing after deghosting. The faster optimization shows no degradation in performance compared to the SSAD. The method is called ‘energy minimization’, as a measure of the energy in the deghosted wavefield. For display purposes, the cost function is the reciprocal of the energy, such that the minimum energy gives the maximum cost function.

Energy minimization :

$$\max_{t_\Delta} \|U(t_\Delta)\|_2^{-2}, \text{ with } \|U(t_\Delta)\|_2^{-2} = \left\| \hat{U}(t_\Delta) \right\|_2^{-2} = \left(\int_{-\infty}^{\infty} \left| \frac{(\text{filtered}) \hat{P}_{\text{data}}}{\hat{G}_p(t_\Delta)} \right|^2 df \right)^{-1},$$

subject to $\frac{1}{f_{\text{max}}} \leq t_\Delta \leq \frac{2h + \Delta h}{v_p},$ (2-3)

$$\frac{\|U(t_\Delta)\|_2^{-2}}{\|P_{\text{data}}\|_2^{-2}} \geq \frac{3}{2},$$

$$\frac{\|U(nt_\Delta)\|_2^{-2}}{\|P_{\text{data}}\|_2^{-2}} \leq \frac{1}{4}, \quad n \in \{2, 3, 4, \dots\}.$$

The pure cost function is shown in Figure 2-3, the effect of the constraints in Figure 2-4. The constraints are:

1. The detectable time delays. The maximum expected time delay occurs under vertical incidence with the maximum expected streamer depth. The minimum expected time delay depends on the signal bandwidth. If the first notch lies outside of the signal bandwidth ($f_{\text{notch}1} > f_{\max}$, with $t_{\Delta} = \frac{1}{f_{\text{notch}1}}$), the inverse ghost will not significantly boost any frequency, thus no significant ringing will occur. Therefore, the ‘energy’ in the signal is small no matter whether the ghost model is correct or not.

Figure 2-3 demonstrates how with $f_{\max} = 100$ Hz, there is minimum energy below $\frac{1}{100} = 0.01$ s – we would limit the ghost delay search to above 10 ms to protect against picking these minimum energy solutions.

2. The upgoing signal must have significantly less energy than the original data. If no peak can be found, the deghosting is simply not carried out.
3. If an integer multiple ($n = 2, 3, 4, \dots$) of the tested delay time also has minimal energy, we disregard this tested delay. Recall that the ghost notches occur with period of $f_{\text{notch}} = \frac{1}{t_{\Delta}}$. Then if $t_{\text{test}} = \frac{t_{\Delta}}{n}$, the ghost notches are filled with period n (e.g. the 3rd, 6th, ..., ghost notches) rather than all ghost notches. This action amplifies *some* of the correct frequencies, and leaves the other notches untouched. The result thus has no ringing, and can be of minimum energy too.

Figure 2-3 demonstrates how the first integer fraction (12.5 ms) of the true ghost delay (25 ms) has minimal ringing, showing up as a peak in the cost function – the constraint makes sure we disregard these integer fractions as potential ghost delays.

By Parseval’s theorem (equality of the L2 norm in the time and frequency domain), we can do a line-search of potential t_{Δ} candidates *and* apply the selected deghosting model rapidly in the frequency domain. The line-search may thus be performed on a trace-by-trace basis in the f - x - y , f - k or f - p domain. We established the time delay in different domains: $t_{\Delta} = \{\frac{2h}{v_p} \cos \theta\}_{fx,y} = \{\frac{2h}{f} k_z\}_{fk} = \{2hp_z\}_{fp}$. The f - x domain may have events crossing at multiple $\cos \theta$, thus may have several time delays in a window and is not a favourable domain to do the adaptive deghosting. Similarly, the f - k domain has a time delay that depends on the frequency, putting multiple time delays on the same k trace. We can see that the f - p domain is the only domain that ideally presents a single delay per p trace. The adaptive deghosting is thus favourably done in the f - p domain.

However, coarse crossline sampling prohibits the PWD of the wavefield into the f - p_x - p_y domain, and we can only do an unaliased f - p_x PWD. This means that events with separate crossline components (p_y), thus with separate time delays, are not separated in the f - p_x domain. To approximate the separation of events, overlapping tapered time-windows are used, small enough to capture single events, yet big enough to capture enough frequency information to estimate the ghost delay. This approach may be faulty for strongly varying events, such as the first reflection which may have a rapidly varying 3D incidence angle, but maps predominantly into a single p_x point.

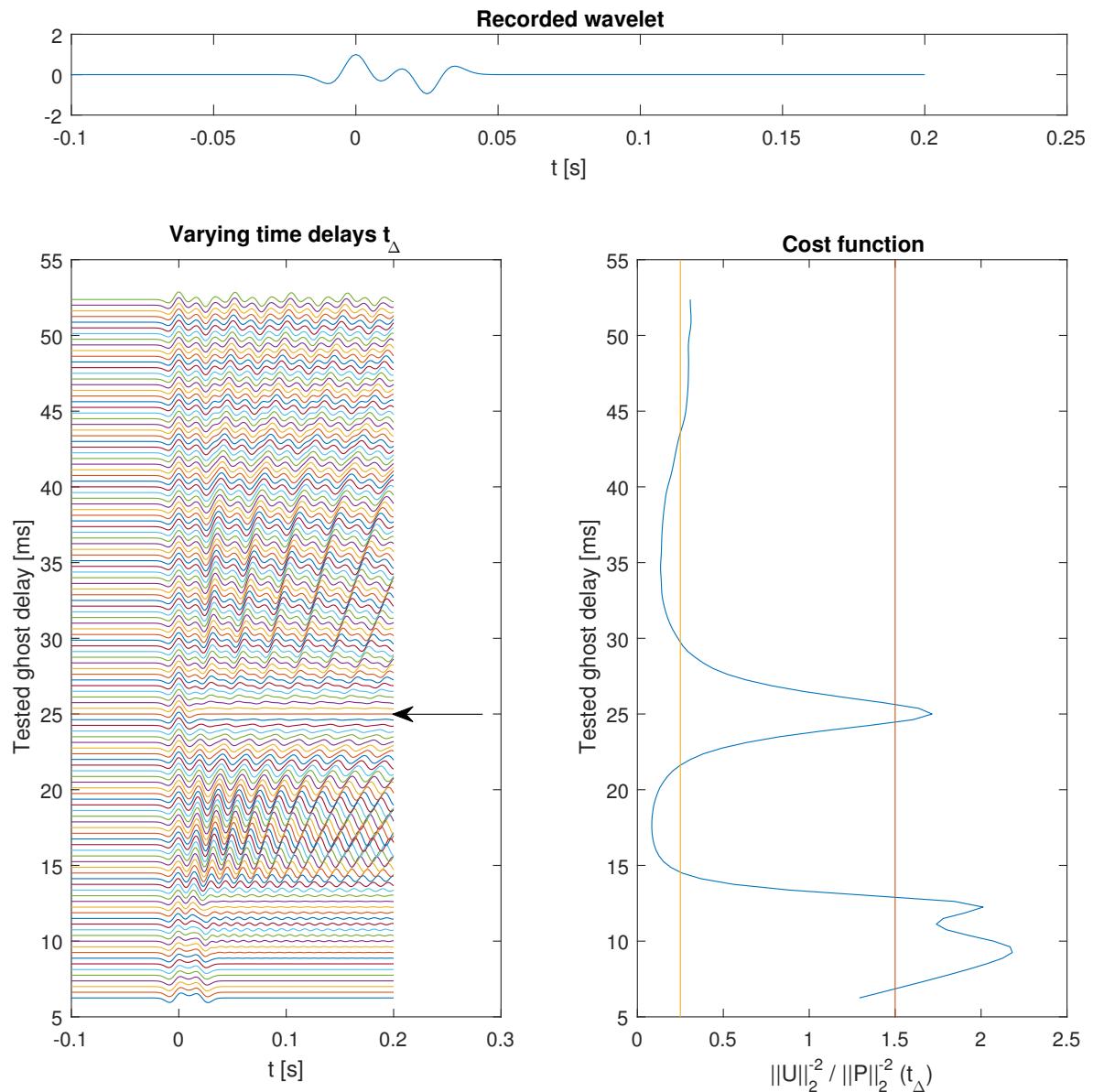


Figure 2-3: An illustration of the unconstrained cost function $\|U(t_\Delta)\|_2^{-2}$ from the energy minimization technique, normalized by the energy in the ghosted data. The maximum peak in this case lies at 9 ms, whereas the desired solution lies at 25 ms. The constraints (Figure 2-4) help to pick the correct maximum.

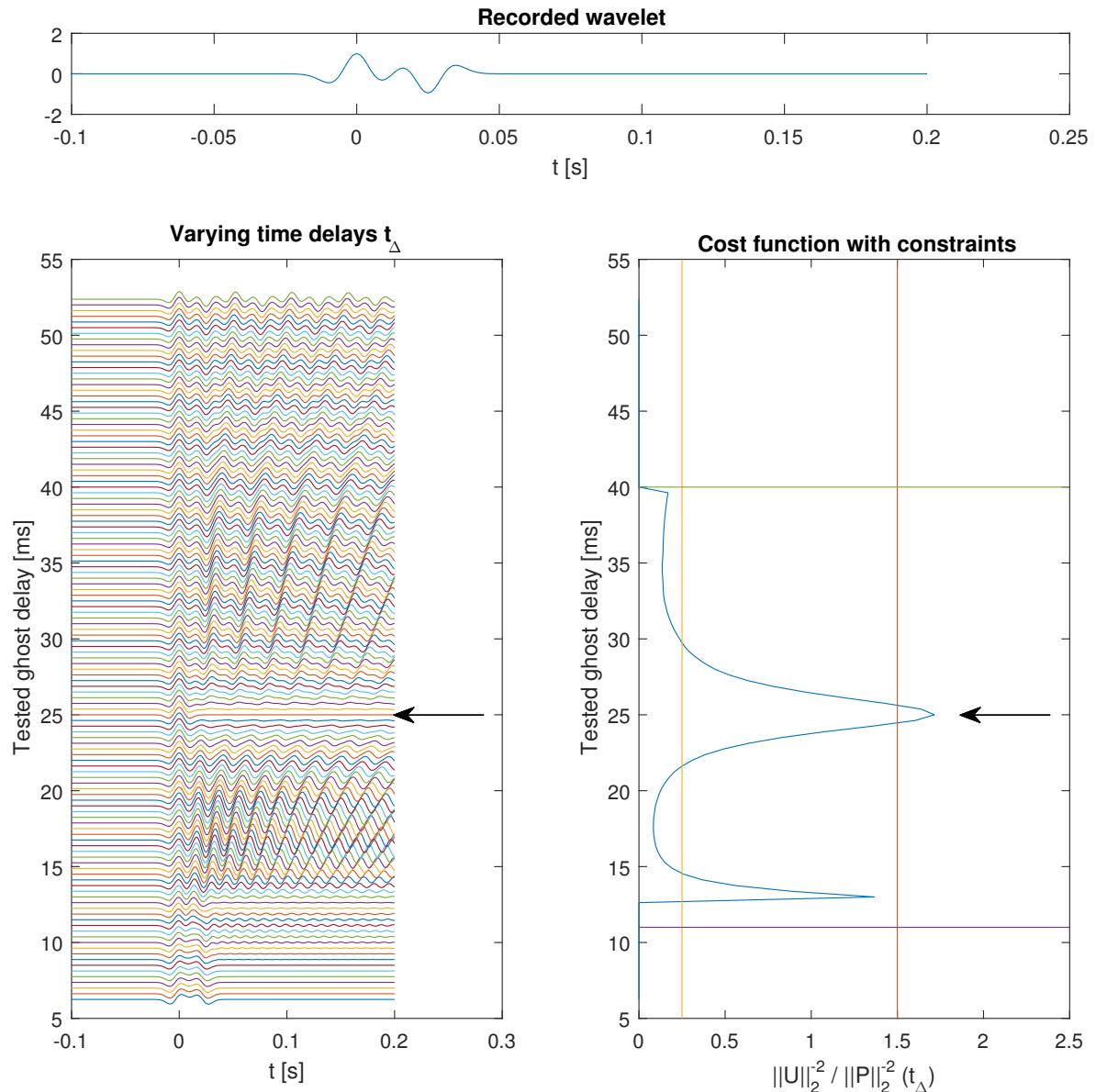


Figure 2-4: An illustration of the constrained cost function (Equation (2-3)) of the energy minimization technique, normalized by the energy in the data. The minimum time delay is limited to the inverse of the maximum frequency $t_{\Delta,\min} = \frac{1}{f_{\max}}$ and limited by a maximum expected time delay. Then, the cost function at > 1.5 (beyond the orange line) is tested against integer multiple energy at value $> \frac{1}{4}$ (beyond the yellow line). The result is a single remaining peak at the true time delay.

2-2 Multisensor deghosting

For the multisensor techniques, we will make a separation of techniques which are truly data-independent (which are simply applying a function to the data), and techniques which are data-dependent (which adapts the applied functions to optimally fit the data). This is different still from data-driven, which design their complete filters based on the data itself, such as the methods in the previous section. The two data-driven models that are proposed in the next two chapters build upon the methods described in this section.

2-2-1 Data-independent – Single streamer

PZSUM

Wapenaar & Berkhout (1989, p. 88) assume a horizontally homogeneous subspace, and tell us that we can transform between $\tilde{P}(f, k_x, k_y, z)$ and $\tilde{V}_z(f, k_x, k_y, z)$ in the following way:

$$\frac{\partial}{\partial z} \begin{pmatrix} \tilde{P} \\ \tilde{V}_z \end{pmatrix} = \begin{pmatrix} 0 & -i\omega\rho \\ \frac{k_z^2}{i\omega\rho} & 0 \end{pmatrix} \begin{pmatrix} \tilde{P} \\ \tilde{V}_z \end{pmatrix} \quad (2-4)$$

We may do eigenvalue decomposition of the transform matrix as such:

$$\begin{pmatrix} 0 & -i\omega\rho \\ \frac{k_z^2}{i\omega\rho} & 0 \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 \\ \frac{k_z}{\omega\rho} & -\frac{k_z}{\omega\rho} \end{pmatrix}}_{\tilde{\mathbf{L}}} \underbrace{\begin{pmatrix} -ik_z & 0 \\ 0 & ik_z \end{pmatrix}}_{\tilde{\mathbf{A}}} \underbrace{\frac{1}{2} \begin{pmatrix} 1 & \frac{\omega\rho}{k_z} \\ 1 & \frac{-\omega\rho}{k_z} \end{pmatrix}}_{\tilde{\mathbf{L}}^{-1}}. \quad (2-5)$$

We interpret the last operator $\tilde{\mathbf{L}}^{-1}$ as the two-way wavefield decomposition, with

$$\frac{1}{2} \begin{pmatrix} 1 & \frac{\omega\rho}{k_z} \\ 1 & \frac{-\omega\rho}{k_z} \end{pmatrix} \begin{pmatrix} \tilde{P} \\ \tilde{V}_z \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \left(\tilde{P} + \frac{\omega\rho}{k_z} \tilde{V}_z \right) \\ \frac{1}{2} \left(\tilde{P} - \frac{\omega\rho}{k_z} \tilde{V}_z \right) \end{pmatrix} = \begin{pmatrix} \tilde{P}_{\text{up}} \\ \tilde{P}_{\text{down}} \end{pmatrix}. \quad (2-6)$$

The *upgoing* wavefield is fully retrieved when summing the two filtered spectra. This is generally known as the ‘PZSUM’ (Caprioli et al., 2012), i.e. summing the pressure P and scaled vertical velocity $\rho v_p V_z = Z$, which can be done in any linear domain (Figure 2-5). Normal choices in seismic processing are:

$$P_{\text{up}}(x, t) = \frac{1}{2} \left(P + \frac{v_p \rho}{\cos \theta} V_z \right), \quad (2-7)$$

$$\bar{P}_{\text{up}}(\tau, p) = \frac{1}{2} \left(\bar{P} + \frac{\rho}{p_z} \bar{V}_z \right), \quad (2-8)$$

$$\tilde{P}_{\text{up}}(f, k) = \frac{1}{2} \left(\tilde{P} + \frac{\omega \rho}{k_z} \tilde{V}_z \right). \quad (2-9)$$

The excellent property of the PZSUM is that it makes zero assumptions on the ghost model. Everything that happens (variations in sea surface roughness, water velocity, reflection coefficient, etc.) to the P component happens to the V_z component as well, it thus does not rely on a ghost model or ghost delay time t_Δ at all.

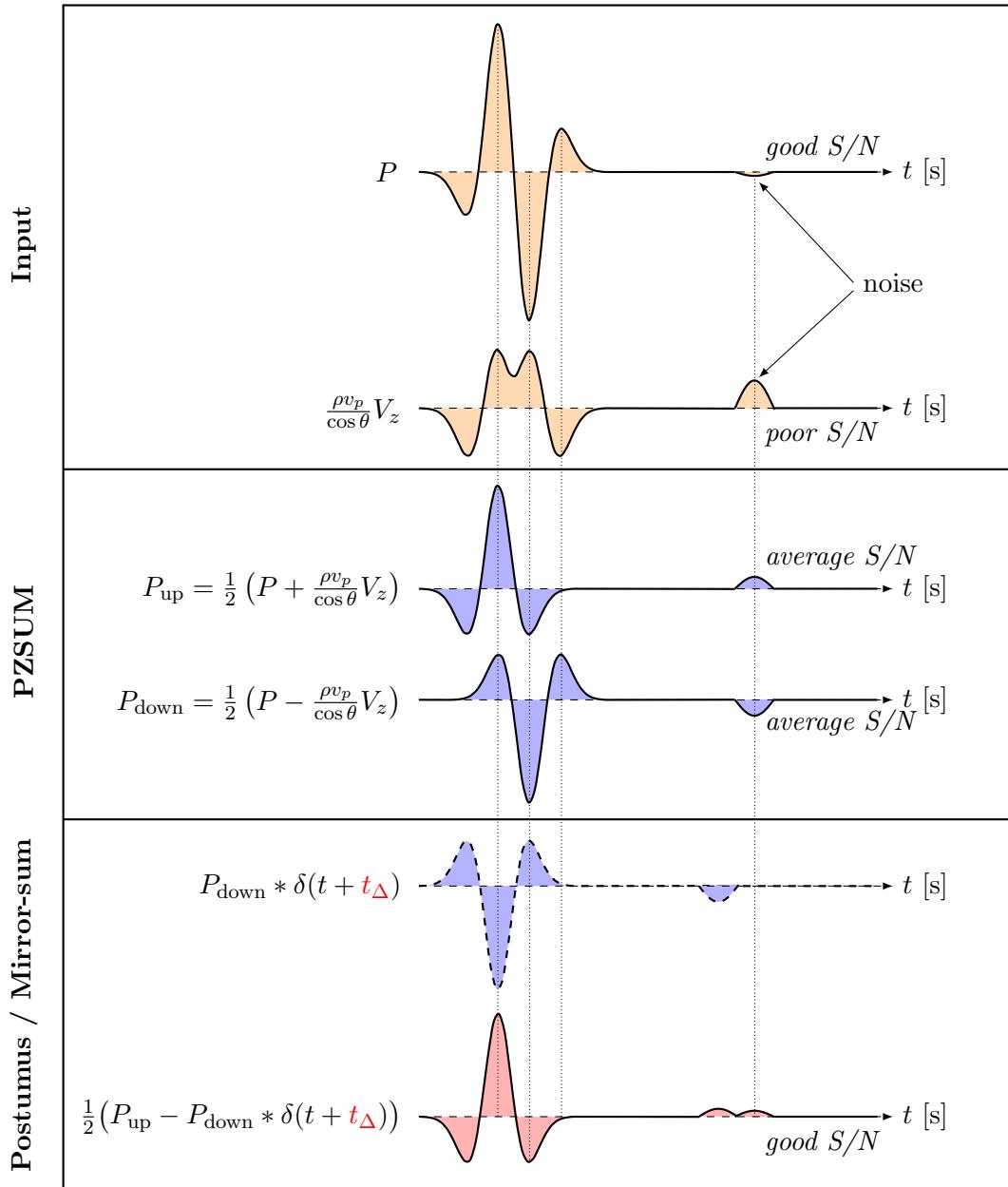


Figure 2-5: Example of the PZSUM (center) and Mirror-sum (bottom) principle. The PZSUM is a weighted sum of the pressure and velocity data, such that the upgoing wave is kept but the downgoing wave is cancelled. The mirror-sum is the sum between the upgoing PZSUM and time-advanced and inverted version of the downgoing PZSUM. Note how the PZSUM does not rely on the ghost model (t_{Δ}), whereas the mirror-sum does. Take note of the difference in S/N on the right side of the image. The PZSUM achieves an average level of noise, while the mirror-sum stacks coherent signal and reduces noise, at the expense of relying on a ghost model.

The difficulties of the PZSUM are threefold. First, the scalar in front of V_z . Without the 3D incidence angle θ one cannot achieve the proper scaling to cancel the ghost. The incidence angle is usually only known in the inline direction, through an inline PWD. The PZSUM then only compensates for the inline incidence angle. The scaling is too small in case of crossline energy propagation, leaving a residual ghost.

Second, the expressions have a pole for $\theta = 90^\circ$. Although it seems reasonable to simply not carry out the scaling factor for this angle, such a simple filter leaves artefacts in the final image. To better deal with the poles, one can again opt for the regularization or complex frequency techniques mentioned in Chapter 1. Another option is to design optimal filters through quadratic (i.e. constrained) programming, for example Røsten et al. (2002).

The third and final drawback of the PZSUM is that the vertical particle velocity usually has a worse S/N than the pressure data at frequencies below about 20 Hz (Caprioli et al., 2012). Kamil et al. (2014) introduce an approach to enhance the PZSUM result with the use of a prior estimate of the noise covariance. Another approach could be to apply the PZSUM only for frequencies above 20 Hz, and use a different deghosting method for the lower frequencies, as in Caprioli et al. (2012).

Posthumus & Optimal Deghosting (ODG)

Analogous to the train of thought in single component deghosting, we may propose that both our P and V_z recordings were caused by the same upgoing wavefield, convolved with their ghost models.

$$\begin{pmatrix} \tilde{P}_{\text{data}} \\ \tilde{V}_{z\text{data}} \end{pmatrix} = \begin{pmatrix} \tilde{G}_p \\ \tilde{G}_z \end{pmatrix} \tilde{P}_{\text{up}}. \quad (2-10)$$

with

$$\begin{aligned} \tilde{G}_p &= 1 - |r_0| e^{-2\pi i 2hk_z}, \\ \tilde{G}_z &= \left(\frac{k_z}{\omega\rho} \right) \left(1 + |r_0| e^{-2\pi i 2hk_z} \right). \end{aligned} \quad (2-11)$$

The system is solved by premultiplying the system with the conjugate transpose of the ghost vector (this solution first appears in Posthumus, 1993, it is the least squares solution to the noise-free ghost problem, also derived in Appendix A-4-1):

$$\tilde{G}_p^* \tilde{P}_{\text{data}} + \tilde{G}_z^* \tilde{V}_{z\text{data}} = \left(\tilde{G}_p^* \tilde{G}_p + \tilde{G}_z^* \tilde{G}_z \right) \tilde{P}_{\text{up}}, \quad (2-12)$$

$$\tilde{P}_{\text{up}} = \frac{\tilde{G}_p^* \tilde{P}_{\text{data}} + \tilde{G}_z^* \tilde{V}_{z\text{data}}}{|\tilde{G}_p|^2 + |\tilde{G}_z|^2}. \quad (2-13)$$

Note that these are element-wise multiplications, performed per frequency-wavenumber combination – not matrix-matrix multiplications. Applying the conjugate ghost models corresponds to dephasing the ghost and summing it together with the primary, illustrated in Figure 2-5 as the ‘mirror-sum’, and derived in Section A-3 (p. 85).

The method can be extended upon by incorporating a noise model. For example, Optimal Deghosting (ODG, Özdemir & Özbe^k, 2010; Caprioli et al., 2012) can deal with the noise by solving the more involved system:

$$\begin{pmatrix} \tilde{P} \\ \tilde{V}_z \end{pmatrix} = \begin{pmatrix} \tilde{G}_p \\ \tilde{G}_z \end{pmatrix} \tilde{P}_{\text{up}} + \begin{pmatrix} \tilde{N}_P \\ \tilde{N}_Z \end{pmatrix}. \quad (2-14)$$

The general solution is derived in Appendix A-4-2. In case of uncorrelated (Gaussian) noise, it can be shown the system is solved in a least squares sense by:

$$\tilde{P}_{\text{up}} = \frac{\frac{\tilde{G}_p^* \tilde{P}_{\text{data}}}{\sigma_p^2} + \frac{\tilde{G}_z^* \tilde{V}_{z,\text{data}}}{\sigma_z^2}}{\frac{|\tilde{G}_p|^2}{\sigma_p^2} + \frac{|\tilde{G}_z|^2}{\sigma_z^2}} = W_1 \tilde{P}_{\text{data}} + W_2 \tilde{V}_{z,\text{data}}. \quad (2-15)$$

where σ_p^2 and σ_z^2 represent the variance of the normally distributed noise on the pressure and the vertical particle velocity. The second formulation makes clear that the weighting W_1 and W_2 puts more emphasis on the pressure and velocity data, depending on the S/N characteristics and ghost model at that frequency.

Kamil & Caprioli (2014) show that the noise model may be replaced by the covariance of the measurement data, as also derived in Appendix A-4-3. In case of uncorrelated noise, the system to solve now becomes the ‘robust ODG’ (RODG):

$$P_{\text{up}} = \frac{\frac{\hat{G}_p^* \hat{P}_{\text{data}}}{|\hat{P}_{\text{data}}|^2} + \frac{\hat{G}_z^* \hat{V}_{z,\text{data}}}{|\hat{V}_{z,\text{data}}|^2}}{\frac{|\hat{G}_p|^2}{|\hat{P}_{\text{data}}|^2} + \frac{|\hat{G}_z|^2}{|\hat{V}_{z,\text{data}}|^2}}. \quad (2-16)$$

Notice that all these formulations (Posthumus, ODG, RODG) depend on getting the ghost time delay right. Using the wrong time delay will result in an incorrect upgoing wave, and may produce ‘ringing’ – similar to that found in single sensor deghosting. This fundamentally differs from the PZSUM, which merely depends on getting the angle of incidence right. At the cost of depending explicitly on the ghost model, the ODG method gains control over the S/N characteristics of the signal.

2-2-2 Data-dependent – Multistreamer

Data-dependent here refers to the fact that part of the ghost model – the 3D incidence angle – is obtained by finding the optimal ghost incidence angle from the data. It is still considered a model based approach because partial knowledge of the ghost model is assumed: the receiver depth, reflection coefficient and acoustic wave speed.

Generalized Matching Pursuit (GMP)

We established before that the large crossline streamer separation prohibits a three dimensional plane wave decomposition as the data would be severely aliased in the crossline direction. However, recent ‘greedy’ algorithms have been able to overcome this problem using the

multisensor redundancy of information. The greedy algorithm used is the *matching pursuit* method, which solves a linear system based on picking the best fitting signal in a set of candidate signals. “*A matching pursuit isolates the signal structures that are coherent with respect to a given dictionary*” (Mallat & Zhang, 1993). Multichannel Interpolation with Matching Pursuit (MIMAP) exploits the fact that the crossline particle acceleration is the gradient of the pressure wavefield in the crossline direction (Vassallo et al., 2010). The data data-adaptive technique uses the two coupled measurements (P_{data} and $V_{y,\text{data}}$) to overcome crossline aliasing at the Nyquist frequency. The method then allows an interpolation or reconstruction of the wavefield between the streamers, by evaluating the achieved PWD of the data.

In the marine seismic case, we already established that the measurements of P , V_z and V_y can all be considered to come from the same upgoing wavefield (Figure 1-9). We can apply MIMAP to this data to achieve interpolation of the data between the streamers, but we may go one step further. Choosing a receiver depth h , reflection coefficient $|r_0|$, acoustic wave speed v_p , we may jointly achieve the interpolation and deghosting which takes into account the full 3D propagation of the wavefield. The system of equations is solved for a chosen dictionary length n of crossline wavenumbers k_y at streamer locations \vec{y} :

$$\begin{pmatrix} \tilde{P}_{\text{data}} \\ \tilde{V}_{y,\text{data}} \\ \tilde{V}_{z,\text{data}} \end{pmatrix} = \begin{pmatrix} G_{p,1}e^{ik_{y,1}\vec{y}} & \dots & G_{p,n}e^{ik_{y,n}\vec{y}} \\ G_{y,1}e^{ik_{y,1}\vec{y}} & \dots & G_{y,n}e^{ik_{y,n}\vec{y}} \\ G_{p,1}e^{ik_{z,1}\vec{y}} & \dots & G_{p,n}e^{ik_{z,n}\vec{y}} \end{pmatrix} \begin{pmatrix} A_1 \\ \vdots \\ A_n \end{pmatrix}. \quad (2-17)$$

Note that each entry in the data vector on the left hand side, as well as the matrix with ghost models, represents a vector with the data for each streamer position \vec{y} (Özbek et al., 2010). Generalized Matching Pursuit (GMP) then achieves joint interpolation and deghosting by finding the optimal entries A_j (a magnitude and phase for the corresponding wavenumbers $k_{y,j}$) which minimize the error between the data and model. In an iterative fashion, the vector of A is expanded, until the ghosted data set is sufficiently described. The PWD in the total vector \vec{A} then fully describes the upgoing wavefield.

Conceptively, the GMP models many monochromatic plane waves of varying crossline wavenumbers, computes what their ghosted measurements would look like along all the streamers, and compare these trial solutions to the actually measured data. If a good match is found between a potential crossline wavenumber and the multisensor measurements along all the streamers, the event may be subtracted from the data, and the matching pursuit tries to find the best event to explain the residual data. This method continues iteratively.

GMP is an effective deghosting technique, as it senses the ghost coming from the crossline direction, unlike PZSUM and ODG. This method overcomes the assumption of inline propagation as found in for example the PZSUM and ODG. However, it depends on an explicit choice of the streamer depth, reflection coefficient and acoustic velocity. The GMP method is also computationally expensive. The question is if simpler methods exist to also account for the 3D propagation of energy.

Chapter 3

Data-driven multisensor PZSUM: PYZSUM

3-1 Motivation

3-1-1 Pitfalls of PZSUM

The PZSUM, as carried out on a single streamer, does not account for crossline energy propagation. The method is thus a 2D method. Summing the pressure and scaled vertical particle velocity *should* achieve:

$$P_{\text{up}} = \frac{1}{2} \left(P + \frac{\rho v_p}{\cos \theta} V_z \right), \quad (3-1)$$

but by using only the inline *apparent* incidence angle, one actually achieves:

$$P_{\text{up}} = \frac{1}{2} \left(P + \frac{\rho v_p}{\sqrt{1 - \sin^2 \theta \cos^2 \phi}} V_z \right). \quad (3-2)$$

This error is shown in Figure 3-1. For $\phi = 0^\circ$, the scaling is correct, but for increasing ϕ the scaling is too small. Without further data, this problem can only be overcome by making additional assumptions on the wavefield or the ghost. For example, Equation (A-4) in Appendix A-1 shows how to approximate $\frac{\rho v_p}{\cos \theta} V_z$ from the ghosted pressure data and an estimated ghost model. Estimating the scalar is then possible by for example minimizing the norm between the acquired and estimated V_z data ¹.

3-1-2 3D angle PZSUM

Figure 3-2 illustrates the error resulting from applying a ‘2D’ (i.e., inline compensation only) PZSUM when there is a significant crossline component.

¹scalar = fminbnd(@(scale) norm(scale*Vz_data-Vz_target, 2), 1, 10) minimizes the ℓ^2 norm in MATLAB.

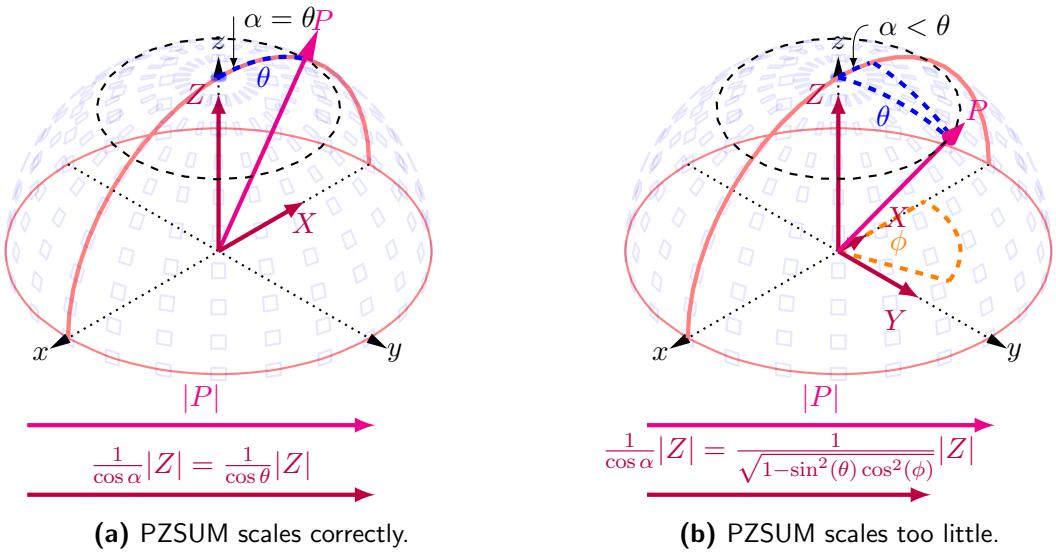


Figure 3-1: Graphical explanation of the scaling applied by the 2D PZSUM, which corrects for the inline incidence angle α rather than the 3D incidence angle θ . The hemispheres are centered upon a recording station, the purple lines showing the amplitudes of the pressure P and particle velocities multiplied with ρv_p : X , Y and Z . This example shows two events incoming at $\theta = 35^\circ$, but with a varying crossline component. To achieve a good summation of both components, we have to scale the Z component by $\frac{1}{\cos \theta}$. **(a)** The PZSUM correctly scales the amplitude of the Z component which propagates fully along the inline. **(b)** The PZSUM does not take into account crossline propagation, thus scales too little. Summing the P and Z component in this second case would not fully remove the ghost. Note that P is not a vectorial property as one may take away from this image.

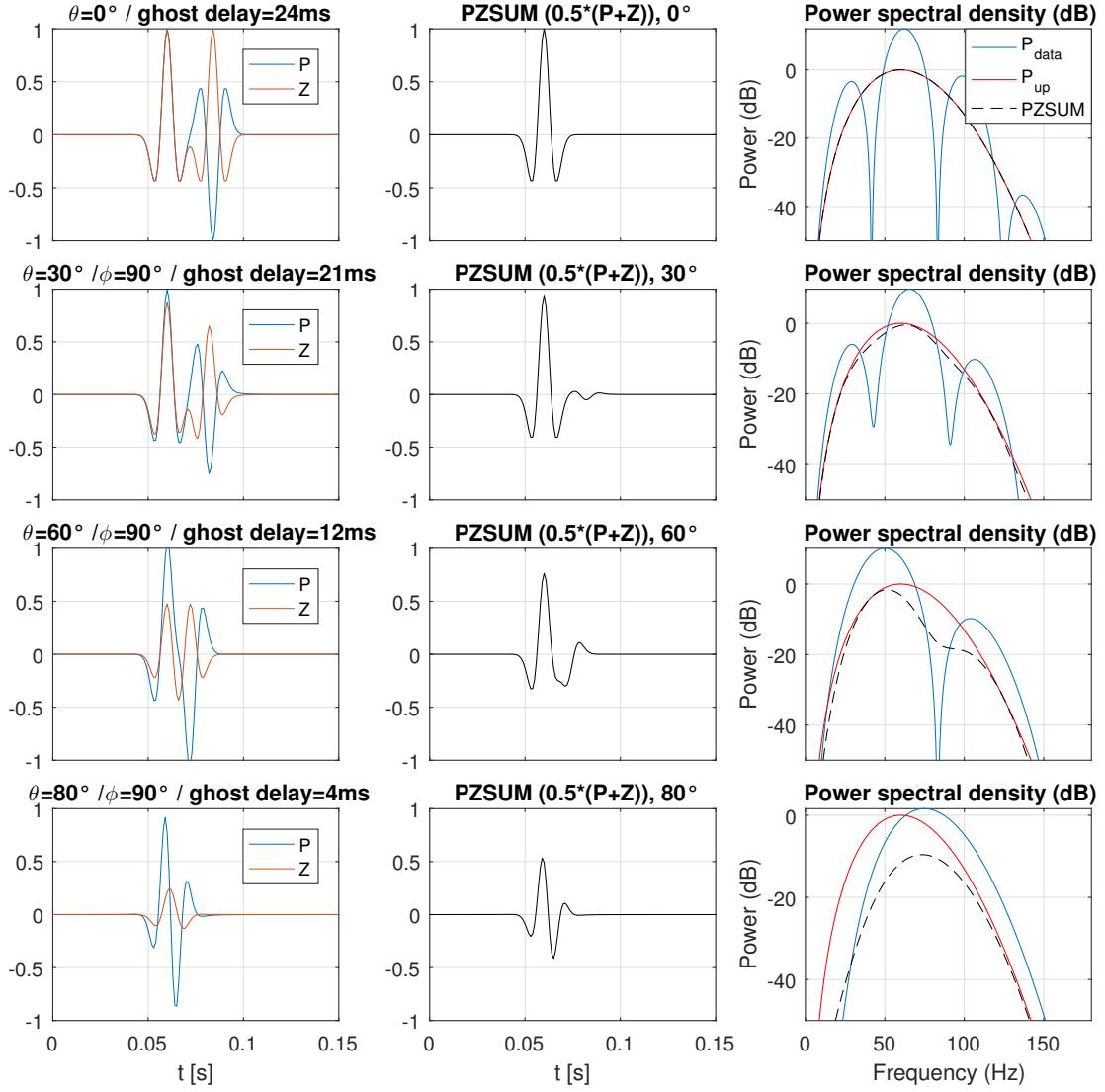


Figure 3-2: Example of the PZSUM failing to correctly deghost a ghosted Ricker wavelet in case of crossline energy propagation. Consider a fixed $p_x = \sin(\theta) \cos(\phi)/v_p = 0$ but varying $p_y = \sin(\theta) \sin(\phi)/v_p$. The time delay of the ghost is given by $t_\Delta = 2hp_z = 2h\sqrt{\frac{1}{v_p^2} - p_y^2}$. Whereas the pressure data remains of fixed magnitude, the vertical velocity data must scaled up as $Z = \rho v_p V_z$, and further scaled by an incidence angle dependent $\frac{1}{\cos(\theta)}$. The 2D PZSUM is unable to take this crossline propagation into account, thus does not properly scale Z before the summation. The top three figures show vertical incidence, which gives the correct target result and spectrum. The three lines below demonstrate the PZSUM at increasing angles of incidence, decreasing in performance and a showing a disturbed spectrum (the target spectrum being the orange line).

3-2 PYZSUM

The PZSUM thus only compensates for the inline incidence angle of the data. We can, however, acquire an estimation of the crossline slowness, using the crossline particle velocity V_y , pressure P , and a technique called the Time Gradient Ratio (TGR), as explained in the next section. The combined knowledge of the inline slowness (p_x) and the crossline slowness (p_y) allows an estimation of the true 3D incidence angle, $v_p p_z = \sqrt{1 - p_x^2 v_p^2 - p_y^2 v_p^2} = \cos \theta$. This opens up the opportunity to do a PZSUM that incorporates crossline energy.

Why not use the TGR method on V_z directly? The PZSUM requires the incidence angle of the upgoing wave – but TGR gives the slowness of the up- *and* the downgoing wave at the streamer. Upon reflection with the sea surface, the wave propagates in the same x and y direction, but will turn from an up- to a downgoing wave, i.e. the p_z component reverses. Therefore, p_x and p_y are the same for both the incident wave and its ghost, whereas p_z differs between the primary and ghost. If these two values interfere, we do not find the correct p_z (as per Equation (3-7)). We thus compute the upgoing vertical slowness as $p_z = \sqrt{\frac{1}{v_p^2} - p_x^2 - p_y^2}$.

3-2-1 Time Gradient Ratio

Consider the one-way linear (i.e. ‘plane’) wave equation for a single event (LeVeque, 1992):

$$\nabla P + \vec{p} \frac{\partial P}{\partial t} = 0. \quad (3-3)$$

Similarly, consider the source-free linearized equation of motion (Wapenaar & Berkhouwt, 1989):

$$\nabla P = -\rho \frac{\partial \vec{V}}{\partial t}. \quad (3-4)$$

Under the assumption of a single plane wave in our data, we may eliminate ∇P from the two equations above, to find the slowness vector:

$$\begin{aligned} -\vec{p} \frac{\partial P}{\partial t} &= -\rho \frac{\partial \vec{V}}{\partial t}, \\ \rightarrow \vec{p} &= \rho \frac{\dot{\vec{V}}}{\dot{P}} \quad . \end{aligned} \quad (3-5)$$

This relation is called the Time Gradient Ratio (TGR, Ravasi et al., 2011; Rentsch-Smith et al., 2013). The equation is valid on a trace-by-trace basis, thus is not inherently limited by any type of aliasing. The V_y measurement thus allows an estimation of the crossline slowness p_y :

$$p_y = \rho \frac{\dot{V}_y}{\dot{P}} \quad \text{if } \dot{P} \neq 0. \quad (3-6)$$

The dots over the variables represent differentiation with respect to time, the ‘if’ statement prevents division by 0.

The single event requirement

If two overlapping events of different p_y are input to the TGR, we would evaluate:

$$p_{y,\text{TGR}} = \rho \frac{\dot{V}_{y,1} + \dot{V}_{y,2}}{\dot{P}_1 + \dot{P}_2}, \quad (3-7)$$

which as we can see does not evaluate to a representation of either p_y .

The TGR requirement for single events in the data is strong, and is not met in a typical shot gather – even the reflection and refraction signal of a horizontally layered subsurface interfere in a shot gather. A more complicated subsurface will furthermore lead to reflections and diffractions coming in from different angles, leading to further overlap in the data. To avoid the crossing of events, [Fortini & Vassallo \(2012\)](#) suggest the use of $f\text{-}k$ filtering to cluster the data into many sections of near-parallel propagation, before the application of the TGR. This ‘fan-filtering’ is not an optimal solution against overlap, because even near-parallel propagating events can cross. The TGR derived from these sections may therefore still be tainted with overlapping events.

One option to overcome this problem would be to use more fan-filters, to achieve an even better approximation of only near-parallel propagating events. We propose a novel different approach: to apply TGR in the $\tau\text{-}p_x$ domain instead. The $\tau\text{-}p_x$ transform retains a time domain axis, and yet achieves a separation of events with different apparent velocities. We may thus transform a shot gather into the $\tau\text{-}p_x$ domain using least squares based or sparse inversion based techniques ([Turner, 1990](#); [Zhou & Greenhalgh, 1994](#); [Wang et al., 2014](#)) and apply the TGR within this domain. A consequence is that we cannot map the obtained p_y values back into the $x\text{-}t$ domain. We must thus also process the data in the $\tau\text{-}p_x$ domain.

Median filtering to improve S/N

The extrema in the pressure and particle velocity have a time derivative of 0 – where we thus cannot evaluate the TGR to find p_y . Additionally, noise in the data poses a problem for the TGR as per Equation (3-7): if the time derivative of the noise is much larger than that of the signal, the p_y estimate of the TGR will not be representative of the signal. In practice, this noise tends to show up as ‘salt and pepper noise’ (i.e. scattered outliers at minima and maxima).

[Ravasi et al. \(2011\)](#) suggests the use of bandpass filtering to limit the noise in the data, and to apply a 1D median filter along the time axis to fill the unevaluated points. The application of a bandpass filter on P and V_y may select the high S/N frequencies of the data, thus may limit the influence of noise in further steps. The median filter along the time axis is able to ‘fill in’ the unevaluated p_y values at the minima and maxima.

As we know that the S/N of the particle velocity is worst at low frequencies, we could follow [Ravasi et al. \(2011\)](#) and use a low-cut filter on P and V_y as input to the TGR. However, the low-cut broadens the wavelet in the form of additional side-lobes. The side-lobes may interfere with events at nearby τ , violating the single event requirement. We thus decided not to use a frequency filter strategy, in order to keep events maximally separated. Instead, we propose the use of a 2D median filter rather than a 1D median filter. Whereas the 1D

median filter replaces every center entry by the median of the values around it along one dimension, the 2D median filter replaces each entry with the median of those surrounding it along two dimensions. In the $\tau-p_x$ case, a 2D median filter replaces its center with the median of a window in the τ and p_x direction. The median filter is a known technique to remove ‘salt and pepper’ noise from images (Ahmed et al., 2015). We furthermore propose the use of a cascaded 2D median filter, known to further improve the S/N ratio (Ahmed et al., 2015).

3-2-2 Implementation

The practical implementation of the method is straightforward. It comprises of the following steps:

1. **Transform P , V_z and V_y to the $\tau-p_x$ domain.** The TGR requires the data, separated into single events, along a time axis. A domain satisfying these requirements is the $\tau-p_x$ domain. This requirement is generally met within the $\tau-p_x$ domain. An exception is for multiple events which have equal inline slowness and intercept time, but differ in crossline slowness – as these would not be separated in the $\tau-p_x$ domain.

The range of p_x considered in the transform influences the quality of the $\tau-p_x$ transform. Extending the slowness range beyond the ‘signal cone’ ($\pm \frac{1}{v_p}$) increases the resolution of the transform and its inverse. This means that the signal is better preserved, and incoherent noise is mostly mapped to slowness beyond the signal cone. Extending the slowness range beyond $\frac{1}{1200}$ s/m, however, degrades the performance again, causing spatial aliasing issues. After several tests, the optimal slowness range on all data sets is found to be $-\frac{1}{1200}$ (s/m) to $+\frac{1}{1200}$ s/m.

The 2D implementation of the inline PZSUM in the $\tau-p_x$ domain takes the form:

$$\bar{P}_{\text{up}}(\tau, p_x) = \frac{1}{2} \left(\bar{P} + \frac{\rho v_p}{\sqrt{1 - v_p^2 p_x^2}} \bar{V}_z \right) = \frac{1}{2} (\bar{P} + W \circ \rho v_p \bar{V}_z), \quad (3-8)$$

Where \circ is the Hadamard (i.e. element-by-element) matrix product. The scalar W is an $(n_\tau \times n_{p_x})$ matrix (i.e. the number of intercept times τ times the number of slowness values). We apply this scalar only within the signal cone:

$$W(\tau, p_x) = \begin{cases} \frac{1}{\sqrt{1 - v_p^2 p_x^2}} + \epsilon & \text{if } |p_x| < \frac{1}{v_p} \\ \frac{1}{1 - v_p^2 p_x^2} + \epsilon & \text{else} \\ 1 & \text{else} \end{cases}. \quad (3-9)$$

This scalar is regularized to prevent amplifying the \bar{V}_z too much – as its worse S/N will overshadow the signal otherwise. The regularization smoothens the scalar to prevent artefacts, similar to the discussion at the PZSUM. An $\epsilon = 0.01$ limits the amplification of \bar{V}_z to a maximum of 5 (corresponding to an incidence angle $\arccos(\frac{1}{5}) = 80^\circ$), shown in Figure 3-3. The complete matrix W , i.e. the complete scalar for the $\tau-p_x$ domain, is shown in Figure 3-4.

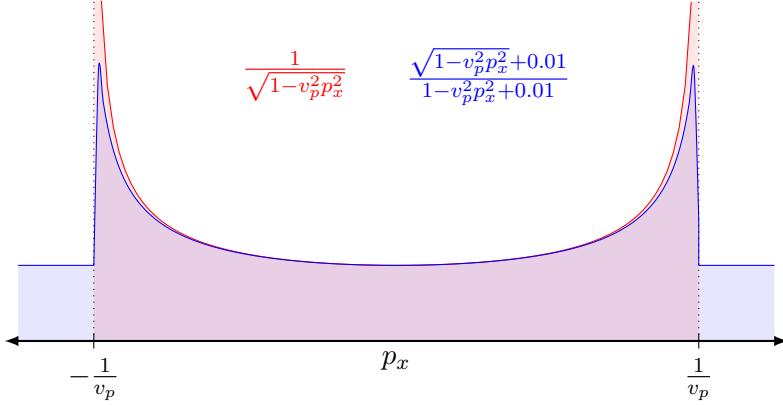


Figure 3-3: The regularized scalar for the PZSUM in the $\tau-p_x$ domain approximates the true scalar, but avoids too much boosting at high angles of incidence. It also preserves signal outside the signal cone.

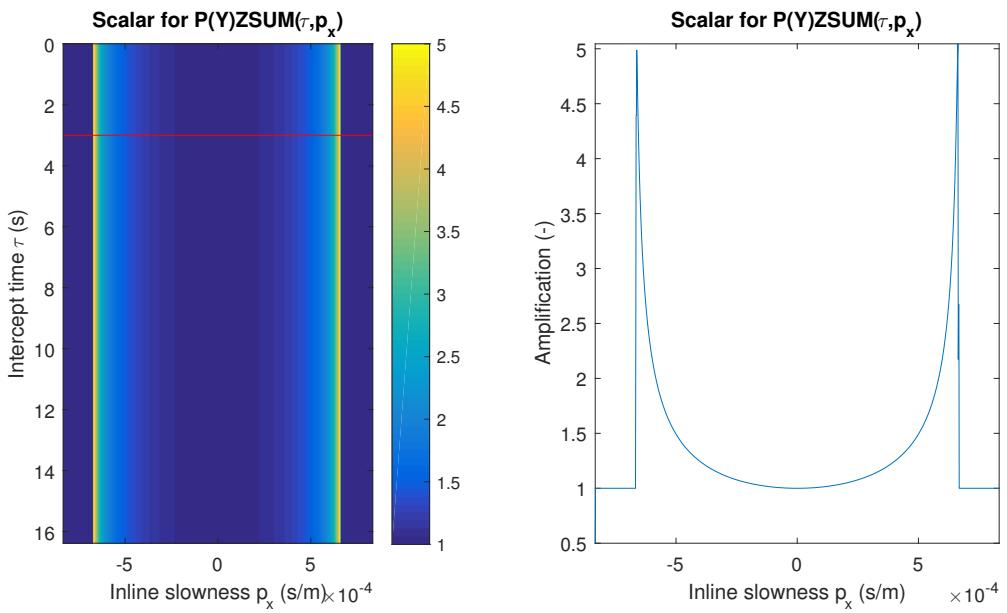


Figure 3-4: The PZSUM scalar as a function of time and slowness (left), and a cross section (red, blue, right) of the scalar. The 2D PZSUM has no time dependency, hence the scalar is constant through time.

2. Compute the TGR.

The TGR involves two time derivatives. This differentiation can be performed in the time domain (finite differences) or frequency domain (multiplication with $2\pi i$). Figure 3-5 shows that the frequency domain differentiation is more stable against noise than finite differences, and thus is the preferred choice.

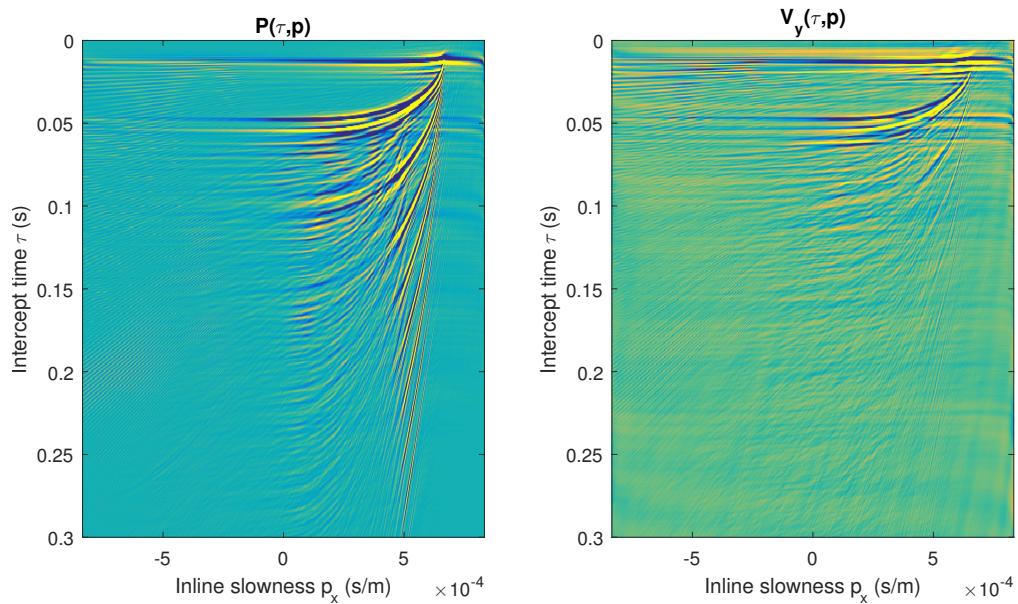
The evaluated TGR step is:

$$p_y(\tau, p_x) = \rho \frac{\dot{\bar{V}}_z}{\dot{\bar{P}}} = \rho \frac{\mathcal{F}^{-1}(2\pi i f \mathcal{F}(\bar{V}_y(\tau, p_x)))}{\mathcal{F}^{-1}(2\pi i f \mathcal{F}(\bar{P}(\tau, p_x)))} \quad \text{if } |\dot{\bar{P}}| > \varepsilon. \quad (3-10)$$

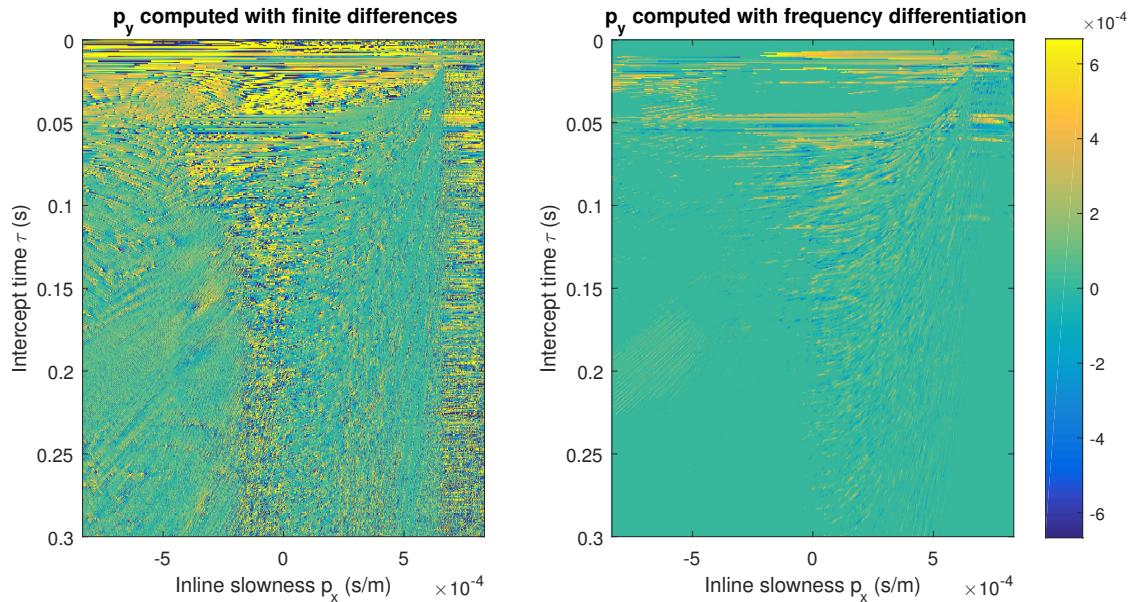
The ‘if’ statement prevents division by zero, i.e. $\varepsilon = 0$. The threshold may also prevent against noise, ε can generally be set to a fraction of the maximum amplitudes in $\dot{\bar{P}}$. We found that setting ε to 1% of the maximum provided excellent results blocking out noise. (Figure 3-5 uses this 1% threshold on both methods.)

The obtained p_y may still be unphysical due to noise (i.e. $|p_y| > \sqrt{\frac{1}{v_p^2} - p_x^2}$). These values are clipped.

The estimated p_y still requires an additional step of filtering. The minima and maxima of the wavelet have 0 as their time derivative, causing an estimation of p_y as 0. To remove such erroneous estimations, coherent events in the data are brought out by smoothing with a median filter. This smoothing also removes ‘salt and pepper noise’, i.e. positive and negative outliers, thus actually *increases* the S/N. In theory, only a 1D median filter over time is required to remove the non-essential zeroes. In practice, it is found that a 2D median filter is better able to bring out coherent effects in real data with noise. The choice for the window size can be critical – too large, and it removes all signal, too small and it doesn’t improve the S/N of the signal. Testing shows that a 5×5 window (that is: two samples in all directions around the sample) achieves this compromise. Following Ahmed et al. (2015), processing the image again with the same median filter (a ‘cascaded median filter’) further improves the S/N, shown in Figure 3-6. Consider the first pass of the median filter as a way to remove ‘salt and pepper’ noise, and the second pass of the median filter a way to smoothen the data before use in the scalar.



(a) Cut out of the P and V_y components of the real data set 1 in the τ - p_x domain.



(b) The two different p_y estimations using finite differences or differentiation in the frequency domain.

Figure 3-5: The TGR methods show different sensitivities to noise. The 5-point finite difference stencil formulation produces a noisy TGR result. The frequency differentiation is less noisy, and visually corresponds to the events found in the input τ - p_x data.

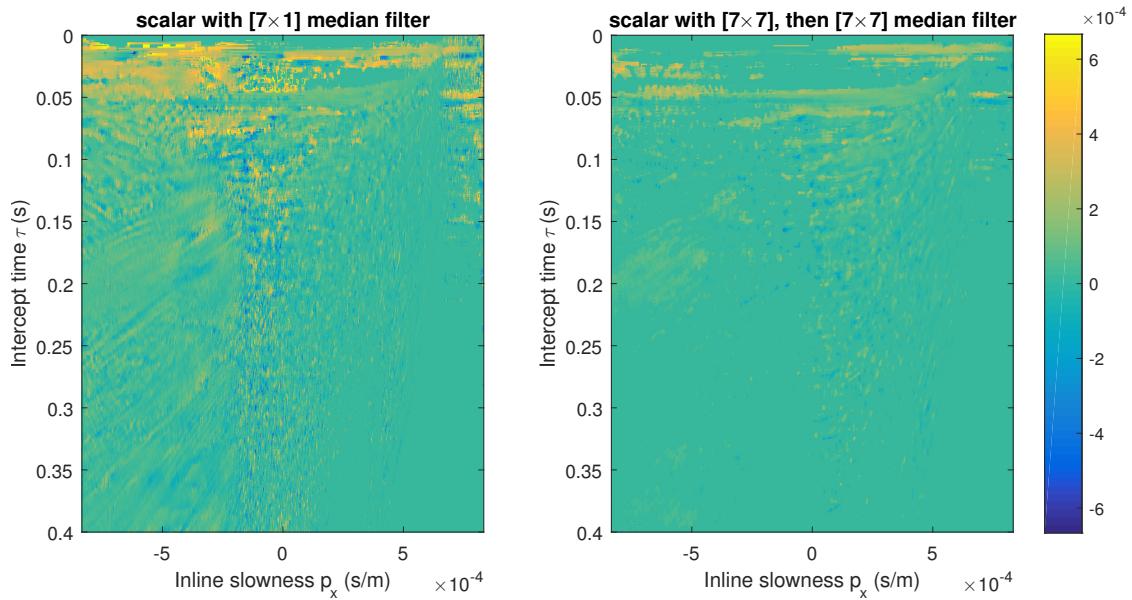


Figure 3-6: Two different median filters are applied to the TGR result. The crossline energy is all directed in just the positive p_y direction (i.e. all should be between green and yellow), but noise has caused patches of negative p_y (blue). The left one is a 1D median filter along the time direction, the right one a 2D median filter. Note that the right signal has a lower resolution, but also contains less outliers.

3. Compute the full $P(Y)ZSUM$ scalar.

We now have the ingredients to build the final scalar W for the PZSUM $W = \frac{1}{v_p p_z} = \frac{1}{\sqrt{1-p_x^2-p_y^2}}$. We then use the value of this scalar where it *differs* from the 2D scalar, Equation (3-9). This way, we retain the smooth filter qualities of the data towards the high p_x edges if the event has no crossline energy, and compensate fully in case of crossline energy. The scalar computed after application of two different median filters is shown in Figure 3-7. The cascaded median filter is smoother, which leads to cleaner PZSUM result.

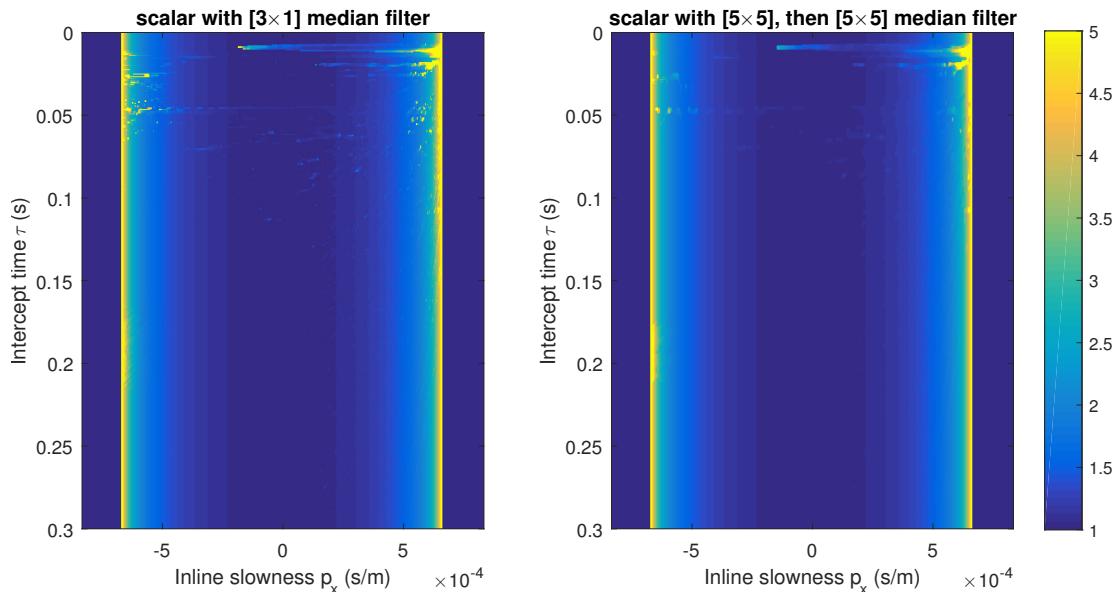


Figure 3-7: The total ‘3D’ scalar for the PYZSUM.

4. The $P(Y)ZSUM$ can now be carried out in the $\tau-p_x$ domain, and brought back to the time domain.

We can evaluate:

$$\bar{P}_{up}(\tau, p_x) = \frac{1}{2} (\bar{P} + W \rho v_p \bar{V}_z). \quad (3-11)$$

And correspondingly, we can return the data to the $x-t$ domain with an inverse $\tau-p_x$ transform.

3-2-3 Illustration on synthetic data

The steps specified in the previous section have been implemented in MATLAB. To illustrate the performance of the technique, we will apply it to a synthetic data set (Figure 3-8). The synthetic data was created using the 3D Green’s function solutions to the inhomogeneous Helmholtz equation $(\nabla^2 + \left(\frac{2\pi f}{v_p}\right)^2)G = -\delta(t)$: $G_{3D} = \frac{\delta(t-r)}{4\pi r}$, where $r = \sqrt{(x_s - x_r)^2 + (y_s - y_r)^2 + (z_s - z_r)^2}$ represents the Euclidian distance between source and receiver. The Green’s functions for the velocity components are found by taking the

derivatives to the respective x , y and z dimension. The solution to the wave equation to a different input is then found through the convolution between G_{3D} and the input signal S , which is efficiently done in the frequency domain. The Fourier transform $\hat{G}_{3D}(f) = \frac{e^{2\pi i f r}}{4\pi r}$ is easily found. For the signal S we use a 40 Hz central frequency Ricker wavelet. The data is simulated by creating synthetic data with $z_r = 50$ m as the receiver depth. The ghosted synthetic data is then produced by virtually positioning the receiver above the sea surface ($z_r = -50$ m), and inverting the polarity for this ghost for the P , V_x and V_y components. The data is produced for inline receivers x_r from -3000 m to 3000 m, at $y_r = 0$ and a receiver spacing of $\Delta x = 5$ m. The data is sampled at $\Delta t = 2$ ms, for a total of 3 seconds.

The method thus allows us to simulate a point-source buried underground at coordinates (x_s, y_s, z_s) . The chosen locations are specified in Table 3-1. The synthetic data contains four experimental cases:

1. Without y component, the results should be equal for both PZSUM and PYZSUM.
2. With a ‘strong’ y component ($\theta = 75^\circ$ at the apex), to test the performance difference where the PZSUM assumes $\theta = 0$ at the apex.
3. With two intertwined upgoing and downgoing waves of different p_y – to test if the scalar can differentiate between closely spaced events.
4. With events of opposite p_y , but shifted along the inline direction. These apex of these events will overlap at $p_x = 0$. The scalar $\frac{1}{v_p p_x}$ should be equal for both waves – but the V_y data will destructively interfere in the $\tau-p_x$ domain, causing the TGR to fail and find a $p_y = 0$ at these inline slowness values.

The true 3D incidence angle at every receiver location is $\theta_{3D} = \arctan(\frac{\sqrt{(y_s-y_r)^2+(x_s-x_r)^2}}{z_s-z_r})$, while the PZSUM angle is a 2D approximation, thus equals $\theta_{2D} = \arctan(\frac{x_s-x_r}{z_s-z_r})$. The difference between PZSUM and PYZSUM is thus naturally largest for large crossline distance.

Table 3-1: The locations of the buried sources, and the test case that they represent.

Case		x_s (m)	y_s (m)	z_s (m)
No y component		0	0	500
Strong y component		0	800	200
Overlapping time	No y component	0	0	1300
	Little y	0	600	1200
Overlapping space	Medium y	0	-1500	1500
	Opposite y	800	1500	1500

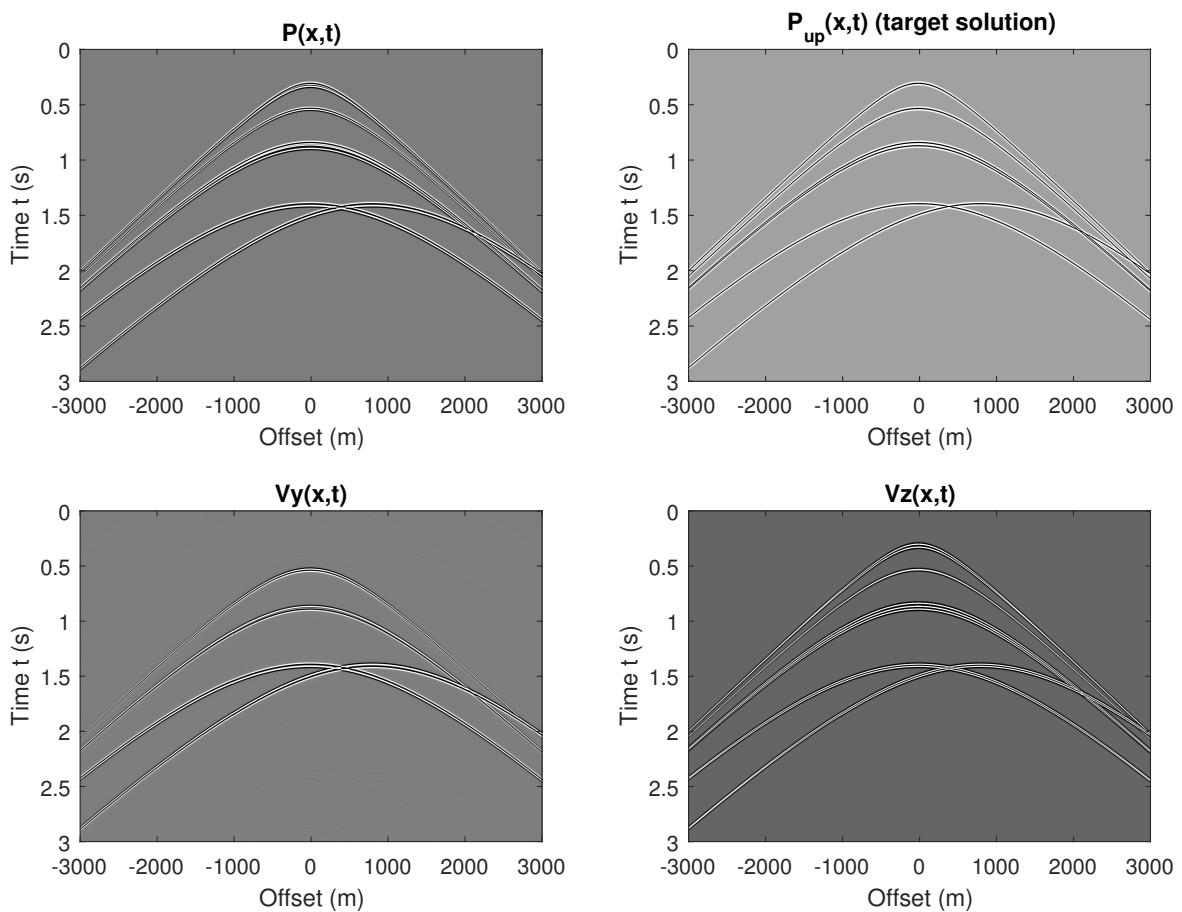


Figure 3-8: The synthetic data created using buried sources at locations specified in Table 3-1.

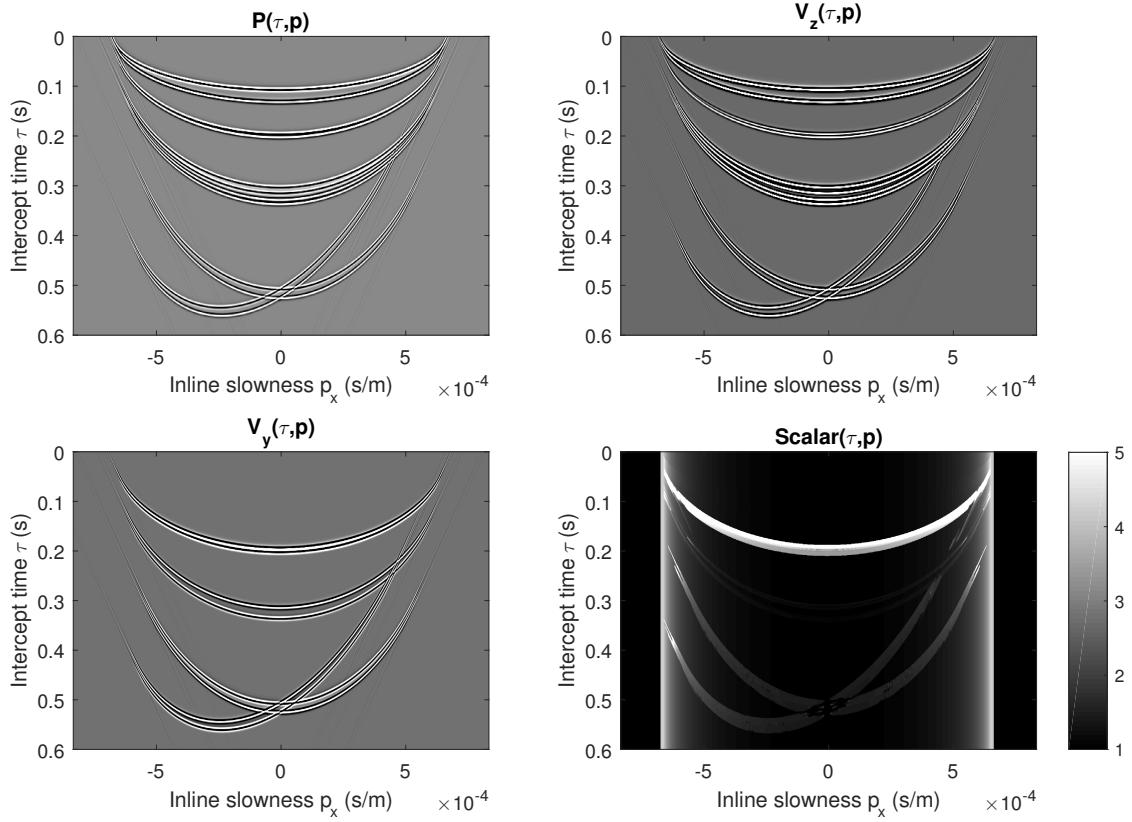


Figure 3-9: The τ - p_x transform of the synthetic data created using buried sources at locations specified in Table 3-1, and the PYZSUM scalar computed from this data. Note that the scalar varies smoothly, except on those points where the τ - p_x transform shows two overlapping events.

Quality check of the deghosting results

We will use three main methods to quality check (QC) the data: visually, with the use of the frequency spectrum, and using the autocorrelation function.

The first method we will use is a visual check of the shot gather: we may visually establish whether the deghosting has been able to sharpen the image, and whether artefacts show up. In the case of synthetic data, we may look at individual traces and compare the found trace to the true upgoing wavefield.

The second method is a look at the spectra, as we assume the deghosting procedure removes the ghost notches from the data. In the synthetic case we may again judge the performance with respect to the known solution. In real data, we may observe which method visually seems to fill the notch. Notches in the spectra of the input data may result from the destructive interference of different events rather than be caused by the ghost.

The third and last method, is to look at the autocorrelation function (ACF) of the data. The ACF is defined as

$$r(t) = \int_{-\infty}^{\infty} f(\tau)f(\tau+t) d\tau. \quad (3-12)$$

If the data f is an infinite and random signal, as the earth response may be approximated, the ACF will have a spike at $t = 0$, and return 0 elsewhere – the signal could be considered incoherent if it is advanced in time. The presence of the ghost at fixed t_Δ will however show up in the ACF at $r(t_\Delta)$, as every recorded event appears twice, at a lag of t_Δ . We may thus expect that the deghosting operation will remove this lag. Because this result is only valid along traces with equal ghost delay, we preferedly apply the ACF along single events. In the synthetic data set it is possible to provide almost complete separation of events (except where they overlap), thus we may employ the ACF in the offset-time domain. For real data, the interference between events with different ghost delays prohibits a detailed study of the ACF in the offset-time domain. However, by applying a normal moveout (NMO) correction to the data, we can turn the data into zero-offset data when using the correct velocity model. In the real data sets, we may NMO correct the data for the strong sea bottom reflection, using the acoustic wave velocity in water. Because the sea-bottom reflection is much stronger than surrounding events, it dominates the ACF result, and will provide another check to see if the ghost energy has been removed.

QC of the synthetic results

We can now compare the result of the PZSUM and the PYZSUM. First visually, as in Figures 3-10 and 3-11. The difference between the two, is that one applies the PYZSUM for the entire section (hence, has a wrong scalar where events in the $\tau-p_x$ domain overlap), and a PYZSUM which is applied through 50% overlapping tapered, moving windows. We use this second approach (overlapping windows) to look closely at the zero-offset traces in Figure 3-12. We can come back to our four cases: the PYZSUM is in all cases at least as good as the PZSUM for restoring the upgoing wavefield.

1. No y component: the two methods produce the same results.

2. Strong y component: The PYZSUM deghosts the event (the wavelet looks like the one without y component above), whereas the PZSUM has a ghost left visible as the black shadow beneath the wavelet. Looking at a single trace (Figure 3-12) we notice the upgoing wave was not fully restored. This is due to the median filter smoothing of the scalar, switching between no amplification (1) and the amplification required (5).
3. The intertwined events: The PYZSUM again outperforms the PZSUM, by removing the ghost.
4. The interfering events: As was predicted, and is visible in the scalar (Figure 3-9), there is no amplification for the interfering events near $p_x = 0$, thus these events are not deghosted. Note, additionally the ‘bow-tie’ artefacts around the apex. These artefacts arise from the strong variation in the scalar near these slowness values.

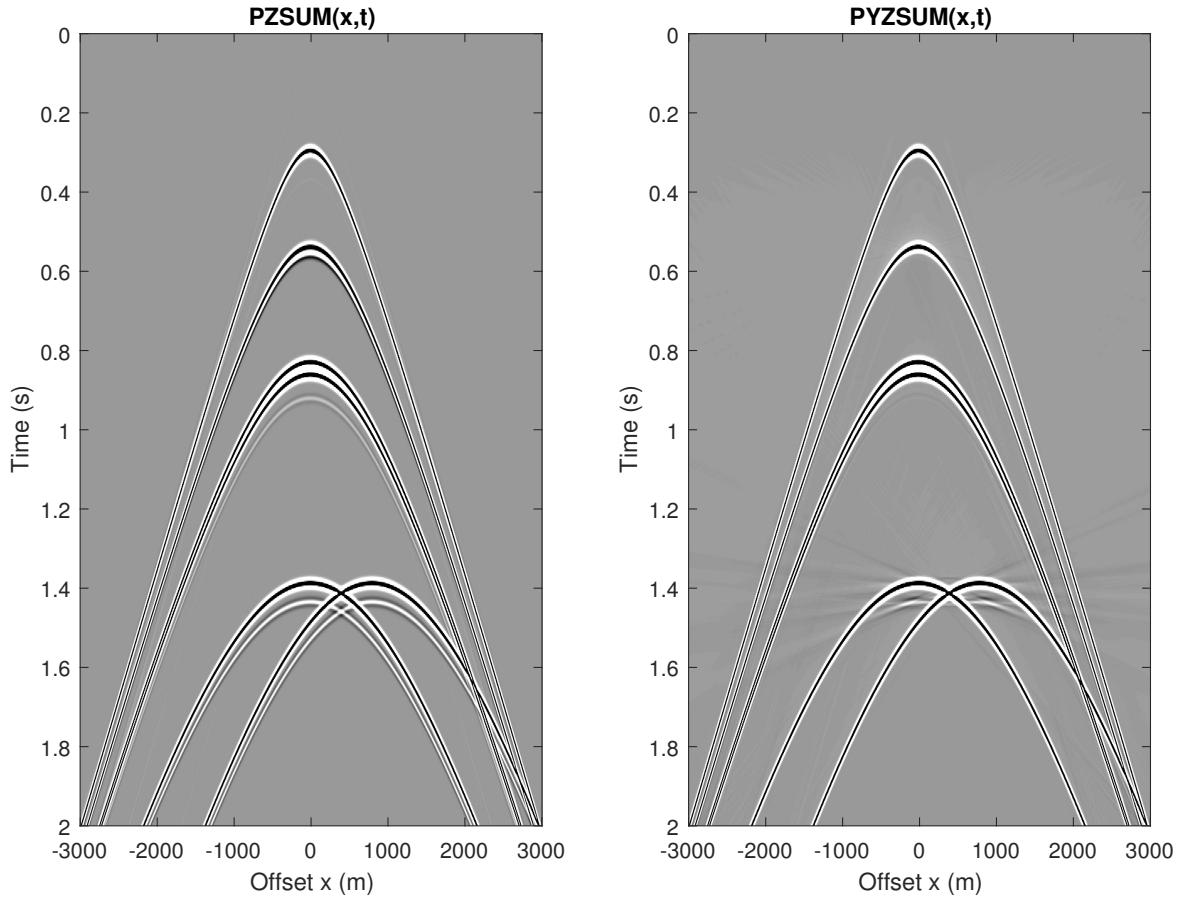


Figure 3-10: Comparison between the PZSUM and the PYZSUM applied on the entire synthetic data set. A time gain of t has been applied to the traces, the colorscale is equal for both images.

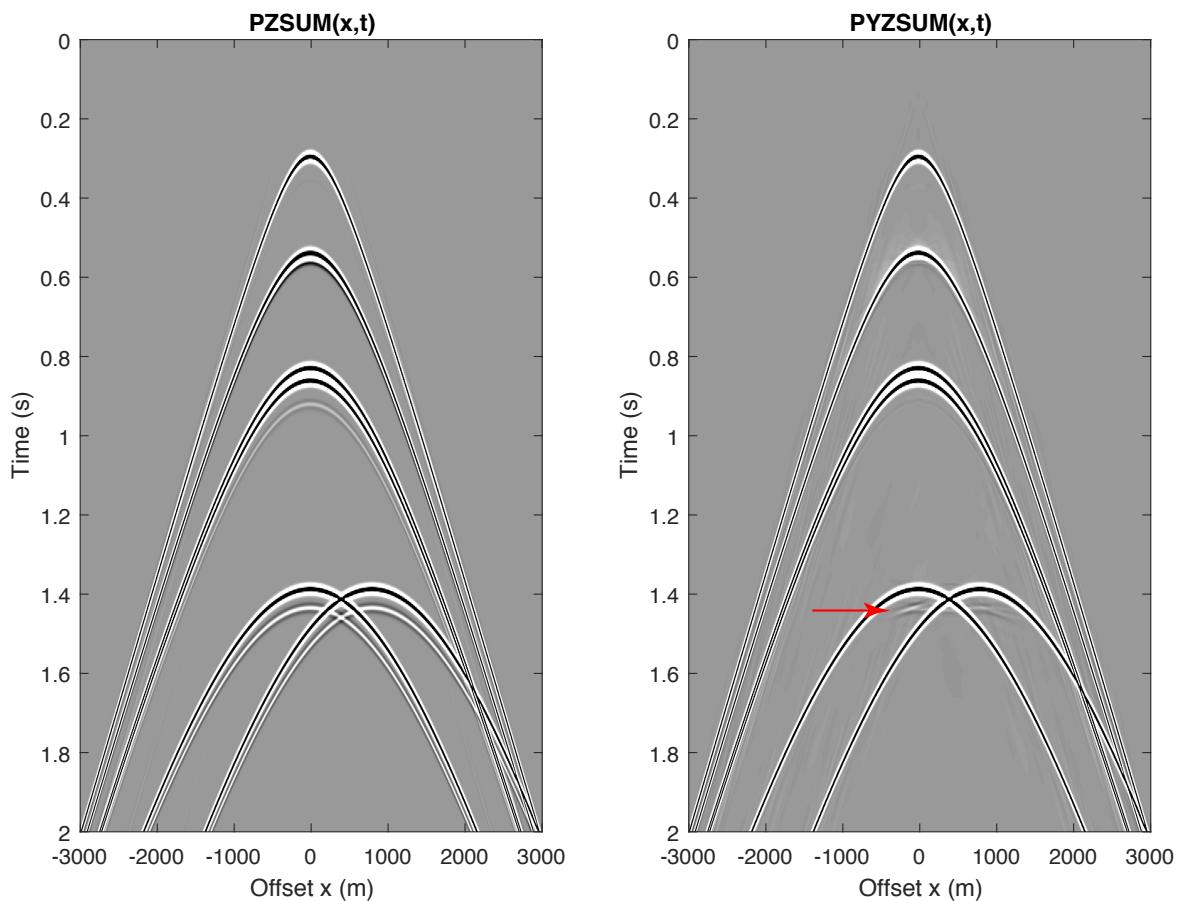


Figure 3-11: Comparison between the PZSUM and the PYZSUM as applied in tapered overlapping, moving windows. A time gain of t has been applied to the traces, the colorscale is equal for both images. Note that the errors of the previous section are now merely localised remaining ghosts. The moving windows thus improve upon the method.

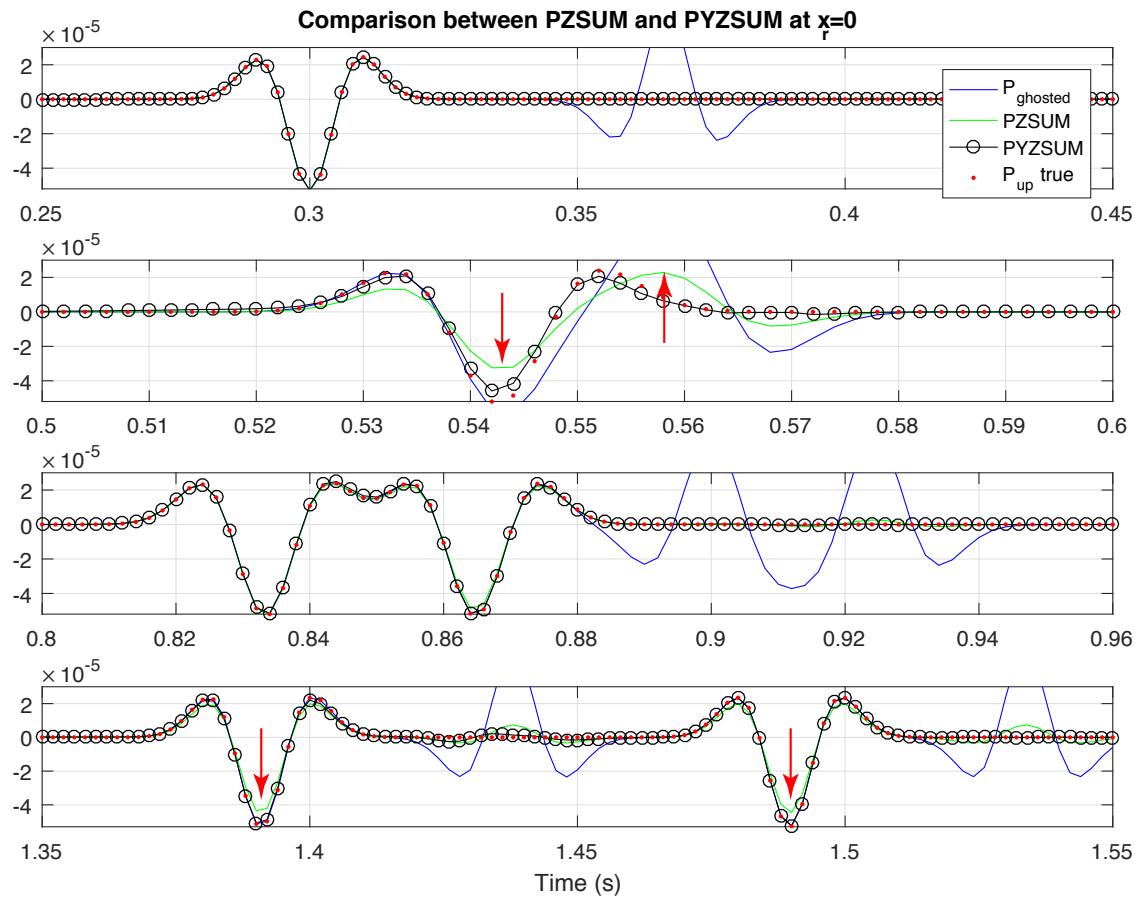


Figure 3-12: Comparison between the PZSUM and the tapered overlapping, moving window PYZSUM by looking at a single trace for $x_r = 0$. A time gain of t has been applied to the traces. We observe that the PYZSUM does a better job at approximating the true upgoing wavefield.

Furthermore, we may take a look at the power spectral density found in the traces, Figure 3-13. We note similar conclusions as made before: the PYZSUM achieves a better approximation to the upgoing wavefield than the PZSUM, in case of crossline energy propagation.

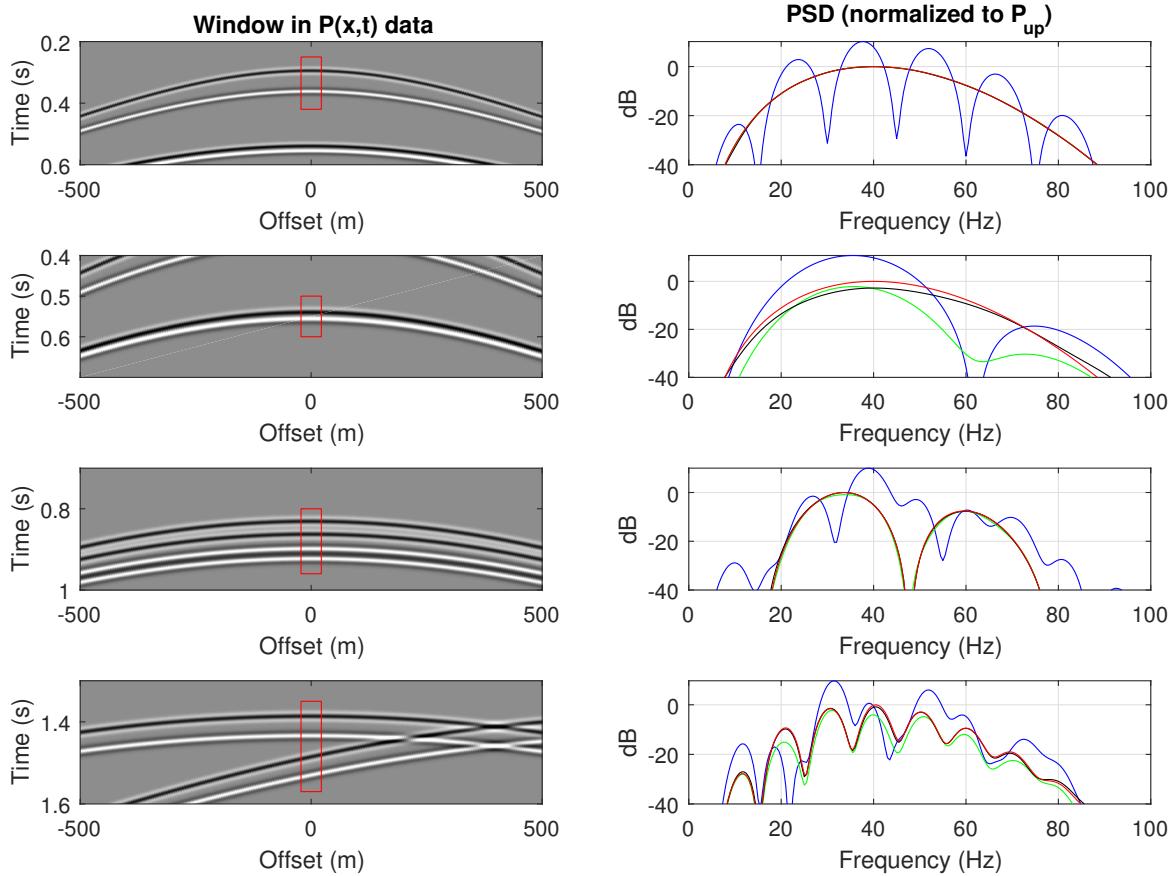


Figure 3-13: Power spectral density for the four events. The legend is the same as for Figure 3-12: blue for the ghost P , green for the PZSUM, black for the PYZSUM, red for the true upgoing wave.

We may draw the conclusion that the PYZSUM outperforms the PZSUM if:

1. **There is a (strong) crossline component.** In seismic data, this will usually be limited to the first few reflections and refractions.
2. **Events are separated by the $\tau-p_x$ transform.** Interference of events with different p_y in the $\bar{V}_y(\tau-p_x)$ domain will generate a scalar that is wrong for *both* events. One way to counteract this possibility, is to perform the PYZSUM on individual overlapping and moving windows in the $x-t$ domain. The windowing approach will further separate data, and keep errors localised.
3. **Special attention needs to be paid to the design of a median filter for short events in the $\tau-p_x$ domain.** The smoothing resulting from the 2D median filter will naturally favor events with energy along both the τ and p_x axis. The proposed workflow here naturally favors hyperbolas and broad wavelets, which will cover large areas in the $\tau-p_x$ panel.

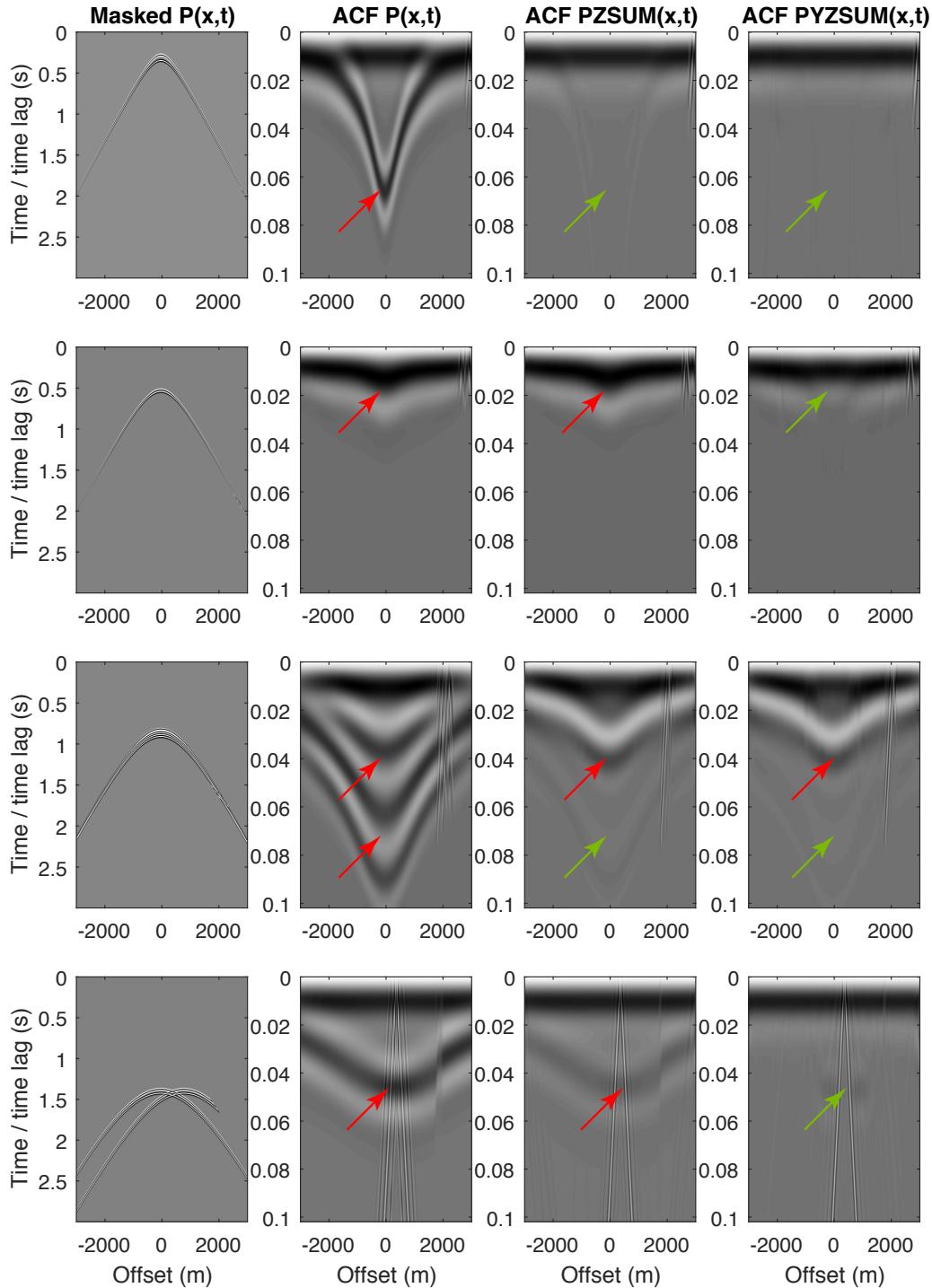


Figure 3-14: The autocorrelation for the four events, using a mask. The ghost in the autocorrelation has been annotated by a red arrow, the removal of the ghost by a green arrow. Linear artefacts arise when two events with different ghost time delays interfere, violating the ‘single time delay’ we assumed in the ACF.

3-3 Application to real data

The above described implementation of the PYZSUM was applied in 50% overlapping moving windows in the x - t , on two different real data examples.

3-3-1 Real data 1

We consider a first data set, characterized by simple vertically stratified geology. The data is relatively noisy, as it was acquired during unruly sea conditions. This will test whether the method is able to deal with noise. To highlight the improvement brought by the PYZSUM, we require strong crossline energy propagation. We therefore pick an outer cable with largest offset from the source (Figure 3-15). This makes the 2D approximation assumed by PZSUM invalid as the early reflections contain noticeable crossline energy. The nominal towing depth is $h = 18$ m, leading to a vertical incidence ghost delay of 24 ms.

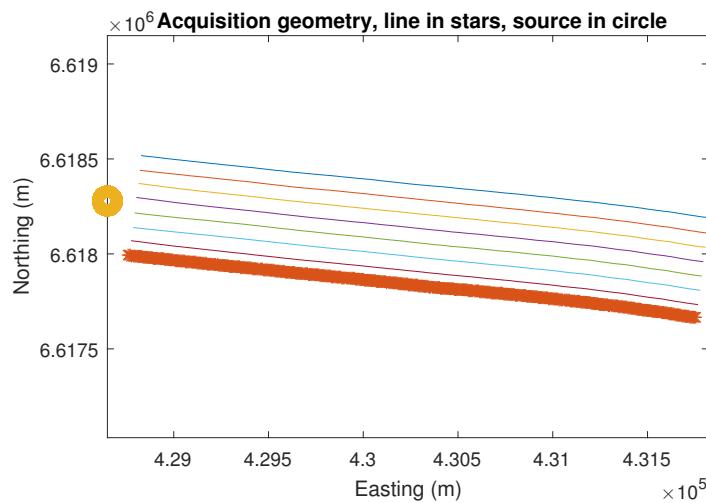


Figure 3-15: The acquisition geometry of the considered streamer in the real data set uses an outer cable, to maximize the crossline energy.

The complete set of pressure measurements, vertical particle velocity and crossline particle velocity is shown in Figure 3-16.

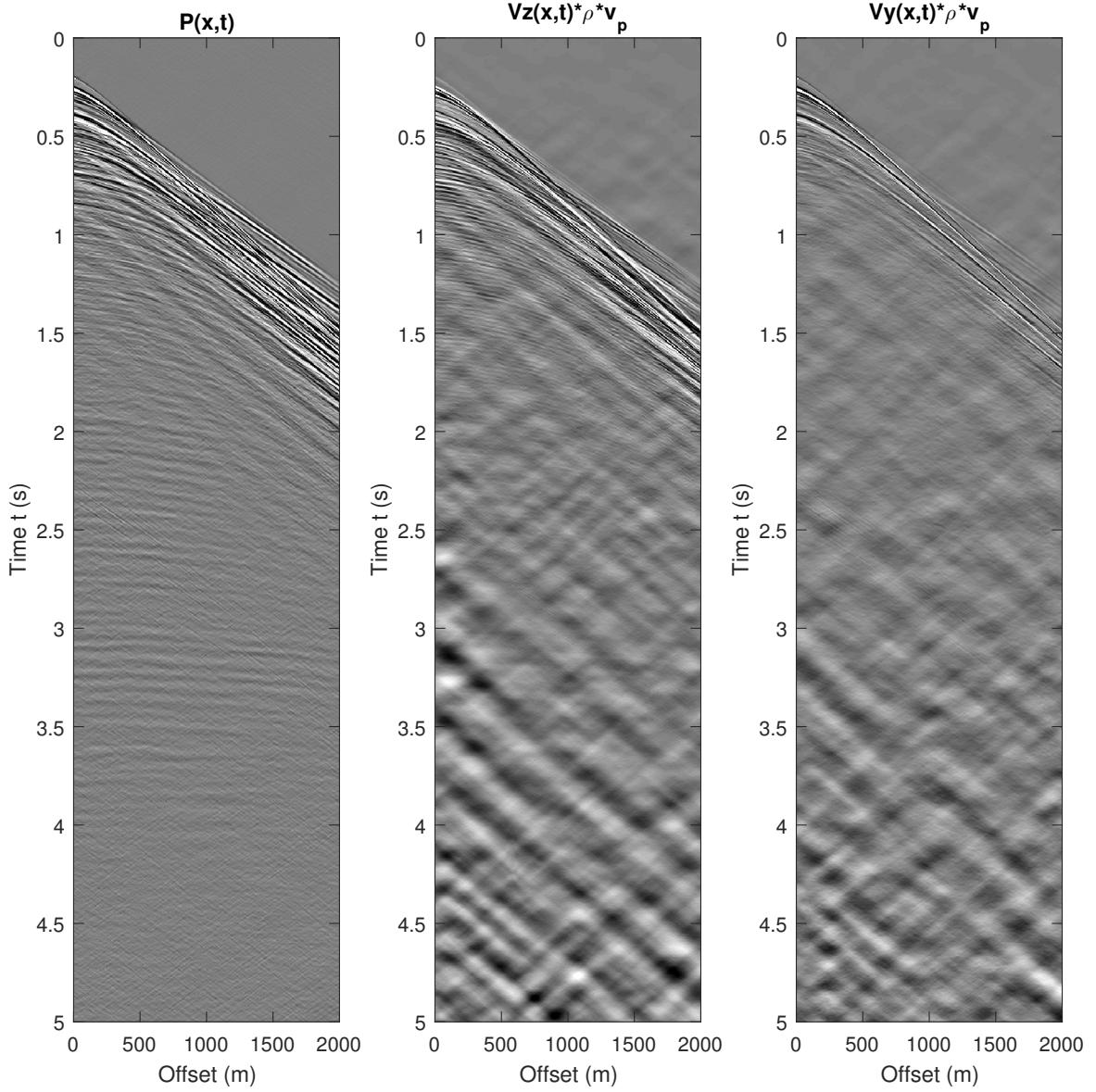


Figure 3-16: The multisensor measurements of real data set 1, after applying a time gain of t . Note that the V_y component is strong only on the first events. Also notice the strong low frequency noise characteristics (which become apparent particularly at later times after applying the time gain) on both particle velocity measurements. The color scale is the same for all figures.

The first second of the shot, deghosted with PZSUM, ODG and PYZSUM, is shown in Figure 3-17 (the full deghosted shot is shown in the appendix, Figure B-1). The deghosting by ODG leaves clear artefacts over the first reflection. This is expected due to the strong crossline propagation of energy, which invalidates the 2D approximation adopted by ODG. The PZSUM and PYZSUM results appear very similar and generally less blurred than the result from ODG.

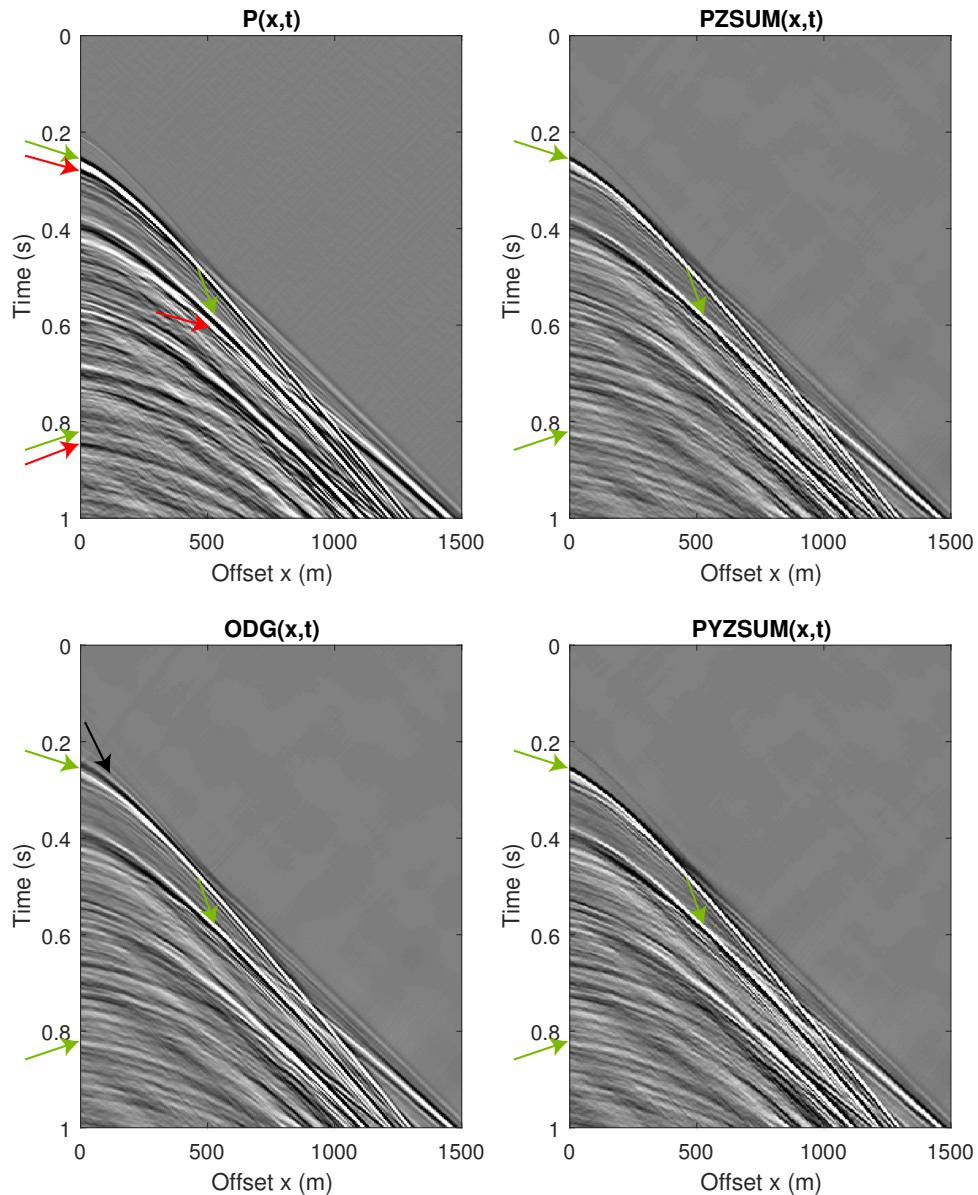


Figure 3-17: The shot gather of the real data set 1, deghosted with PZSUM, ODG and PYZSUM. The color scale has been clipped on purpose to show artefacts. A time gain of t^2 has been applied to the data. The green arrows denote some examples of the upgoing wavefield, red arrows their ghost, and the black arrow at the ODG highlights an artefact before the wavefield (the precursor).

The frequency spectra in Figure 3-18 show significant differences for the first small offset reflection events (the first three panels): a gain of about +6 dB (or, a factor 2) in filling the first frequency notch. This effect is expected for the following reason: the crossline energy propagation is greater than the inline propagation for a small inline – but large crossline – distance to the source. The 2D approximation of the PZSUM and ODG imply that the methods will not be able to recover the upgoing wavefield. The PYZSUM, conversely, takes this crossline component into account, and thus restores the upgoing wavefield better. The PZSUM and PYZSUM obtain similar results at larger offsets, as demonstrated in panel 4.

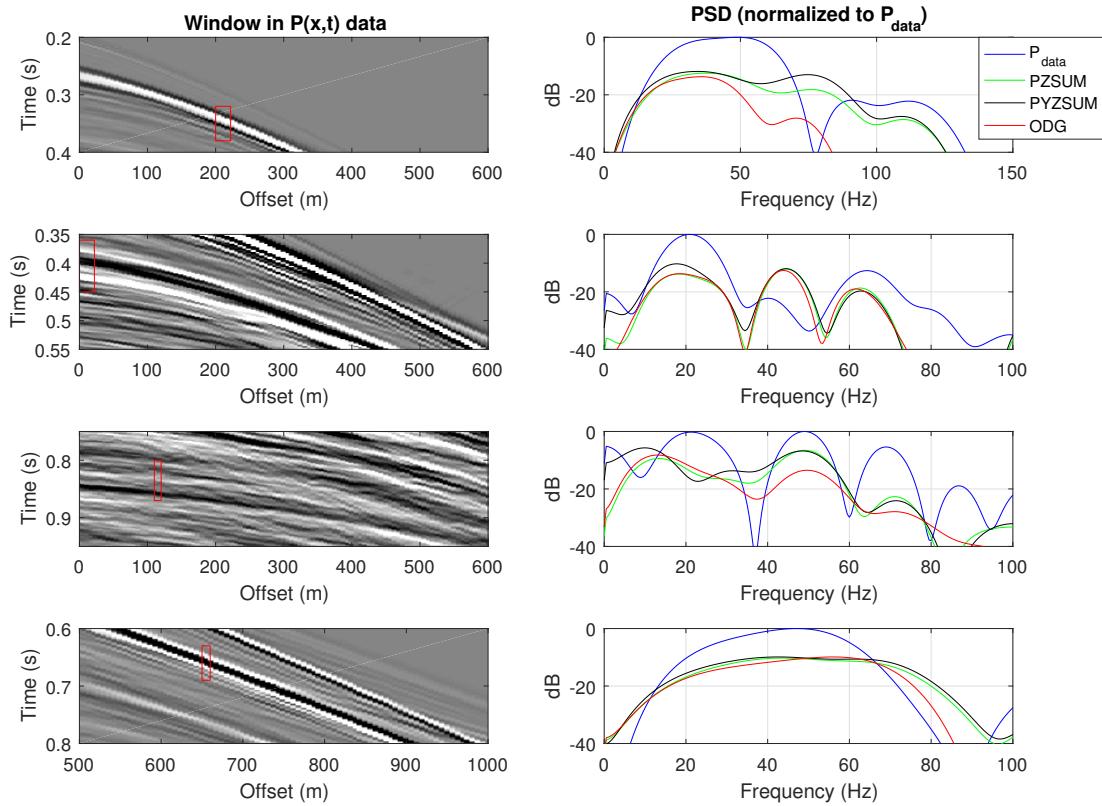


Figure 3-18: Power spectral density plot for the real data set 1 for the PZSUM, ODG and PYZSUM.

A normal moveout correction is applied to the data, with constant velocity of 1484 m/s (Figure 3-19). This flattens the first sea bottom reflection (0.23 s) and the multiples afterwards (0.38 s, 0.53 s, 0.68 s), which are assumed to contain the largest crossline energy components, and correspondingly the largest difference between the deghosting methods ODG, PZSUM and PYZSUM. We observe the ghost clearly on the NMO corrected input pressure data. The ODG shows an artefact advanced in time before the first arrival (annotated) – signalling again that the ODG uses a ghost model that does not correspond to the data. The PZSUM and PYZSUM both show a similar quality of deghosting.

We then perform the ACF on this NMO corrected section from $t = 0.15$ s to 0.79 s, as shown in the bottom of Figure 3-19. The input pressure data shows the ghost as a strong negative value

in the ACF function at approximately 0.018 s, corresponding to the ghost delay time. The deghosting should aim to remove this negative band. The ODG does not significantly reduce this negative band, but we observe a general improvement for the PZSUM and PYZSUM. As visible in their respective NMO windows, there appears to be an event shortly after the sea bottom reflection that interferes with the sea bottom reflection, which appears like a residual ghost, but is in fact a recovered primary reflection event. This reflection event also becomes visible in the ACF of the PZSUM and PYZSUM.

We then stack the ACF of all four panels, resulting in the coloured lines. The stack allows an easier comparison of the general ACF result. This ACF stack suggests that the ghost is removed best by the PYZSUM, as it shows the smallest negative value at the ghost time delay.

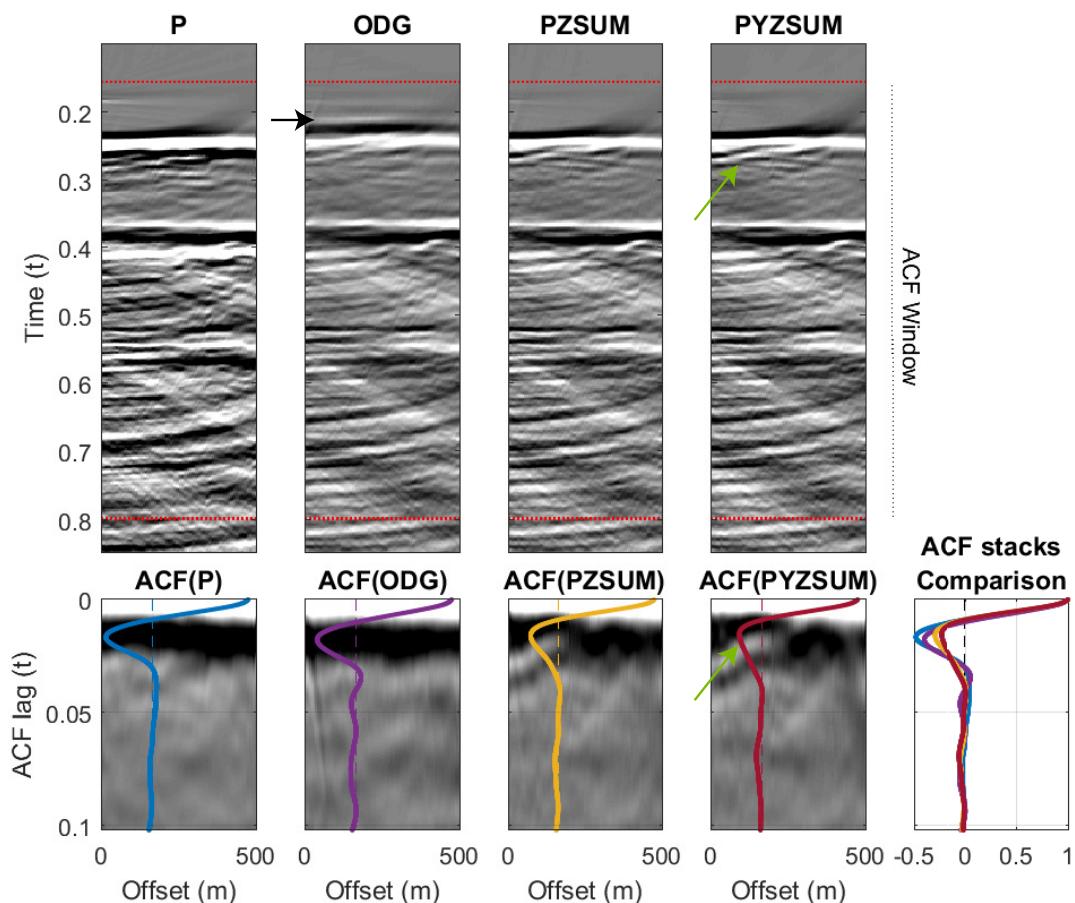


Figure 3-19: The ACF as calculated for the NMO corrected original and deghosted shots of real data 1. The coloured lines are the stacked autocorrelations. The PYZSUM has a marginally smaller negative ACF at the ghost delay time than PZSUM, and clearly performs better than the ODG. The green arrow at the PYZSUM shows a primary reflection just after the first sea bottom reflection, visible on the NMO corrected section and the ACF. The black arrow highlights the precursor artefact on the ODG.

3-3-2 Real data 2

The second set of real data comes from a more structurally complex geology, containing several diffractions, including energy propagating in the crossline direction. The nominal towing depth is 25 m, thus the time delay for vertical incidence is approximately 33 ms. The data comes again from an outer cable, suggesting significant crossline energy propagation must be expected for the first few reflections, similar to the previous data set. The data is generally more clean than the first real data set, although the poor S/N at low frequencies of the particle velocity measurements is still visible (Figure 3-20).

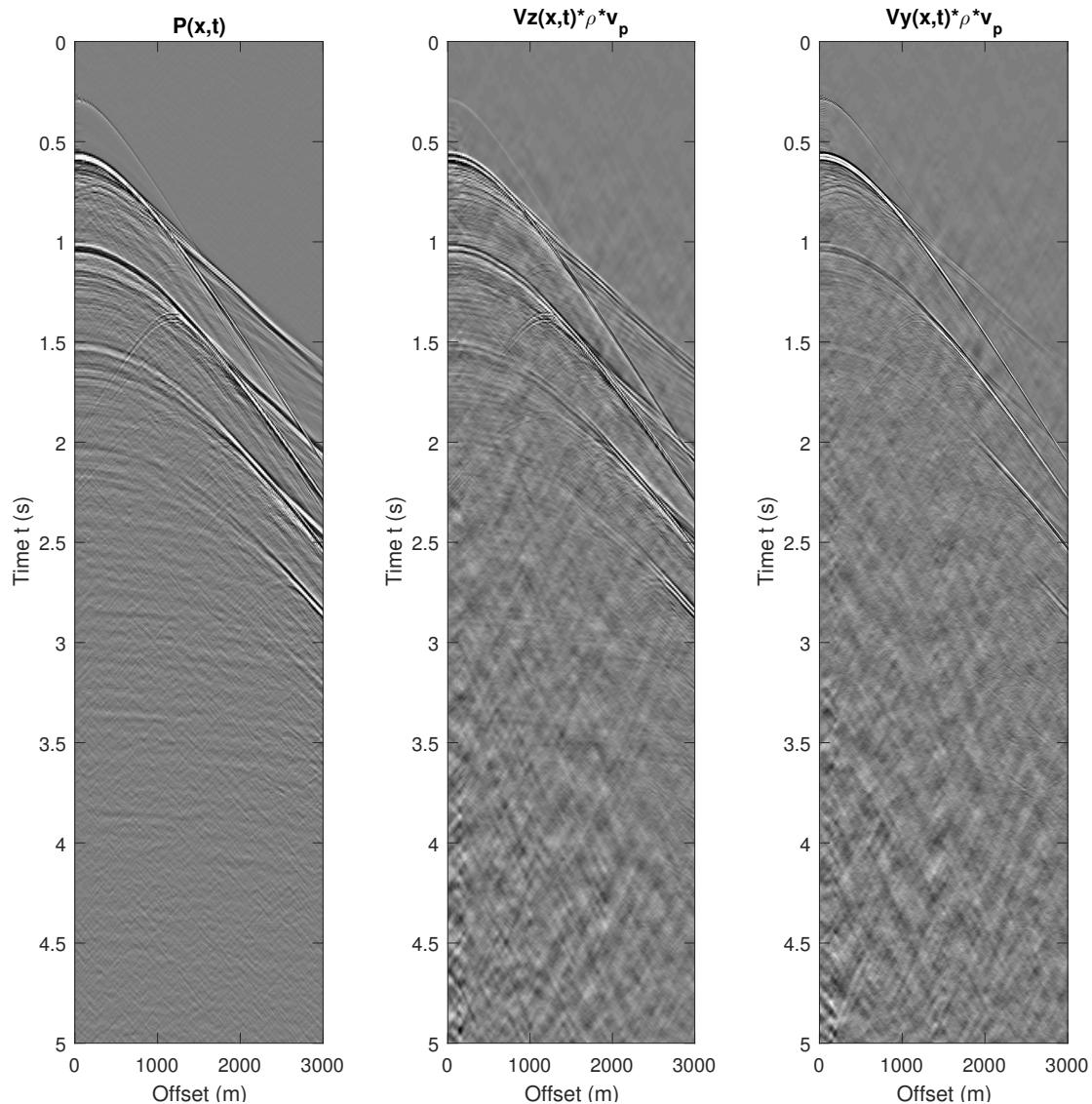


Figure 3-20: The multisensor measurements of real data set 2, after applying a time gain of t . Note that the V_y component is strong particularly on the sea bottom reflection (0.52 s) and its multiples (1 and 1.5 s). Note the strong low frequency noise characteristics, which become apparent at later times after applying the time gain, on both particle velocity measurements. The color scale is the same for all figures.

The first two seconds of the shot, deghosted with PZSUM, ODG and PYZSUM, are shown in Figure 3-21 (the full deghosted shot is shown in the appendix, Figure B-3). The deghosting by ODG again generates artefacts before the sea bottom reflection, again due to the invalid 2D approximation of the wavefield. The PZSUM and PYZSUM appear similar and generally less blurred than the result from ODG.

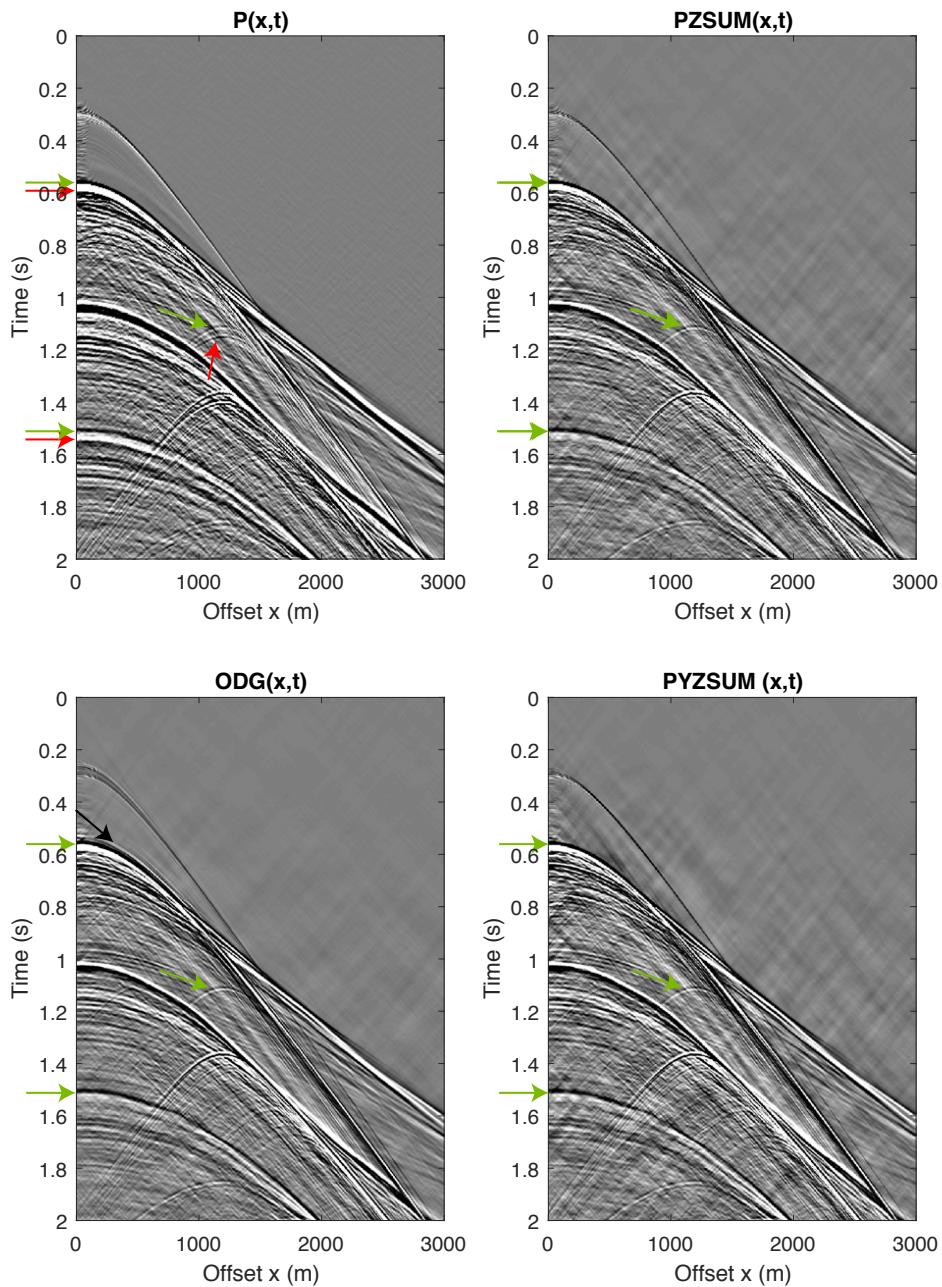


Figure 3-21: The shot gather of the real data set 2. The color scale has been clipped on purpose to show artefacts clearly. A time gain of t has been applied. The green arrows denote some examples of the upgoing wavefield, red arrows their ghost, and the black arrow at the ODG highlights an artefact before the wavefield (the precursor).

Like the first processed data set, we may again plot the frequency spectra for selected windows, as done in Figure 3-22. We observe a similar effect that the PYZSUM manages to fill in the first frequency notches with more energy than ODG and PZSUM at small inline receiver offsets (panels 1, 2 and 4). The PZSUM and PYZSUM obtain similar results for the diffraction event (panel 3).

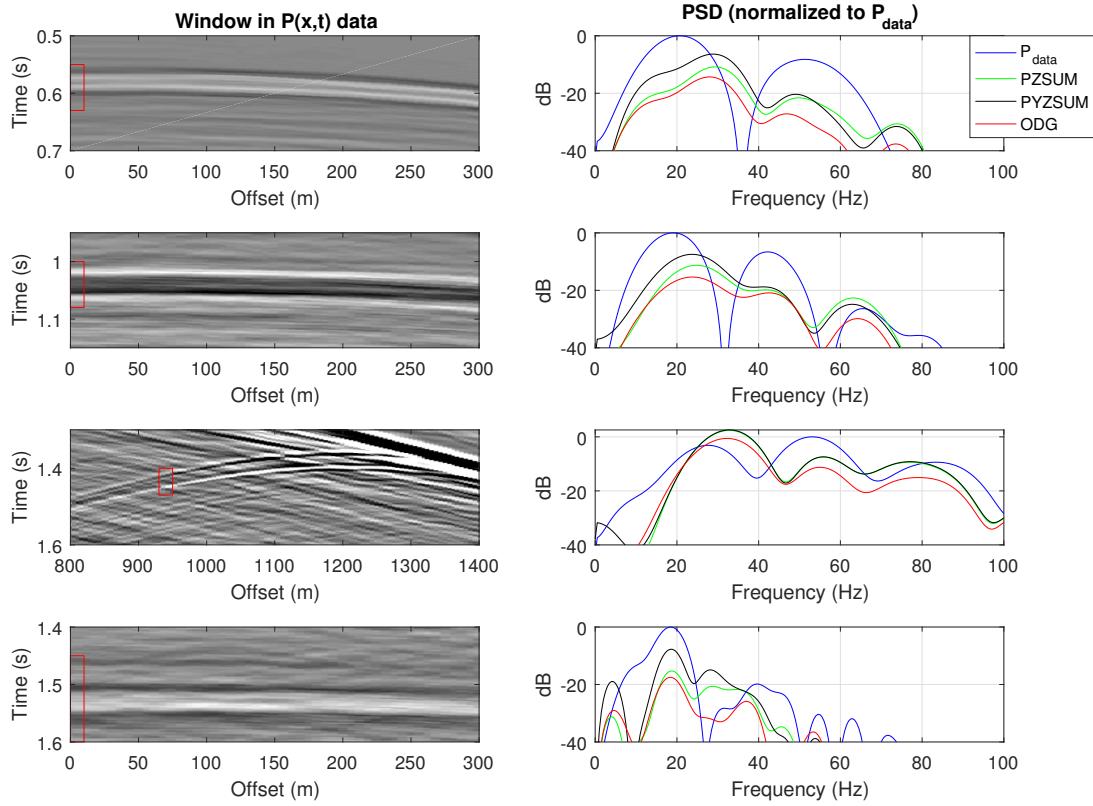


Figure 3-22: Power spectral density plot for the real data set 2 for the PZSUM, ODG and PYZSUM.

We then apply NMO correction to the data again with a constant velocity of 1484 m/s, to flatten the sea bottom reflection (0.55 s) and its multiple (1.02 s), shown in Figure 3-23. The ghost and its removal are again easy to observe.

The NMO correction is then used as an input to the ACF, shown in the bottom of Figure 3-23. We may observe the ghost as a negative value at ACF lag of 29 ms in the input pressure data. The ODG has not been able to fully remove this ghost artefact from the data, as the ACF still shows significant energy at this time lag. The PZSUM shows less energy at the ghost delay time, but its stacked ACF value (yellow) still shows a small remaining trough at this lag. The PYZSUM appears visually similar to the result of the PZSUM, but it's stacked ACF shows no remaining trough at the ghost delay time. In the comparison of all stacked ACF sections, we again observe that the PYZSUM removes most of the ghost.

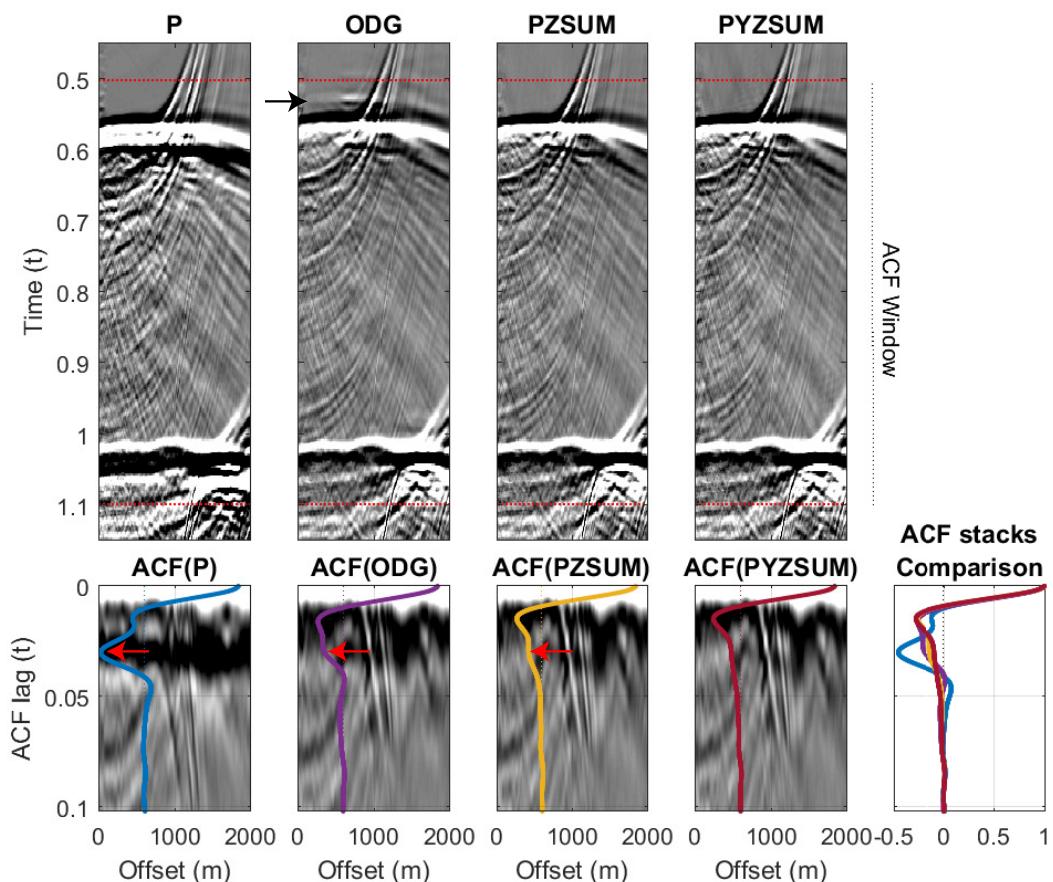


Figure 3-23: The ACF as calculated for the NMO corrected original and deghosted shots of real data 2. The red line is the stacked autocorrelation. The PYZSUM outperforms the other two deghosting methods, which show a residual ‘bump’ corresponding to the ghost (at about 29 ms).

Chapter 4

Data-driven multisensor adaptive deghosting

4-1 Motivation

4-1-1 Pitfalls of Energy Minimization

We established in Chapter 2 that energy minimization of single sensor data (cf. Schuberth, 2015) has limitations. Notably, it cannot estimate a ghost delay smaller than the reciprocal of the usable bandwidth (i.e. does not detect $t_\Delta < \frac{1}{f_{\text{notch}_1}}$). Secondly, the method obtains several potential time delays, and picks the one without a multiple integer time delay. In real data, a frequency notch may originate from the ghost but may also originate from destructive interference of primary reflections – the time delay corresponding to a notch in the data may accidentally lead to the ‘minimum energy’ solution. Multisensor acquisition offers advantages in this situation, as the ghost on P and V_z is complementary: notches due to the ghost are offset by exactly half a period, while the inherent notches of the upgoing wavefield are present in both. We will explore a way to expand the adaptive deghosting cf. Schuberth (2015) to the multisensor case, to overcome the limitations in energy minimization.

4-1-2 Multisensor cost functions

As in Schuberth (2015), we will assume our data is described by a ghost model, and will test which ghost model in a range of ‘trial’ ghost models best corresponds to the data. We thus need a cost function that indicates the proximity between the ghost model in the data and our trial ghost model.

Assume, for a moment, a single event and its ghost on the pressure \tilde{P}_{data} and particle velocity $\tilde{V}_{z,\text{data}}$. Assume we know the correct reflection coefficient, and we want to test a range of potential time delays t_Δ , and assess the performance. A range of multisensor cost functions suggest themselves:

1. Applying energy minimization on both components individually (\tilde{P}_{data} and $\tilde{V}_{z,\text{data}}$). The correct ghost model should appear as a ‘peak’ in both cost functions, if we take the reciprocal of the minimum energy for display purposes. We could combine the terms in a single cost function where we add both cost functions together. Note that deghosting $\tilde{V}_{z,\text{data}}$ by \tilde{G}_z returns the units of pressure, i.e. should also be normalized to the recorded pressure field.

$$J_1 = \frac{1}{2} \left(\frac{\left\| \frac{\tilde{P}_{\text{data}}}{\tilde{G}_p} \right\|_2^{-2} + \left\| \frac{\tilde{V}_{z,\text{data}}}{\tilde{G}_z} \right\|_2^{-2}}{\left\| \tilde{P}_{\text{data}} \right\|_2^{-2}} \right). \quad (4-1)$$

We call the method ‘incoherent energy minimization’, as we evaluate the components individually. We may predict that, similar to energy minimization, this method has an estimation limit at approximately half that for just P (as the first ghost notch on V_z lies at half that of the P data). Below this limit, both methods would fail to respond significantly to the inverse ghost filters.

2. Applying energy minimization on an estimation of the upgoing wavefield through ODG. This method is a ‘coherent energy minimization’, as we jointly use the two components prior to the energy minimization. Assuming no noise, the cost function may be formulated as the inverse of a normalized energy, for display purposes:

$$J_2 = \frac{\left\| \frac{\tilde{G}_p^* \tilde{P}_{\text{data}} + \tilde{G}_z^* \tilde{V}_{z,\text{data}}}{|\tilde{G}_p|^2 + |\tilde{G}_z|^2} \right\|_2^{-2}}{\left\| \tilde{P}_{\text{data}} \right\|_2^{-2}} \quad (4-2)$$

3. To formulate the ODG solution, a least squares cost function is derived and then minimized. By substituting the ODG solution back into the least squares cost function, we get a cost function which may be rewritten as a ‘cross ghost’ (Appendix A-4-4). The cross ghost means we apply a trial ghost model G_z to the ghosted pressure data, and apply a trial ghost model G_p to the ghosted vertical particle velocity data. If we subtract both terms, they should provide exactly 0. Again, we take the reciprocal of this function for display purposes below:

$$J_3 = \left\| \tilde{G}_z \tilde{P}_{\text{data}} - \tilde{G}_p \tilde{V}_{z,\text{data}} \right\|_2^{-2} \quad (4-3)$$

As a test case, consider a 40 Hz Ricker wavelet, which has a maximum frequency of 100 Hz. This means that the single component energy minimization has an estimation limit for $t_\Delta < \frac{1}{100}$ s, or 10 ms. Assuming vertical incidence, we set $p_z = \frac{1}{v_p}$ in the ghost model G_z . The three different cost functions are tested on time delays of 25 ms (Figure B-5), 5 ms (Figure B-6) and 2 ms (Figure 4-1).

The figures demonstrate the following. The first cost function J_1 shows less ambiguity in potential time delays as its maximum is the true time delay for 25 ms and 5 ms (unlike energy minimization). However, it also has an estimation limit at half of the single component energy minimization as predicted, making it unsuitable for the estimation of small delay times. The coherent energy minimization (J_2) manages to pick the time delay also at delays smaller than this estimation limit. However, the cost function has a rather ‘broad’ shape,

such that solutions off by several milliseconds may erroneously be picked as the optimal cost function. The cross ghost (J_3) picks the correct time delay with great precision, as it is the only maximum visible in the plot. Based on the examples, we realise that the cross ghost is the best cost function to estimate an adequate ghost model that fits the data. The other cost functions rely on the generation of ringing artefacts as a result of the deconvolution. The ringing does not significantly show up beyond the estimation limit ($\frac{1}{f_{\max}}$ for energy minimization, $\frac{1}{2f_{\max}}$ for incoherent energy minimization). The cross ghost thus works on a very different principle than energy minimization.

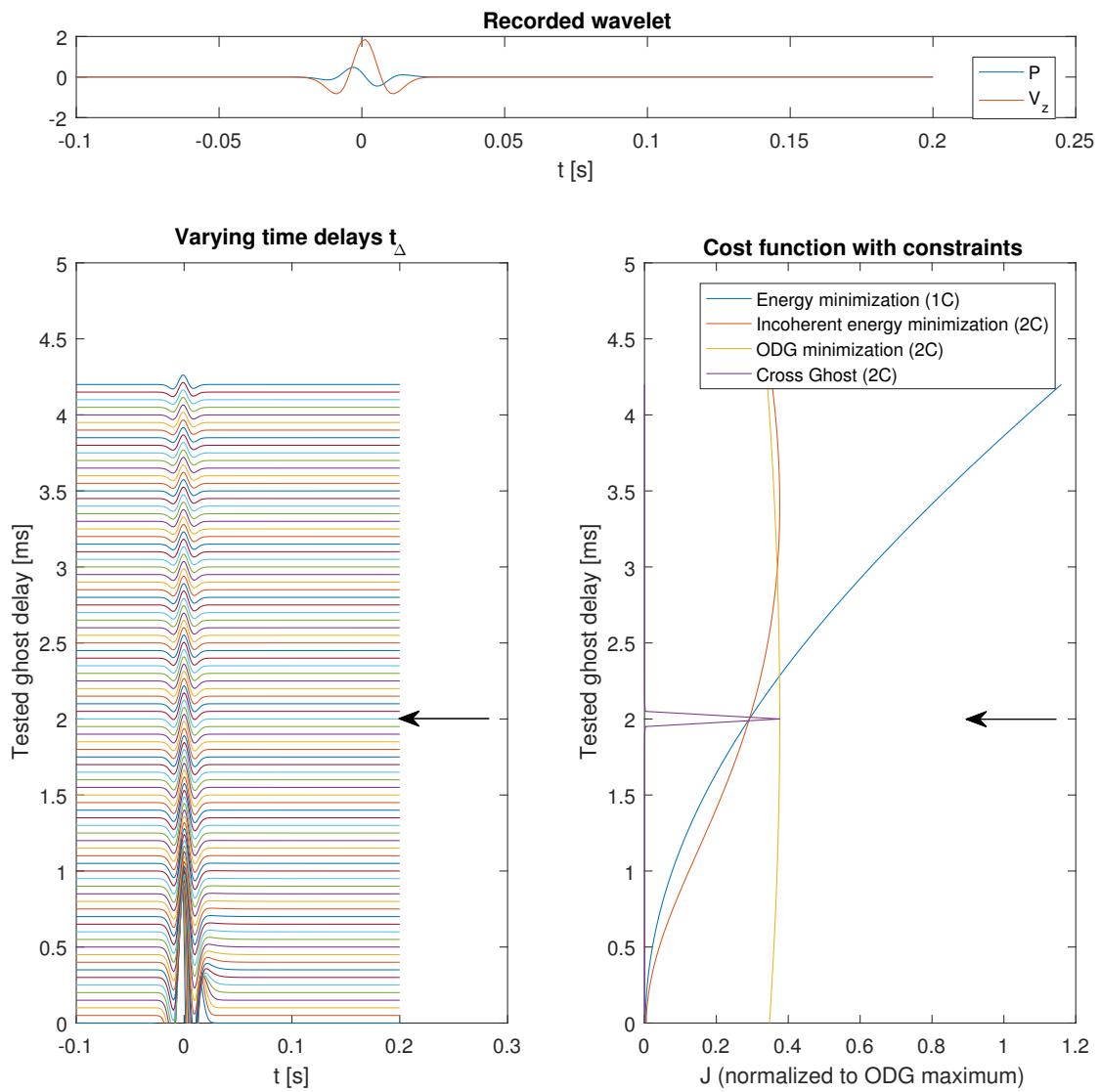


Figure 4-1: Three multisensor cost functions evalued to test their sensitivity to the true ghost delay of 2 ms, using a wavelet with a maximum frequency of 100 Hz. The cross ghost is the only cost function able to accurately detect the correct time delay.

4-2 Two and three component Cross Ghost

4-2-1 Two and three component Cross Ghost

The formulation of the two component cross ghost can be found in Equation 31 of [Amundsen \(1993\)](#). Assuming a source above the receiver, we know that all pressure data has a receiver ghost, except for the direct arrival which is only downgoing. We could describe the recorded data in the $f-k_x-k_y$ domain then as $\tilde{P}_{\text{data}} = \tilde{P}_{\text{direct}} + \tilde{G}_p \tilde{P}_{\text{up}}$, and $\tilde{V}_{z,\text{data}} = \left(-\frac{k_z}{\rho\omega} \tilde{P}_{\text{direct}} + \tilde{G}_z \tilde{P}_{\text{up}} \right)$. We can eliminate \tilde{P}_{up} from both equations to reach $\frac{\tilde{P}_{\text{data}} - \tilde{P}_{\text{direct}}}{\tilde{G}_p} = \frac{\tilde{V}_{z,\text{data}} + \frac{k_z}{\rho\omega} \tilde{P}_{\text{direct}}}{\tilde{G}_z}$, or:

$$2\tilde{P}_{\text{direct}} = \tilde{G}_z \tilde{P}_{\text{data}} - \tilde{G}_p \tilde{V}_{z,\text{data}}. \quad (4-4)$$

If we thus make the assumption that there is no direct arrival in the data, or remove it in prior steps, this function evaluates to 0 for the correct ghost model.

The cross ghost as a cost function comes up also in [Ozdemir et al. \(2009\)](#), where it is a required step in deriving the least squares ODG solution. In the patent, they describe the ghost system as $M_{\text{data}} = GP_{\text{up}}$, with M_{data} a vector with all the measured components, and G a vector with all corresponding ghost models. The least squares solution is found by minimizing $(\tilde{M} - \tilde{G}\tilde{P}_{\text{up}})^* C^{-1} (\tilde{M} - \tilde{G}\tilde{P}_{\text{up}})$, with C the noise covariance matrix. Substituting the ODG solution for \tilde{P}_{up} into this cost function returns an equation which looks exactly like the cross ghost ([Appendix A-4-4](#)). The additional advantage is the inclusion of the noise model. Assuming the noise is uncorrelated between different sensors, we may write the system in the $f-k_x-k_y$ domain as:

$$J(\alpha) = \frac{\sum_{i < j} \frac{|\tilde{G}_j(\alpha)\tilde{M}_i - \tilde{G}_i(\alpha)\tilde{M}_j|^2}{\sigma_i^2 \sigma_j^2}}{\sum_i \frac{|\tilde{G}_i(\alpha)|^2}{\sigma_i^2}}, \quad (4-5)$$

with α the desired parameters of the ghost model, i and j referring to the different components, and σ^2 the noise variance corresponding to each component.

For three components this cost function is written as:

$$J(\alpha) = \frac{\frac{|\tilde{G}_y \tilde{P}_{\text{data}} - \tilde{G}_p V_{y,\text{data}}|^2}{\sigma_p^2 \sigma_y^2} + \frac{|\tilde{G}_z \tilde{P}_{\text{data}} - \tilde{G}_p V_{z,\text{data}}|^2}{\sigma_p^2 \sigma_z^2} + \frac{|\tilde{G}_z \tilde{V}_{y,\text{data}} - \tilde{G}_y \tilde{V}_{z,\text{data}}|^2}{\sigma_y^2 \sigma_z^2}}{\frac{|\tilde{G}_p|^2}{\sigma_p^2} + \frac{|\tilde{G}_y|^2}{\sigma_y^2} + \frac{|\tilde{G}_z|^2}{\sigma_z^2}}. \quad (4-6)$$

4-2-2 Sensitivity analysis

The two components cross ghost

Assuming that our data consists of $\tilde{P}_{\text{data}} = \tilde{G}_p \tilde{P}_{\text{up}}$ and $\tilde{V}_{z,\text{data}} = \tilde{G}_z \tilde{P}_{\text{up}}$, we may rewrite the two-component cross ghost, and show that the cross ghost equals 0 for if the trial ghost models (\tilde{G}_p and \tilde{G}_z) correspond to the data. The formulation is easiest in the f - p domain:

$$\left(\underbrace{\frac{p'_z}{\rho} \left(1 + |r'_0| e^{-2\pi i f t'_\Delta} \right)}_{\text{trial ghost model of } V_z} \underbrace{\left(1 - |r_0| e^{-2\pi i f t_\Delta} \right) \hat{P}_{\text{up}}}_{\text{pressure data}} \right) - \left(\underbrace{\left(1 - |r'_0| e^{-2\pi i f t'_\Delta} \right)}_{\text{trial ghost model of } P} \underbrace{\frac{p_z}{\rho} \left(1 + |r_0| e^{-2\pi i f t_\Delta} \right) \hat{P}_{\text{up}}}_{\text{velocity data}} \right) \quad (4-7)$$

$$= \frac{p'_z}{\rho} \left(1 + |r'_0| e^{-2\pi i f t'_\Delta} - |r_0| e^{-2\pi i f t_\Delta} - |r'_0| |r_0| e^{-2\pi i f (t_\Delta + t'_\Delta)} \right) \hat{P}_{\text{up}} - \quad (4-8)$$

$$= \frac{p_z}{\rho} \left(1 - |r'_0| e^{-2\pi i f t'_\Delta} + |r_0| e^{-2\pi i f t_\Delta} - |r'_0| |r_0| e^{-2\pi i f (t_\Delta + t'_\Delta)} \right) \hat{P}_{\text{up}},$$

$$= \left(\frac{p'_z - p_z}{\rho} \right) \left(1 - |r'_0| |r_0| e^{-2\pi i f (t_\Delta + t'_\Delta)} \right) \hat{P}_{\text{up}} + \left(\frac{p'_z + p_z}{\rho} \right) \left(|r'_0| e^{-2\pi i f t'_\Delta} - |r_0| e^{-2\pi i f t_\Delta} \right) \hat{P}_{\text{up}}. \quad (4-9)$$

It is straightforward to show that the function goes to zero if all tested parameters (t'_Δ, r'_0, p'_z) correspond to the true values. The full sensitivity of the cost function is derived in Appendix A-5, the conclusions of which follow below.

Sensitivity of the two and three component cross ghost

The sensitivity of the three components (pressure, vertical particle velocity and crossline particle velocity) cross ghost is not analytically derived, and has instead been studied by evaluating the cost function for various ghost models and wavelet spectra. An important realisation is that the addition of an additional ghost model in the three component cross ghost – being $\hat{G}_y = \frac{p_y}{\rho} (1 - |r_0| e^{-2\pi i f t_\Delta})$ – appears to introduce the crossline slowness as yet another parameter that has to be established. However, under the assumption that the inline slowness p_x is already known through e.g. a τ - p_x transform, we know $p_y = \pm \sqrt{\frac{1}{v_p^2} - p_x^2 - p_z^2}$. Thus, p_y is a function of p_x and p_z . Thus the three component cross ghost function is sensitive to the same three parameters as the two component cross ghost: $(t'_\Delta, |r'_0|, p'_z)$. We want to study the shape of the cost function, to know if more minima exist in the cost function, and which parameters have the biggest influence on the cost function. This sensitivity is described for the three parameters below:

1. For a correct p_z and $|r_0|$, the cross ghost will find a single, correct, ghost time delay t_Δ by doing a 1D line search for t_Δ . A 1D line search means an evaluation of the cost function for many different t_Δ values, and picking the minimum. The minimum is found regardless of the frequency content of the wavelet. This makes the cross ghost approach superior to the sinigle sensor energy minimization approach which was limited by the frequency content. This confirms our finding that the cross ghost was able to estimate

a time delay of 2 ms in Figure 4-1, beyond the energy minimization estimation limit of either P or V_z .

2. For a varying p_z and t_Δ , we observe a coupling between p_z and t_Δ , additionally dependent on the time delay and frequency content. We employ a 2D line search, evaluating the cross ghost at all combinations of t_Δ and p_z , which is shown below. We plot the results in terms of Δt_Δ and Δp_z , which is the absolute difference with respect to the true ghost model, such that $\Delta t_\Delta = \Delta p_z = 0$ is the true desired result.

- (a) A high time delay (15+ ms). This shows relatively little coupling between the vertical slowness and time delay, such that a single global minimum is present.
 - i. With a low maximum wavelet frequency (50 Hz):

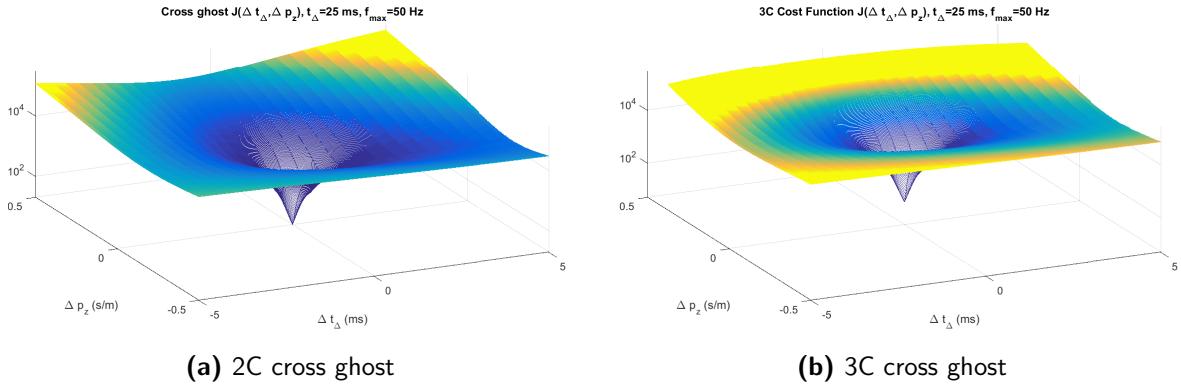


Figure 4-2: 2D line search along the cross ghost for a large time delay of 25 ms, with a low maximum frequency. The correct minimum time delay ($\Delta t_\Delta = 0$) is found irregardless of p_z , but the three component cross ghost shows the most distinct minimum.

- ii. With a high maximum wavelet frequency (180 Hz):

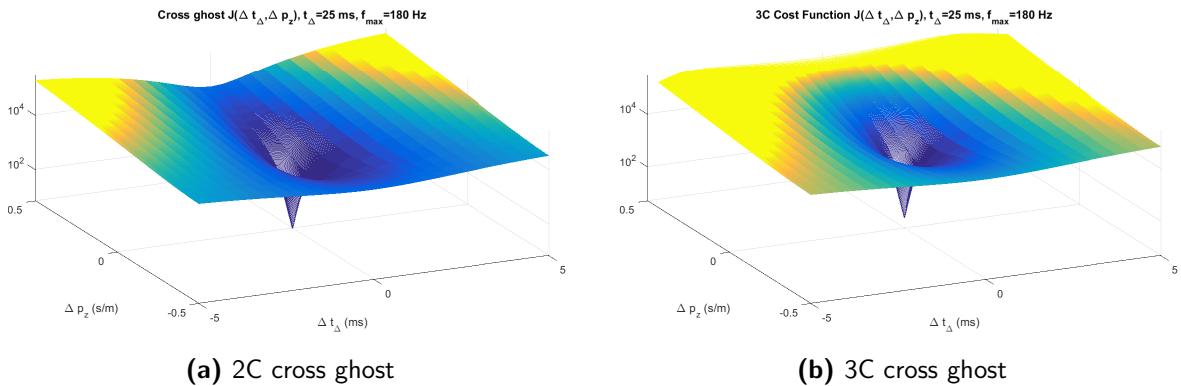


Figure 4-3: 2D line search along the cross ghost for a large time delay of 25 ms, with a high maximum frequency. The correct minimum time delay ($\Delta t_\Delta = 0$) is found irregardless of p_z , but the three component cross ghost shows the most distinct minimum.

We may generally observe that there is a single global minimum, i.e. a single combination of p_z and t_Δ which minimizes the cross ghost. The three component

cross ghost provides the largest distinction between the wrong and right ghost model.

- (b) A small time delay (10- ms). The cross ghost cost function shows a coupling between p_z and t_Δ .

- i. With a low maximum wavelet frequency (50 Hz):

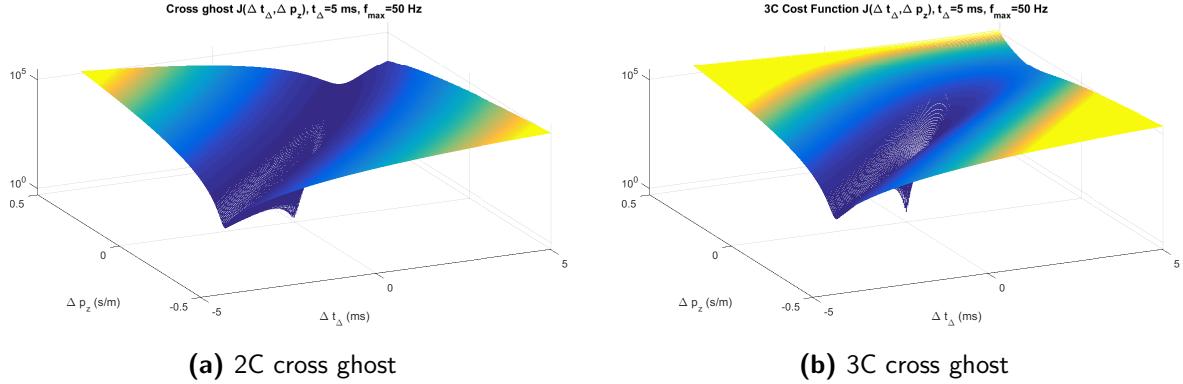


Figure 4-4: 2D line search along the cross ghost for a small time delay of 25 ms, with a low maximum frequency. The minimum in time delay t_Δ and p_z couple, such that picking one variable wrong leads to a wrong estimation of the other.

- ii. With a high maximum wavelet frequency (180 Hz):

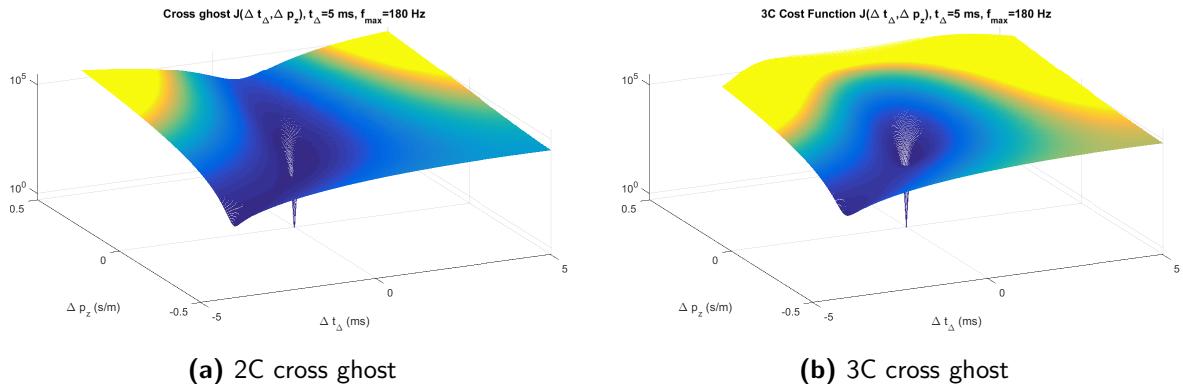


Figure 4-5: 2D line search along the cross ghost for a small time delay of 25 ms, with a high maximum frequency. The minimum in time delay t_Δ and p_z couple, such that picking one variable wrong leads to a wrong estimation of the other.

Although the global minimum still lies at the true ghost model, we observe a ‘valley’ in the cost function of minima corresponding to a wrong t_Δ and p_z . The three component cross ghost and a higher frequency content both provide greater distinction between the global minimum and the other values.

The values t_Δ and p_z clearly couple in a way that affects the cross ghost in a complex manner, with a global minimum only at the correct combination of t_Δ and p_z .

3. The cost function is hardly affected by varying the reflection coefficient, i.e. the cross ghost is minimum for the correct delay time, irregardless of the tried reflection coefficient. This is shown in Figure 4-6.

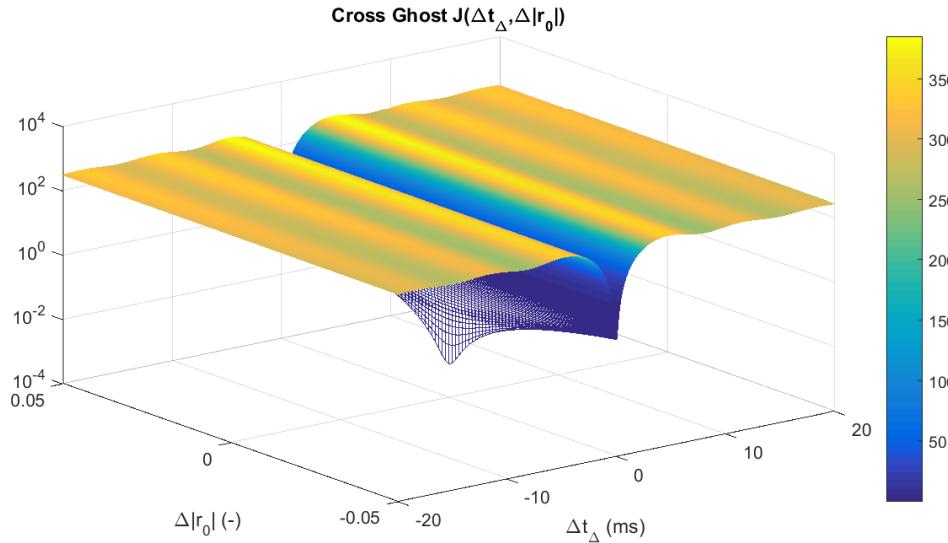


Figure 4-6: Evaluating the upper bound of the the cross ghost cost function, while varying both the dimensionless reflection coefficient and the time delay around the true (desired) values. Its clear that the time delay has the biggest importance. The true solution lies at $\Delta t_\Delta = \Delta |r_0| = 0$, which is the global minimum of this function.

Based on the sensitivity analysis above, we observe that the cross ghost is particularly sensitive to t_Δ and p_z . If we want to use the cross ghost as a cost function to estimate the ghost model in our data, we must thus employ a 2D search (just like the figures above) to find the global minimum at the correct p_z and t_Δ . The reflection coefficient may be ignored in the search initially, and may be assumed to be 0.95. The sensitivity analysis shows that quadratic convergence is found to the true $|r_0|$ after finding the correct t_Δ and p_z . If desired, the reflection coefficient may thus be estimated with a 1D line search, after estimating the correct time delay and vertical slowness.

4-2-3 Implementation

The cross ghost may be used to drive a two or three component adaptive deghosting algorithm. We propose the steps for such an algorithm below.

1. **Transform the section to windows.** We established that the cross ghost can help to find the ghost model parameters (ghost time delay t_Δ and vertical slowness p_z) on a single trace. The natural domain to apply this cost function in is the $\tau-p_x$ domain, because a single p_x trace is associated to approximately a single time delay. As we know that the time delay and slowness may vary through time and space, we choose to apply the method not on the entire shot gather. As such, 50% overlapping tapered

moving windows in the space-time domain are deghosted individually, and recombined afterwards.

The windows that we employ are ideally as small as possible, such that we allow for the largest variation of our parameters through space and time. A high quality $\tau-p_x$ transform, conversely, requires the use of large offsets and long time intervals. A compromise must thus be found. We choose to take windows of approximately 0.2 seconds (100 samples at $\Delta t = 2$ ms) and 612 meters (100 samples at $\Delta x = 6.125$ m), which produce a relatively sharp $\tau-p_x$ panel (Figure 4-7).

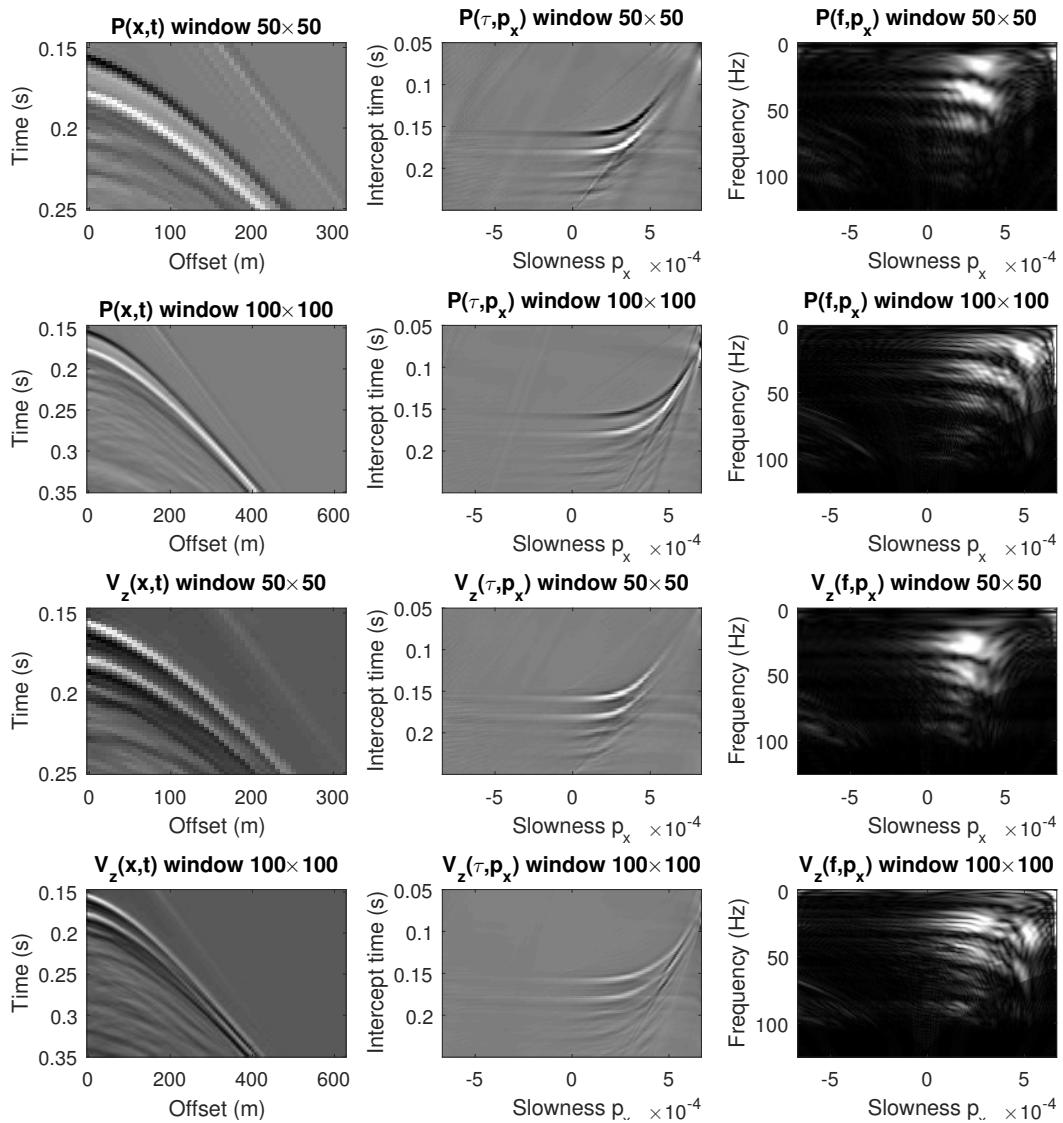


Figure 4-7: The window size influences the quality of the $\tau-p_x$ transform, where a larger window will provide a more ‘focused’ transform. The data is from the real data set 1 (see previous chapter). The used settings in this thesis are windows of 100×100 samples in space and time, such the second and fourth line give a sense of the expected resolution that goes into each step.

2. Transform P , V_z and V_y to the $f\text{-}p_x$ domain.

We apply first a $\tau\text{-}p_x$ transform as in the previous chapter, using the same slowness sampling $\Delta p_z = \frac{2\Delta t}{|x_{\max}|}$ and slowness range $\pm \frac{1}{1200}$ s/m. We can then apply a Fast Fourier Transform (FFT) along the τ direction to get the data in the $f\text{-}p_x$ domain. The number of frequencies in the FFT does not have to be oversampled in the Fourier Transform-based $\tau\text{-}p_x$ transform used in this thesis. The $\tau\text{-}p_x$ transform result contains essentially all the frequency information possible.

3. Perform the 2D line search for the cross ghost per p_x trace.

We can now evaluate the two or three component cost function, Equation (4-6), per p_x trace. We perform a 2D line search along time and vertical slowness.

- (a) The ghost delay times are scanned between horizontal incidence and vertical incidence, i.e. between $t_\Delta = 0$ and $t_\Delta = 2(h + \Delta h)\sqrt{\frac{1}{v_p^2} - p_x^2}$ (allowing some margin of error for the streamer depth h).
- (b) The vertical slowness is scanned between horizontal incidence and vertical incidence, i.e. between $p_z = 0$ and $p_z = \sqrt{\frac{1}{v_p^2} - p_x^2}$.

For the 3 component cost function, we must consider that $p_y = \pm\sqrt{\frac{1}{v_p^2} - p_x^2 - p_z^2}$, it may thus be positive or negative. We decide to pick the sign of p_y as that sign which minimizes the cost function.

Our implementation of the 2D line search evaluates 200 different time delays and 10 different p_z values, for each slowness in the signal cone ($|p_x| < \frac{1}{v_p}$). We then pick the minimum value of the cost function and record the corresponding t_Δ and p_z .

4. Deghost in the $\tau\text{-}p_x$ domain

The information obtained through the line search may now be used to deghost the data. We have at least two methods available now: we can perform a PZSUM in the $\tau\text{-}p_x$ domain using the found p_z as our 3D scalar, i.e. $\bar{P}_{\text{up}} = \frac{1}{2} \left(\bar{P} + \frac{\rho}{p_z} \bar{V}_z \right)$, or perform ODG in the $f\text{-}p_x$ domain. The PZSUM is generally the preferred method, as it will not generate any ringing at all, whereas the ODG with an incorrect ghost model will cause ringing artefacts. We choose to merge the two results, following Caprioli et al. (2012), using the ODG approach to keep control over the noise characteristics of the output for frequencies below 20 Hz, and the PZSUM in the good S/N spectrum above 20 Hz.

4-2-4 Illustration on synthetic data

We can prove the concept on the same synthetic data set used for the PYZSUM (Section 3-2-3). An important distinction is that we deghost the four test cases individually, rather than processing the data set as a whole. This way, we can show the picked time delay and picked vertical slowness compared to their theoretical functions (Figure 4-8). We can make the following observations based on the picked ghost delay time t_Δ and p_z :

1. Inline propagation: The 3C cross ghost is able to perfectly find the ghost parameters for inline propagation, thus can reproduce PZSUM results.

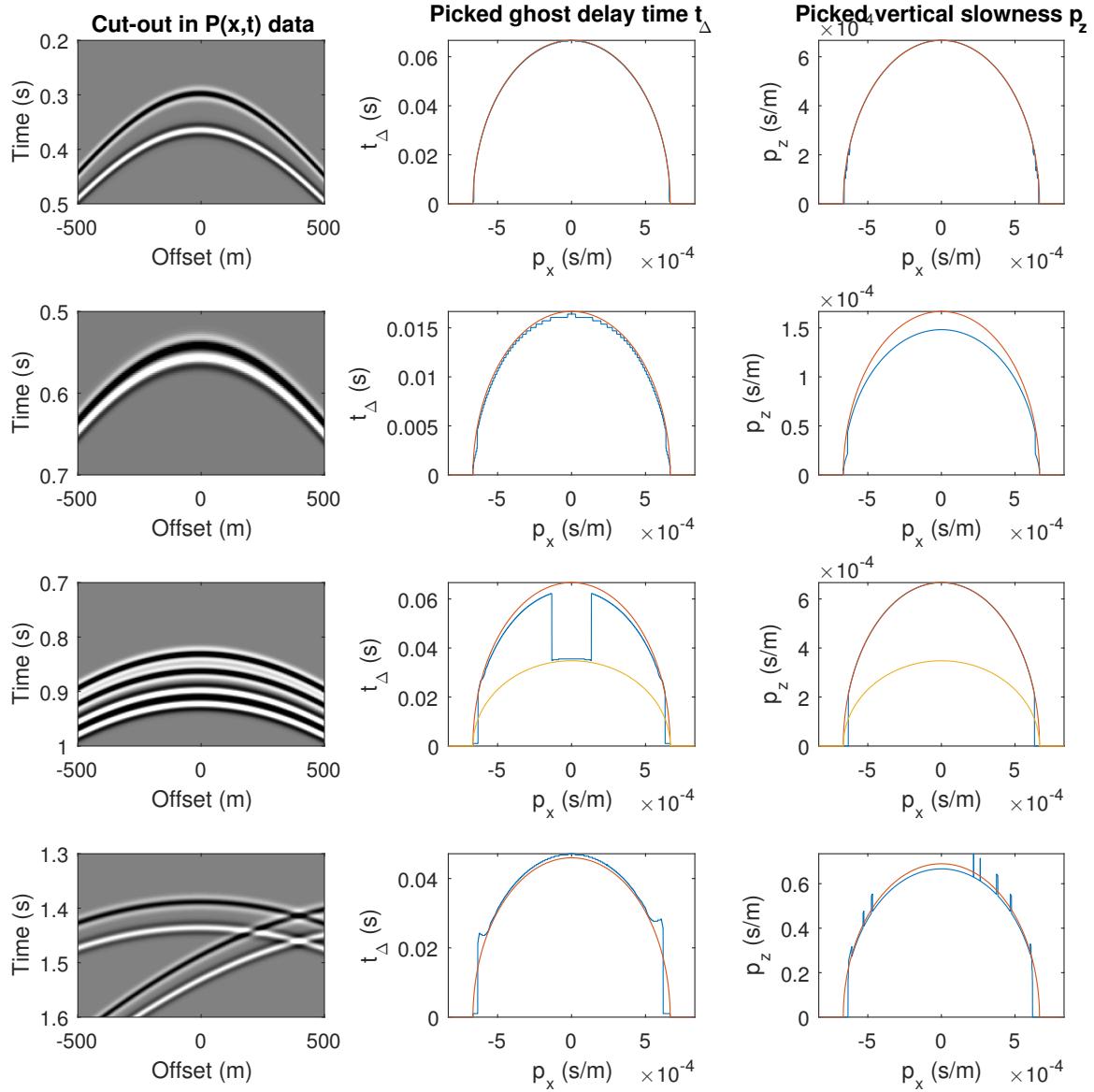


Figure 4-8: The time delay and vertical slowness picked by minimizing the cross ghost function (blue) compared to the true solutions (red, yellow if a second true solution exists). Notice that the time delay corresponds to the true time delay fairly well – whereas the vertical slowness may be off. The reason for this is that the line search along the time delay takes 200 steps, and the line search along p_z takes 10 steps. Whenever the p_z is thus not completely ‘on point’, this is due to the very rough approximation to the true p_z .

2. Significant crossline propagation: The 3C cross ghost finds the correct ghost delay for the extreme angle of incidence, but estimates a too small p_z . The reason for this is that the line search along p_z samples only 10 values. This error is thus caused by the implementation of a low resolution search along p_z .
3. Two intertwined events: If two events with different time delays are present on the same p_x trace, the cross ghost function still can only return a single minimum time delay. In this case, we observe that the cost function is inconsistent. For the vertical incidence, it picks the (correct) smallest time delay, whereas for greater angles of incidence it picks (incorrect) higher time delays. In all cases, it picks the vertical slowness belonging to the event with inline propagation.
4. The interfering events: In the case of two opposing p_y values, the three component cross ghost cannot properly find the correct time delay at all because it can only choose $+/-p_y$, and neither option fits the data well. However, the two component cross ghost (as used for specifically this fourth event), does not have this issue. The two component cross ghost is thus able to find nearly the correct time delay and vertical slowness. The artefacts appearing at the vertical slowness are, again, due to the low resolution line search along p_z . The lack of results beyond $\pm 5.5 \times 10^{-4}$ s/m is due to the lack of these slowness values in the data itself.

The data is deghosted based on the obtained t_Δ and p_z , as is shown in Figure 4-9, and compared against the ODG algorithm with a fixed nominal depth. We see that the adaptive deghosting algorithm clearly outperforms the ODG, except at the intertwined events, which causes artefacts. We can confirm this observations by looking at the zero offset traces in Figure 4-10 and the power spectral density in Figure 4-11. The autocorrelation of Figure 4-12 further confirms that the ghost is removed better with the adaptive deghosting than with the deterministic ODG.

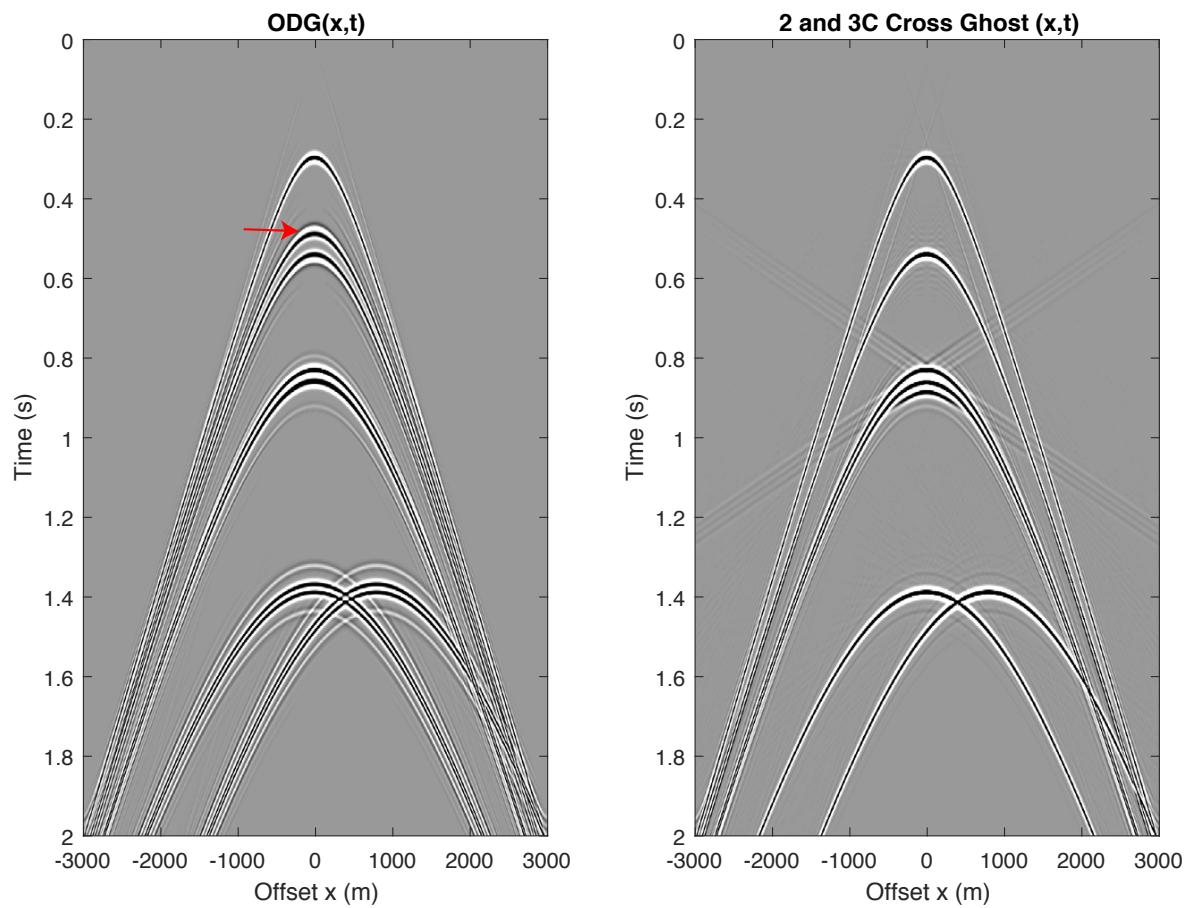


Figure 4-9: The synthetic data deghosted with the ODG method (nominal depth $h = 50$ m), versus the adaptive deghosting method based on the cross ghost. The poor deghosting result from 0.8 seconds and further down was to be expected, as they mix events of different ghost time delays, whereas the adaptive method can only pick a single time delay per slowness p_x . A time gain of t has been applied to the data.

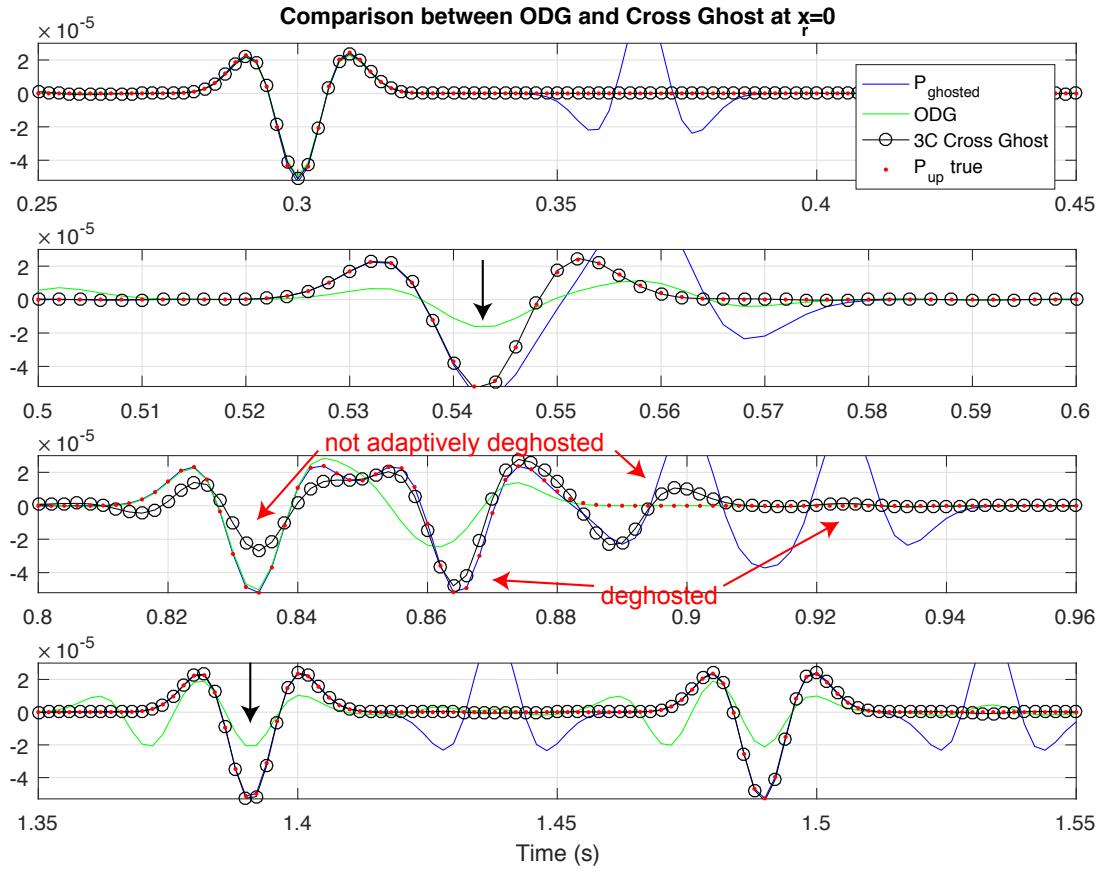


Figure 4-10: Comparison between the deterministic ODG and multisensor adaptive deghosting by comparing the zero offset traces. Note how the adaptive deghosting provides adequate deghosting for the second event only, as it picked the small delay time solution. Conversely, the ODG only correctly deghosts completely inline propagation of events.

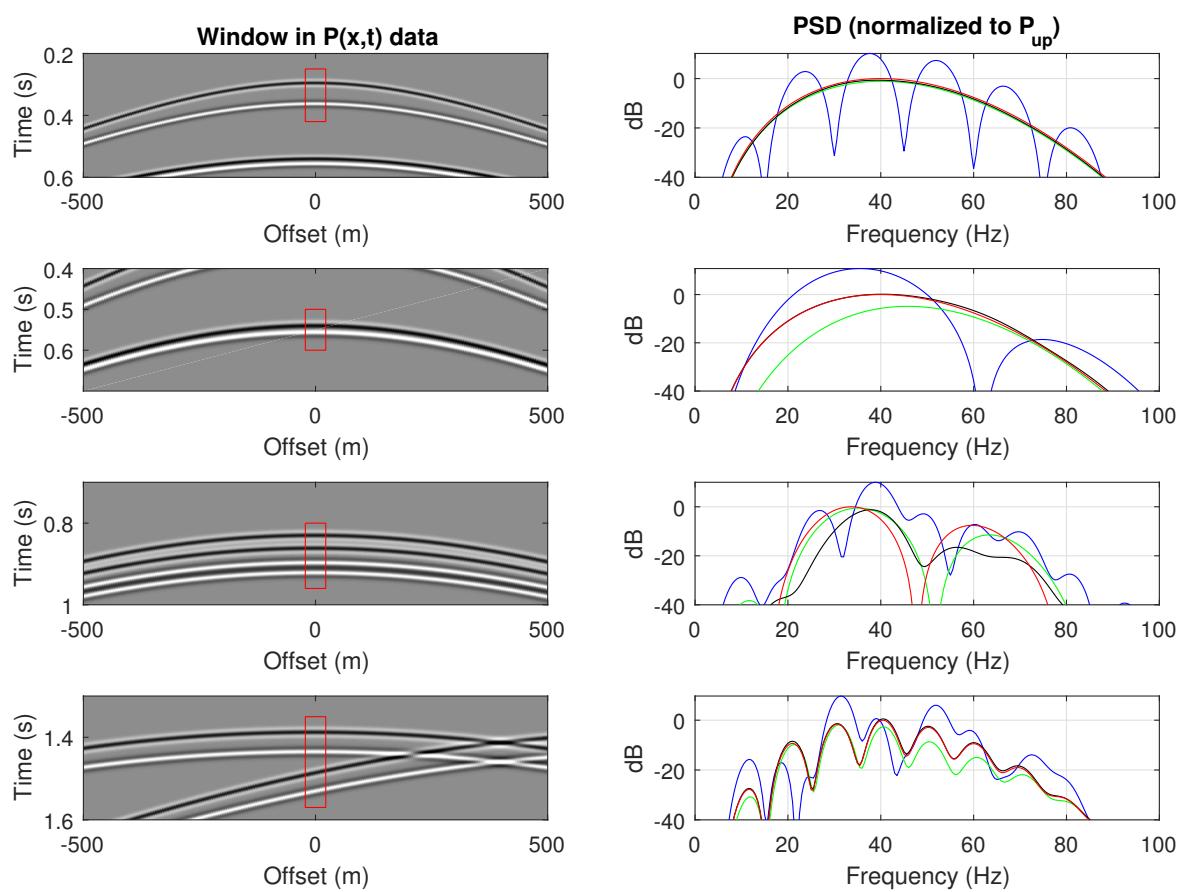


Figure 4-11: Power spectral density plot for the ODG and multisensor adaptive deghosting on synthetic data. The legend is similar to that used in the previous Figure 4-10: blue for the ghosted pressure, green for the ODG, black for the adaptive deghosting with the cross ghost, and red for the true upgoing wavefield.

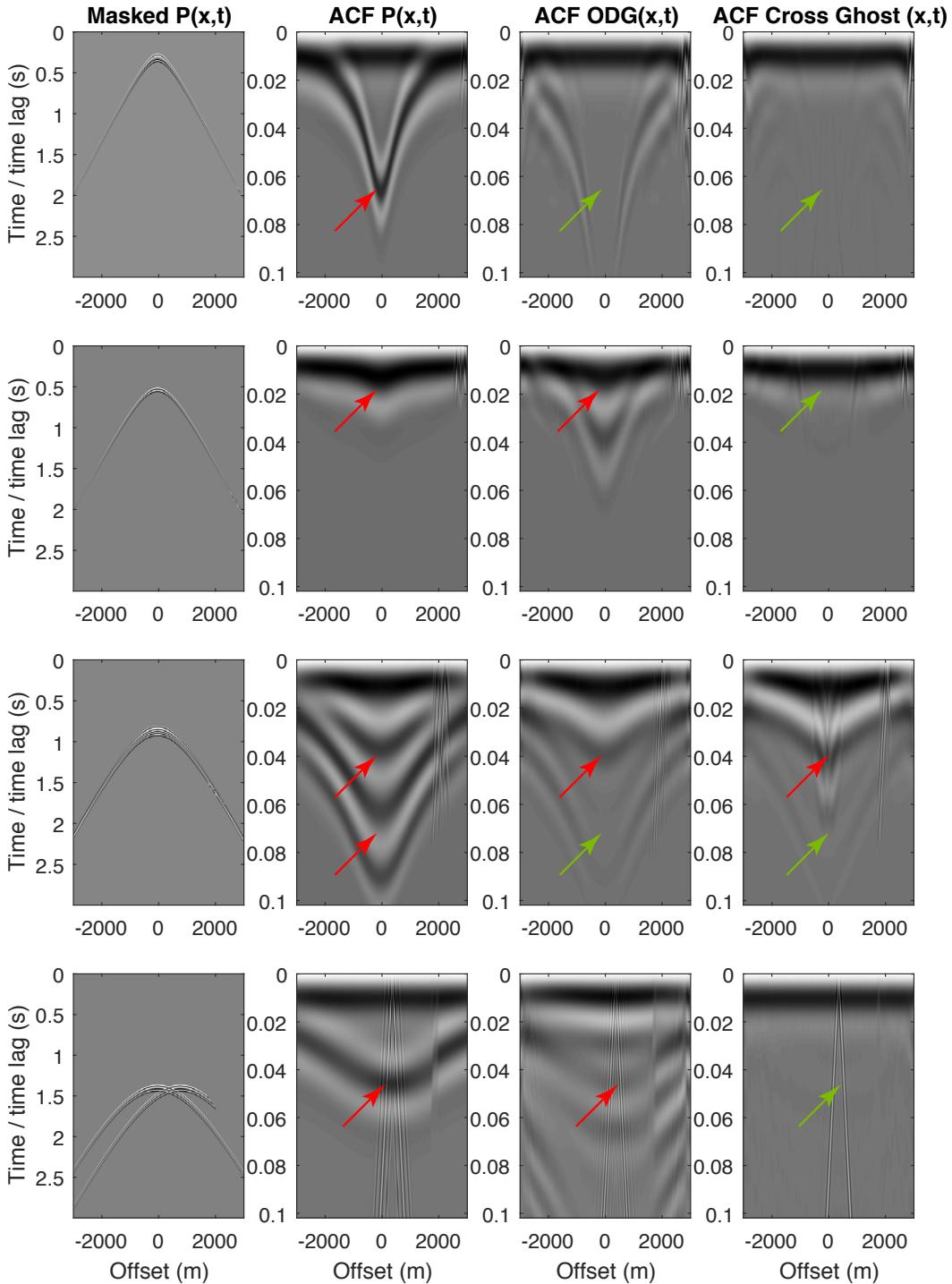


Figure 4-12: The autocorrelation for the four events, using a mask. The ghost in the autocorrelation has been annotated by a red arrow, the removal of the ghost by a green arrow. Linear artefacts arise when two events with different ghost time delays interfere, violating the ‘single time delay’ we assumed in the ACF.

4-3 Application to real data

See Section 3-2-3 for information about the used QC methods and used data.

4-3-1 Real data 1

The first second of the deghosted shot is shown in Figure 4-13 (the full deghosted shot is shown in the appendix, Figure B-2). We may make the same observations as made in the previous chapter about the quality of the ODG (which appears blurry and with artefacts) and PZSUM (looks relatively sharp). We may note that the adaptive deghosting does not create significant artefacts like the ODG, and appears as sharp as the PZSUM.

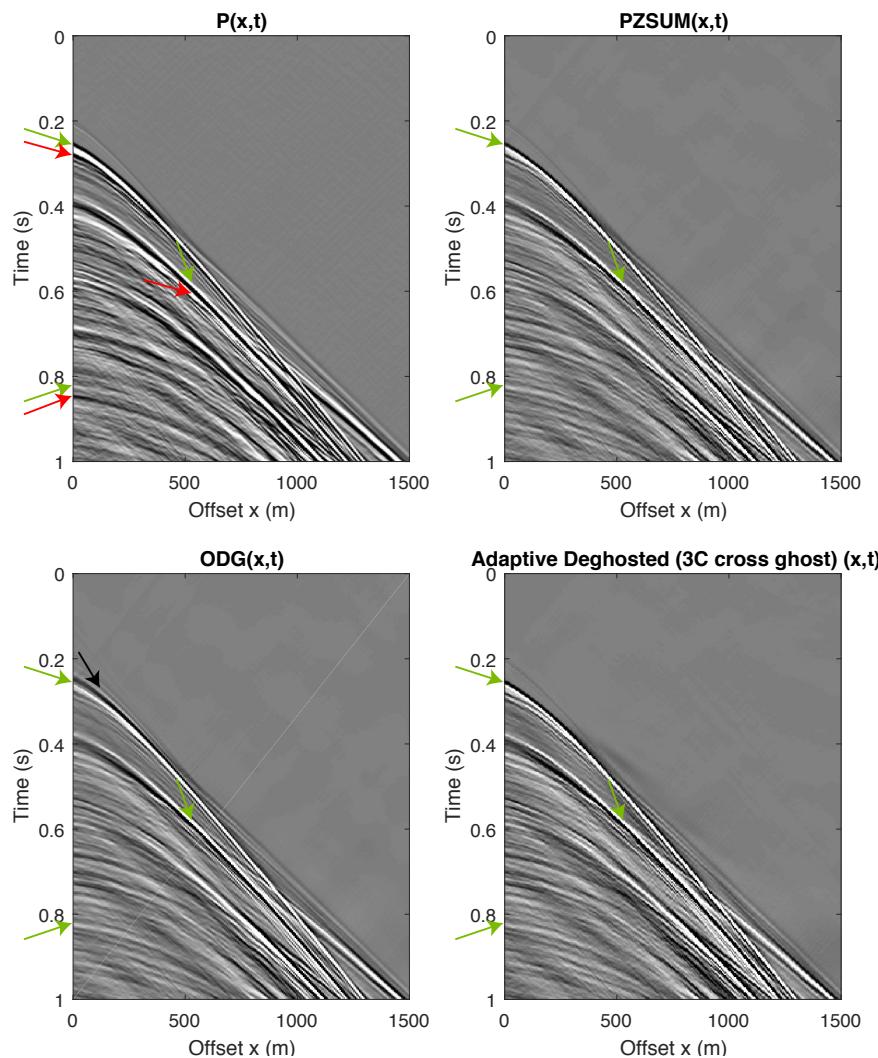


Figure 4-13: The shot gather of the real data set 1, deghosted with PZSUM, ODG and multi-sensor adaptive deghosting. The color scale has been clipped on purpose to show artefacts. A time gain of t^2 has been applied to the data. The green arrows denote some examples of the upgoing wavefield, red arrows their ghost, and the black arrow at the ODG highlights an artefact before the wavefield (the precursor).

We may then look at the spectra of the different deghosting methods (Figure 4-14). We observe a similar gain of about +6 dB in the first frequency notches of the small offset events compared to the PZSUM (panels 1, 2 and 3). We note on the longer offset (panel 4) that the data appears to contain some 2 dB more in the lower and higher parts of the frequency spectrum than the PZSUM. The ODG performs differently, and does not manage to fill the frequency notch at the first panel at all.

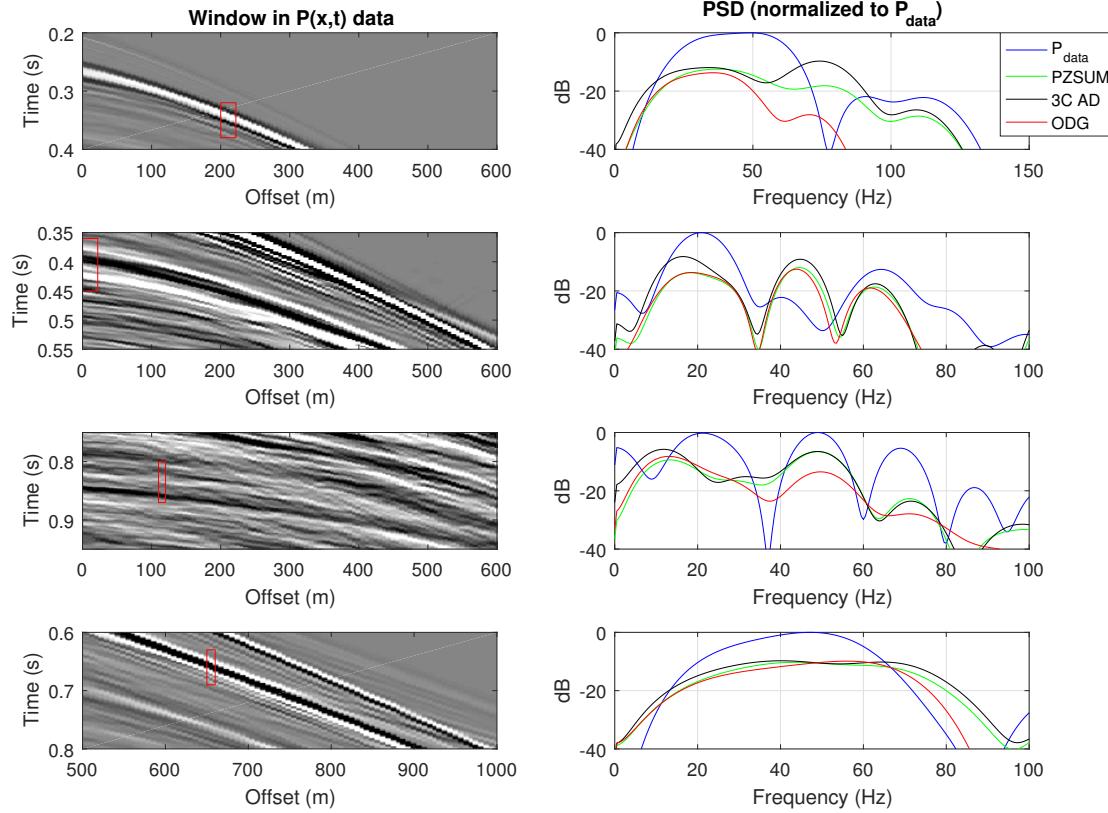


Figure 4-14: Power spectral density plot for the real data set 1 for the PZSUM, ODG and multisensor adaptive deghosting (3C AD).

We apply NMO correction to the data with a constant velocity of 1484 m/s, as in Figure 4-15. Similar to the PYZSUM before, we observe that a second reflection becomes visible directly after the sea bottom reflection. This secondary reflection event is clearly most sharp and continuous in the adaptive deghosting algorithm. We may again compute the autocorrelation on a window of the NMO corrected data, as shown in the bottom of Figure 4-15. We observe that the PZSUM and adaptive deghosting algorithm with the three component cross ghost manage to remove the most energy at the time lag of the input pressure data P . After stacking the ACF along the offset range, we observe that the ghost delay time lag is reduced mostly by the three component adaptive deghosting.

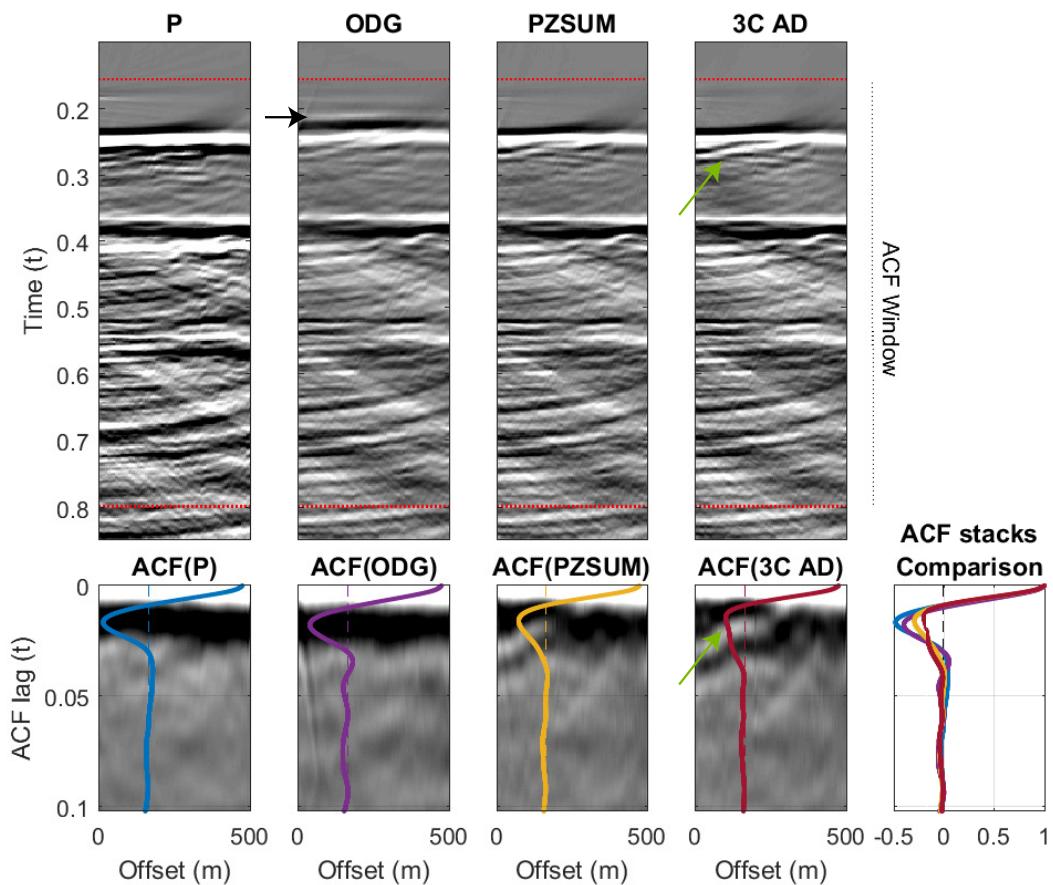


Figure 4-15: The ACF as calculated for the NMO corrected original and deghosted shots for real data 1. The red line is the stacked autocorrelation. The 3C Adaptive Deghosting (3C AD) has a smaller ghost artefact than PZSUM, and clearly performs better than the ODG. The green arrow at the 3C AD shows a second reflection just after the first sea bottom reflection, visible on the NMO corrected section and the ACF. The black arrow highlights the precursor artefact on the ODG.

4-3-2 Real data 2

The first two seconds of the deghosted shot are shown in Figure 4-16 (the full deghosted shot is shown in the appendix, Figure B-4). We may make the same observations as made in the previous chapter about the quality of the ODG (which appears blurry and with artefacts) and PZSUM (looks relatively sharp). We note that the adaptive deghosting did not create significant artefacts like the ODG, and appears as sharp as the PZSUM.

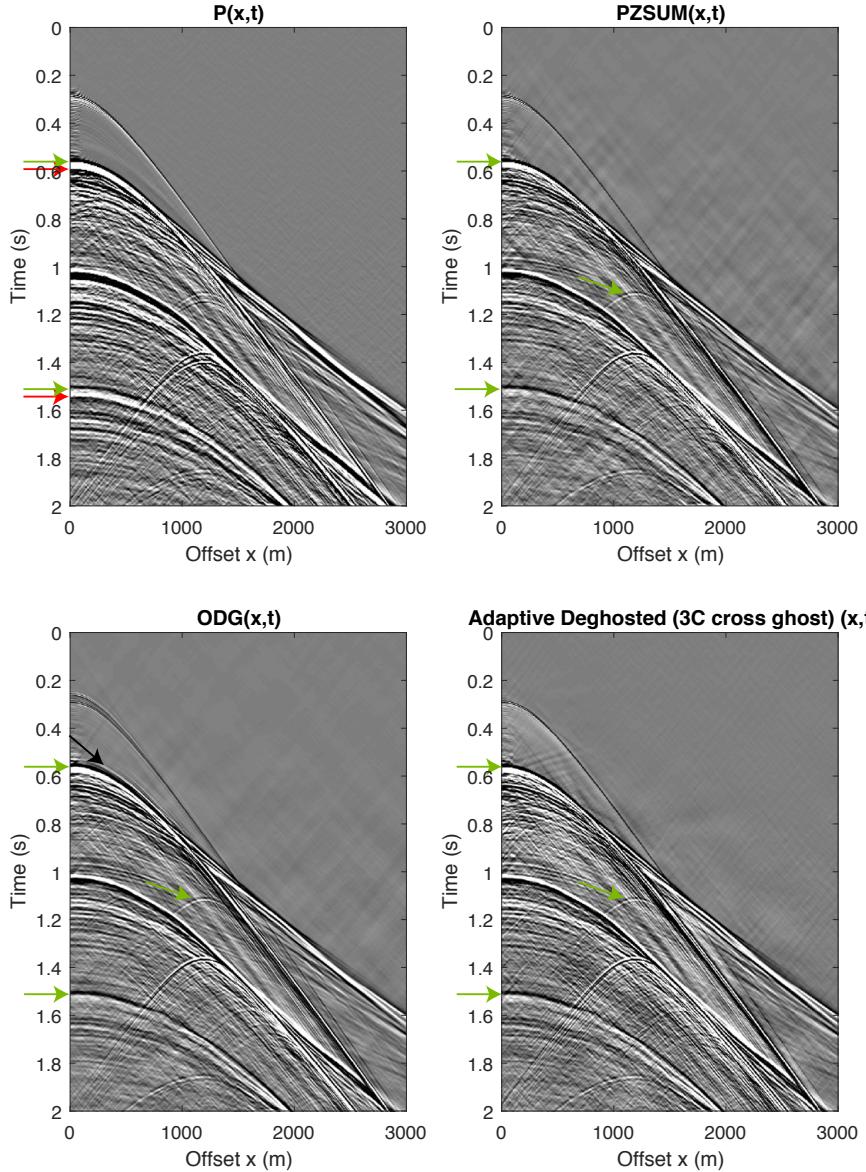


Figure 4-16: The shot gather of the real data set 2, deghosted with PZSUM, deterministic ODG and multisensor adaptive deghosted. The color scale has been clipped on purpose to show artefacts. A time gain of t has been applied. The green arrows denote some examples of the upgoing wavefield, red arrows their ghost, and the black arrow at the ODG highlights an artefact before the wavefield (the precursor).

The frequency spectra of the deghosted data are shown in Figure 4-17. We observe that the multisensor adaptive deghosting achieves a similar gain in amplitudes as observed for the PYZSUM for all near-offset windows (panel 1, 2 and 4). The larger offset panel (window 3) showed a spectrum where the PZSUM and PYZSUM showed equal results, whereas the adaptive deghosting provides a slight boost in amplitude for a number of frequencies. It is impossible to say which method shows the correct result.

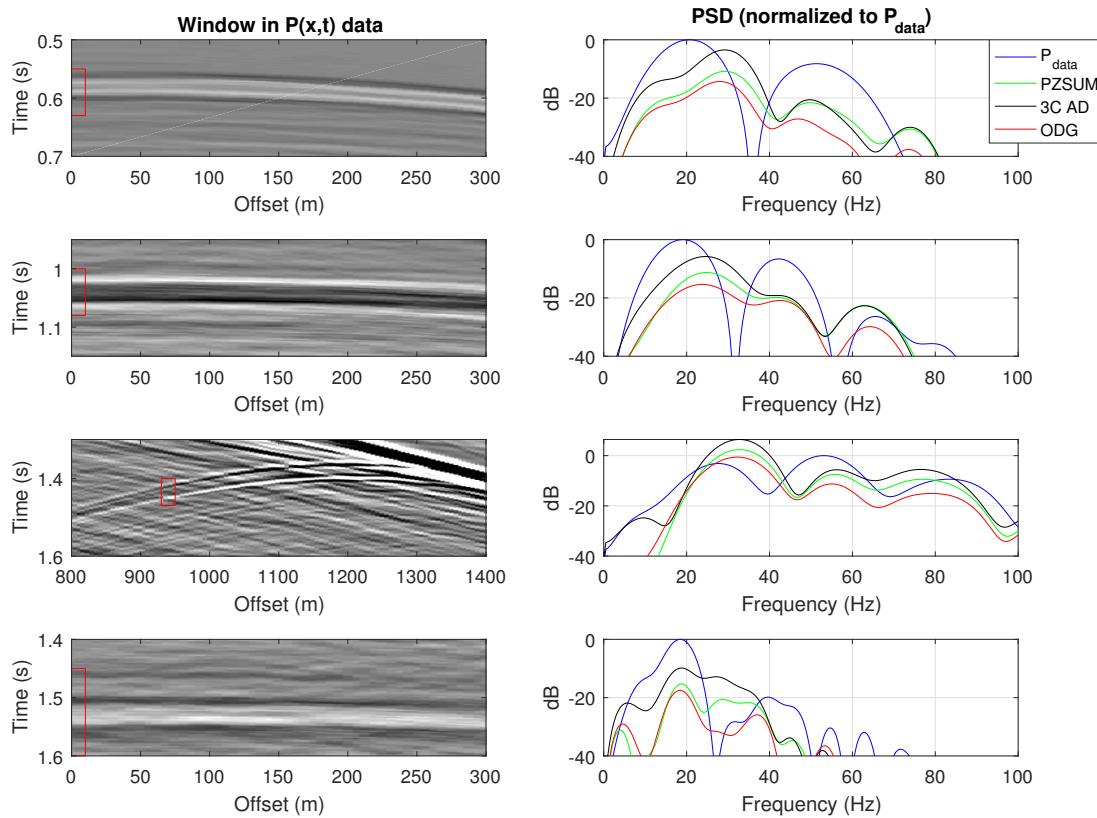


Figure 4-17: Power spectral density plot for the real data set 2 for the PZSUM, ODG and multisensor adaptive deghosting (3C AD).

Lastly, we look at the NMO correct data using a constant velocity of 1484 m/s, as in Figure 4-18. Similar to the results at the PYZSUM before, we observe that the ODG leaves artefacts in the NMO corrected window. We observe that the PZSUM and multisensor deghosting both remove the ghost on the sea-bottom reflection (0.55 s) and its multiple (1.02 s). We turn to the autocorrelation function in the bottom of the figure to make a quantitative statement on the deghosting quality. We observe that all three methods (ODG, PZSUM and three component adaptive deghosting) manage to reduce the negative value on ACF(P) at the ghost delay time, linked to the ghost. However, both ODG and PZSUM have a remaining ‘bump’ at the ghost delay time. Conversely, for the three component adaptive deghosting we find that the negative value is completely removed at the ghost delay time, suggesting complete removal of the ghost. Again, this result differs from that found with the PYZSUM, and it is not directly obvious which result is the truly correct result.

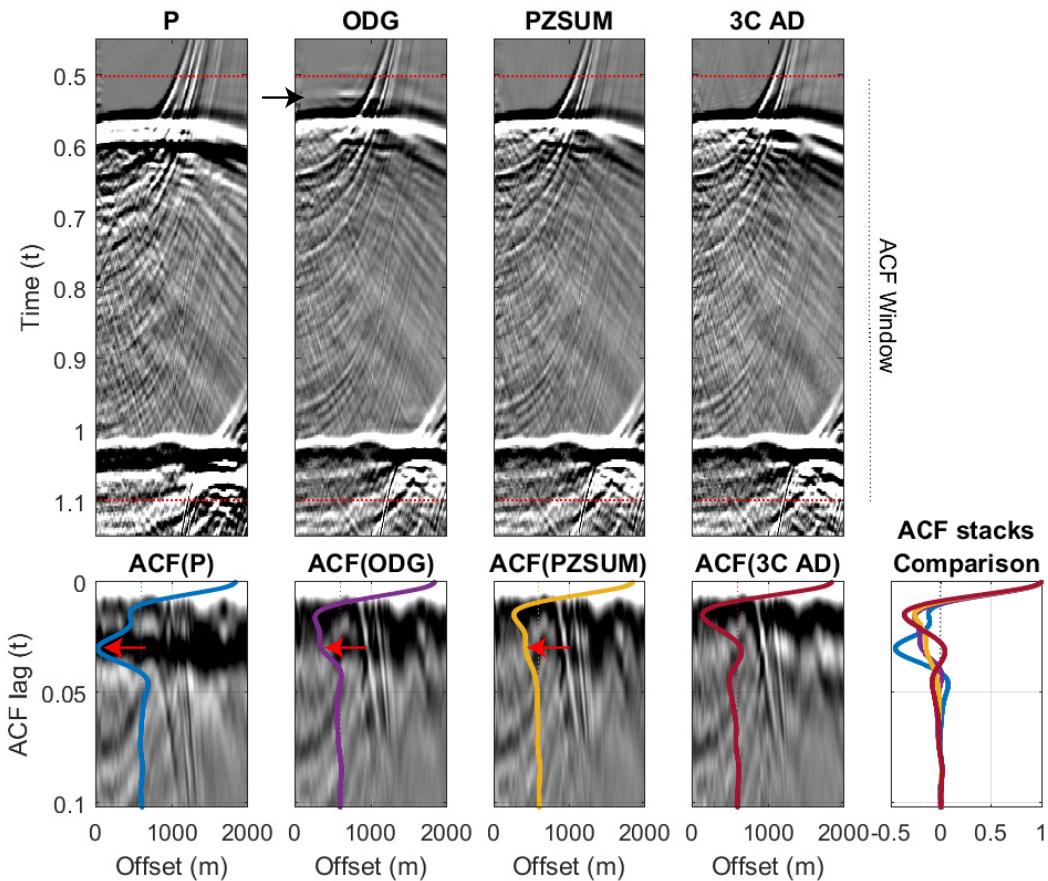


Figure 4-18: The ACF as calculated for the NMO corrected original and deghosted shots of real data 2. The red line is the stacked autocorrelation. The multisensor adaptive deghosting (3C AD) outperforms the other two deghosting methods, which show a residual ‘bump’ corresponding to the ghost (at about 29 ms).

Chapter 5

Discussion & Conclusion

The receiver ghost degrades the broadband good S/N spectrum of the upgoing wavefield. Multisensor deterministic deghosting techniques may restore the upgoing broadband spectrum, if their deghosting assumptions (such as an assumed streamer depth or incidence angle) correspond to the data. We have shown a few examples where current deterministic deghosting methods fail to recover the upgoing wavefield due to for example assuming that no energy propagates in the crossline direction. We thus propose two novel single-streamer data-driven deghosting methods, which adapt their deghosting filters based on the data itself.

The first proposed method overcomes the limitation of the PZSUM, which assumes energy only travels upwards and along the streamer, i.e. a 2D approach. We propose to obtain the 3D incidence angle of events in our data, such that we can extend the PZSUM towards also correcting for crossline (y) propagation of energy, a technique we name PYZSUM. To this purpose, we use the Time Gradient Ratio (TGR) to estimate the crossline slowness p_y based on the crossline particle velocity and pressure measurements. The TGR only works on non-overlapping events. To limit the amount of overlap, we propose to transform the multisensor measurements into the $\tau-p_x$ domain first, and perform TGR in this domain where events are more separated. Using a 2D median filter we improve the S/N of the obtained p_y , by removing outliers while preserving coherent events. These steps assign an inline and crossline slowness to each sample in the $\tau-p_x$ domain, allowing computation of the 3D incidence angle for all events in the data. We can then combine the pressure and scaled vertical particle velocity measurements in the $\tau-p_x$ domain using the 3D incidence angles, the PYZSUM, and transform the deghosted shot back to the time domain.

Testing on synthetic data showed that remaining overlap of events in the $\tau-p_x$ domain invalidated the ‘single event’ requirement of the TGR. We proposed an additional step to separate events, using overlapping tapered windows moving over the data, applying PYZSUM on each window individually. The data was thus separated in windows of $x-t$, and then further separated in windows of $\tau-p_x$. After these steps, we assumed no overlapping events remained in the $\tau-p_x$ domain. The PYZSUM in this form performed as good as, or better than, the PZSUM in deghosting synthetic and real data. The real data set contained little crossline

energy propagation, such that the improvement brought by the PYZSUM was a minor, but noticeable improvement over the deterministic deghosting methods.

Future work on the PYZSUM should focus on improving the S/N characteristics of the TGR result. Such work could study alternative filters to the 2D median filter, to increase the S/N of the TGR output. The current 2D median filter in the $\tau-p_x$ domain favors broad, coherent events, causing it to neglect sharp linear events (which map to a point in the $\tau-p_x$ domain). Additionally, one could explore filters that make the input to the TGR less noisy. We experimented with low-cut frequency filters, leading to poor results on real data, because the low-cut broadened the wavelet, and then caused overlap of events – which invalidated the single event assumption of the TGR.

The second proposed data-driven method is an extension of single sensor (cf. single sensor energy minimization, Schuberth (2015)) to multisensor adaptive deghosting. The energy minimization technique tries to estimate the ghost model (the reflection coefficient $|r_0|$ and the ghost time delay t_Δ) from the data. Once the ghost model of the data is accurately found, we can deghost the data with any deghosting techniques that depend on this ghost model. However, the energy minimization approach cannot estimate a ghost delay smaller than the reciprocal of the usable bandwidth, and may pick wrong ghost delay times due to multiple possible ‘minimum energy’ solutions. The use of multisensor measurements removes both of these limitations by employing a multisensor least-squares cost function called the ‘cross ghost’. This cross ghost is used to estimate the ghost model from the multisensor data through trying a range of potential ghost models. We show analytically that the global minimum of this cost function corresponds to the true ghost model. However, the cross ghost cost function adds an additional unknown to the ghost model estimation: beside t_Δ and $|r_0|$ we also need to estimate the vertical slowness p_z . We show that $|r_0|$ has a minor effect on the deghosting result and cross ghost cost function, thus may be assumed a constant. To remain a fully adaptive method, we assume no relation upfront between the remaining two parameters, t_Δ and p_z . We evaluate the cross ghost cost function for a range of plausible values for t_Δ and p_z (a 2D line search), and pick their global minimum as the ghost model that best describes the data. The cross ghost may be restricted to the pressure and vertical particle velocity only, but is made more robust by including the crossline particle velocity.

We can perform the 2D line search per trace to find an optimal ghost model (t_Δ , p_z and if desired $|r_0|$). The method thus favors the $\tau-p_x$ domain, where a single p_x trace corresponds to approximately a single delay time and vertical slowness. To allow for a varying ghost model in space and time, we use tapered overlapping, moving windows in the $x-t$ domain. We then deghost the data using the obtained ghost model, using for example PZSUM or ODG. A limitation of using the $\tau-p_x$ transform is that the up- and downgoing waves may map to different p_x under rough sea conditions, complicating the time delay search. Nevertheless, the adaptive algorithm generated excellent results in deghosting two real data sets. The method outperforms both the PZSUM and ODG in synthetic and real data.

Future work on multisensor adaptive deghosting may study ways to speed up the sampling of the cost function, rather than employing a 2D line search for all t_Δ and p_z . One could study stochastic methods to optimize this search. We performed tests to turn the 2D line search into a series of 1D line searches. We show that t_Δ and p_z relate as $t_\Delta = 2h_{\text{true}}p_z$, and may use an estimate of the streamer depth h to write $p_z(t_\Delta) = \frac{t_\Delta}{2h_{\text{nominal}}}$. We may then do a 1D line search along the time axis to find a minimum. However, if h is wrong, we will use the

wrong p_z and may correspondingly find a local minimum rather than the global minimum of the cost function (i.e. Figure 4-4a). More research is required to show if an iterative use of the 1D line search could converge to the global minimum.

In conclusion, we have shown two data-driven multisensor deghosting methods. We found that these methods perform as good as deterministic deghosting techniques when the assumptions of deterministic deghosting techniques are met. The data-driven deghosting techniques are superior to the deterministic deghosting techniques when their 2D energy propagation assumption is not met, as illustrated on real data.

Bibliography

- Ahmed, E. S. A., Elatif, R. E., & Alser, Z. T. (2015). Median filter performance based on different window sizes for salt and pepper noise removal in gray and rgb images. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 8(10), 343–352.
- Amundsen, L. (1993). Wavenumber-based filtering of marine point-source data. *Geophysics*, 58(9), 1335–1348.
- Caprioli, P., Özdemir, A., Özbek, A., Kragh, J., Van Manen, D., Christie, P., & Robertsson, J. (2012). Combination of multi-component streamer pressure and vertical particle velocity-theory and application to data. In *74th EAGE Conference and Exhibition incorporating EUROPEC 2012*.
- Day, A., Klüver, T., Söllner, W., Tabti, H., & Carlson, D. (2013). Wavefield-separation methods for dual-sensor towed-streamer data. *Geophysics*, 78(2), WA55–WA70.
- Fortini, C. & Vassallo, M. (2012). Estimate of 3d angle of incidence in presence of conflicting events. Technical report, WesternGeco.
- Kamil, Y. I. & Caprioli, P. (2014). Optimal deghosting robust to nonstationary noise from multimeasurement streamer data. In *2014 SEG Annual Meeting*. Society of Exploration Geophysicists.
- Kamil, Y. I., El Yadari, N., & Vassallo, M. (2014). Bayesian deghosting approach for multimeasurement streamer data. In *2014 SEG Annual Meeting*. Society of Exploration Geophysicists.
- LeVeque, R. J. (1992). *Numerical methods for conservation laws* (2nd ed.). Lectures in Mathematics. Basel: Birkhauser.
- Liu, Y. & Fomel, S. (2011). Seismic data interpolation beyond aliasing using regularized nonstationary autoregression. *Geophysics*, 76(5), V69–V77.
- Mallat, S. G. & Zhang, Z. (1993). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on signal processing*, 41(12), 3397–3415.

- Mallick, S. & Frazer, L. N. (1987). Practical aspects of reflectivity modeling. *Geophysics*, 52(10), 1355–1364.
- Mayhan, J. D. & Weglein, A. B. (2013). First application of Greens theorem-derived source and receiver deghosting on deep-water Gulf of Mexico synthetic (SEAM) and field data. *Geophysics*, 78(2), WA77–WA89.
- Özbek, A., Vassallo, M., Özdemir, K., van Manen, D.-J., & Eggenberger, K. (2010). Crossline wavefield reconstruction from multicomponent streamer data: Part 2Joint interpolation and 3D up/down separation by generalized matching pursuit. *Geophysics*, 75(6), WB69–WB85.
- Ozdemir, A., Caprioli, P., Ozbek, A., & Robertsson, J. (2009). System and technique to estimate physical propagation parameters associated with a seismic survey. US Patent App. 11/779,797.
- Özdemir, A. K. & Özbek, A. (2010). Method for optimal wave field separation. US Patent 7,676,327.
- Phinney, R. A. (1965). Theoretical calculation of the spectrum of first arrivals in layered elastic mediums. *Journal of Geophysical Research*, 70(20), 5107–5123.
- Posthumus, B. (1993). Deghosting using a twin streamer configuration. *Geophysical Prospecting*, 41(3), 267–286.
- Ravasi, M., Vassallo, M., & van Manen, D. (2011). Estimate and use of local slopes with multi-component data. Technical report, WesternGeco.
- Rentsch-Smith, S., Caprioli, P., Vassallo, M., & Van Manen, D.-J. (2013). Using crossline measurement data for an action relating to survey of a target structure. US Patent App. 13/855,610.
- Rickett, J., van Manen, D., Loganathan, P., & Seymour, N. (2014). Slanted-streamer Data-adaptive Deghosting with Local Plane Waves. In *76th EAGE Conference and Exhibition 2014*.
- Røsten, T., van Manen, D. J., Amundsen, L., & Robertsson, J. (2002). Optimal non-recursive and recursive spatial filters for de-multiple of OBS data. In *2002 SEG Annual Meeting*. Society of Exploration Geophysicists.
- Schuberth, M. G. (2015). Adaptive Deghosting of Seismic Data: A Power-Minimization Approach. Master's thesis, ETH Zrich.
- Turner, G. (1990). Aliasing in the tau-p transform and the removal of spatially aliased coherent noise. *Geophysics*, 55(11), 1496–1503.
- Vassallo, M., Özbek, A., Özdemir, K., & Eggenberger, K. (2010). Crossline wavefield reconstruction from multicomponent streamer data: Part 1Multichannel interpolation by matching pursuit (MIMAP) using pressure and its crossline gradient. *Geophysics*, 75(6), WB53–WB67.
- Wang, P., Nimsaila, K., et al. (2014). Fast Progressive Sparse Tau-P Transform For Regularization of Spatially Aliased Seismic Data. In *2014 SEG Annual Meeting*. Society of Exploration Geophysicists.

- Wapenaar, C. P. A. & Berkhouit, A. J. (1989). *Elastic wave field extrapolation: Redatuming of single-and multi-component seismic data*, volume 2. Elsevier.
- Yilmaz, Ö. (2001). *Seismic data analysis*, volume 1. Society of Exploration Geophysicists. Tulsa.
- Zhang, Z., Wu, Z., Wang, B., & Ji, J. (2015). Time Variant De-Ghosting and its Applications in WAZ Data. In *2015 SEG Annual Meeting*. Society of Exploration Geophysicists.
- Zhou, B. & Greenhalgh, S. A. (1994). Linear and parabolic τ -p transforms revisited. *Geophysics*, 59(7), 1133–1149.

Appendix A

Derivations

A-1 Inverse ghost filter

Figure A-1 schematically shows the working of the inverse ghost filter. This to illustrate that the ringing is an inherent feature of the inverse filtering process. It also shows how the filter manages to preserve the phase information, and may provide some intuition for a signal processing view of the ghost.

We may write the following equally valid reformulations of the stabilized inverse filter:

$$P_{\text{up}} = \frac{G_p^*}{|G_p|^2 + \varepsilon} P \quad (\text{A-1})$$

$$= P - \underbrace{\frac{G_p|r_0|}{|G_p|^2 + \varepsilon} P}_{P_{\text{down}}} \quad (\text{A-2})$$

$$= \frac{1}{2} \left(\underbrace{\left(\frac{G_p^*}{|G_p|^2 + \varepsilon} \right) P}_{P_{\text{up}}} + P - \underbrace{\left(\frac{G_p|r_0|}{|G_p|^2 + \varepsilon} \right) P}_{P_{\text{down}}} \right) \quad (\text{A-3})$$

$$= \frac{1}{2} \left(P + \frac{1 - |r_0| + |r_0|^2 e^{-2\pi ift_\Delta} - |r_0| e^{+2\pi ift_\Delta}}{|G_p|^2 + \varepsilon} P \right) \quad (\text{A-4})$$

The SSAD (Rickett et al., 2014) formulation uses the style of Equation (A-3), because the up- and downgoing waves are potentially ‘too’ sparse and therefore lose small portions of the signal. By combining the sparse results with the input data, we regain the original S/N characteristics of the signal. Note how Equation (A-4) resembles the PZSUM formulation.

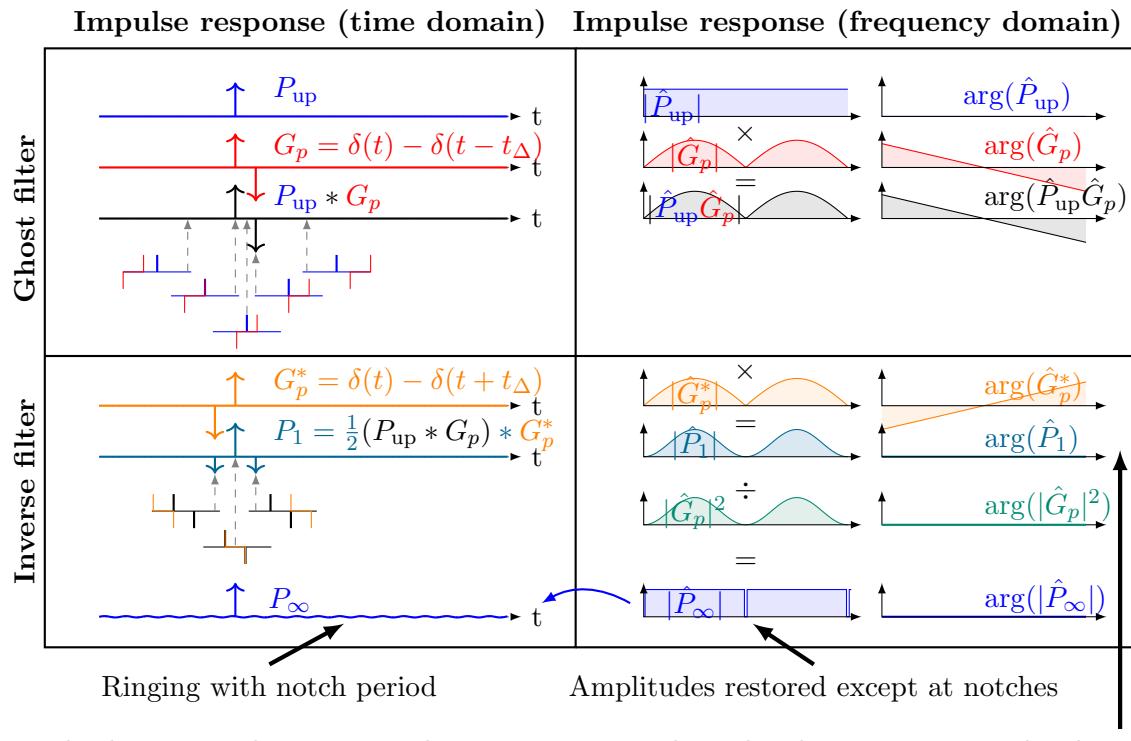


Figure A-1: An impulse response illustration of the forward and inverse ghost filter. The resulting approximation of the upgoing wavefield, P_∞ , can be considered as a weighted repeated application of GG^* (which preserves zero phase) to the initial filter. This preserves the impulse, but copies its own ghost artefacts over the entire trace.

A-2 Complex frequency technique

The complex frequency technique is given in [Mallick & Frazer \(1987\)](#). We use it to regularize divisions by a complex pole. We write $P = Pe^{-at}e^{at}$, and do the processing on only the portion Pe^{-at} . The Fourier transform is as follows:

$$\widehat{Pe^{-at}}(f) = \int_{-\infty}^{\infty} Pe^{-at} e^{-2\pi ift} dt, \quad (\text{A-5})$$

$$= \int_{-\infty}^{\infty} Pe^{-2\pi i(f - \frac{ai}{2\pi})t} dt. \quad (\text{A-6})$$

Such that we have a complex frequency $\hat{f} = f - \frac{ai}{2\pi}$ and must adapt our filters to also use \hat{f} rather than f .

A-3 Mirror-sum & Posthumus

Although the literature presents the Mirror-Sum as a method separate from that proposed by [Posthumus \(1993\)](#), their mathematical description is very similar. Take the following situation

$$\begin{cases} \hat{Z} &= \rho v_p \hat{V}_z \\ \hat{G}_p &= 1 - e^{-2\pi ift_\Delta} \\ \hat{G}_z^* &= \frac{v_p k_z}{\omega} (1 + e^{-2\pi ift_\Delta}) \end{cases} \quad (\text{A-7})$$

The mirror-sum can be rewritten as shown below.

$$\hat{P}_{\text{ms}} = \frac{1}{2} \left(\hat{P}_{\text{up}} - \hat{P}_{\text{down}} e^{-2\pi ift_\Delta} \right) \quad (\text{A-8})$$

$$= \frac{1}{2} \left(\frac{1}{2} \left(\hat{P} + \frac{\omega}{v_p k_z} \hat{Z} \right) - \frac{1}{2} \left(\hat{P} - \frac{\rho \omega}{v_p k_z} \hat{Z} \right) e^{2\pi ift_\Delta} \right) \quad (\text{A-9})$$

$$= \frac{1}{4} \left(\underbrace{\left(1 - e^{2\pi ift_\Delta} \right) \hat{P}}_{\hat{G}_p^*} + \underbrace{\left(1 + e^{2\pi ift_\Delta} \right) \frac{\omega}{v_p k_z} \hat{Z}}_{\frac{\omega^2}{v_p^2 k_z^2} \hat{G}_z^*} \right) \quad (\text{A-10})$$

i.e. it is a ‘dephase and sum’ method: dephasing the downgoing wave and adding it to the upgoing wave, with the two components being weighed individually.

Posthumus’ solution is:

$$\hat{P}_{\text{ph}} = \frac{\hat{G}_p^* \hat{P} + \hat{G}_z^* \hat{Z}}{\left| \hat{G}_p \right|^2 + \left| \hat{G}_z \right|^2} = \frac{\hat{G}_p^* \hat{P} + \hat{G}_z^* \hat{Z}}{4 \left(\sin^2(ft_\Delta \pi) + \frac{k_z^2 v_p^2}{\omega^2} \cos^2(ft_\Delta \pi) \right)}. \quad (\text{A-11})$$

This method thus has the same ‘dephase and sum’ properties, however using a different denominator to scale the amplitudes on both signals simultaneously.

The two expressions are equal for vertical incidence ($k_x = k_y = 0, k_z = \frac{\omega}{v_p}$). Posthumus (1993) shows that his method manages to correctly weigh the amplitudes after applying the conjugate ghosts, while the mirror-sum does not. It should be noted that Posthumus' method is however more sensitive to using the correct ghost delay compared to the mirror-sum: an incorrect ghost model causes the appearance of ringing all over the trace, while this is less apparent for the mirror-sum method (Posthumus, 1993; Day et al., 2013).

A-4 Least squares solution

Assuming no noise, the deghosting problem with multisensor measurements is overdetermined: there is a single unknown (the upgoing wavefield) for multiple measurements. However – the model has its own unknowns. Moreover, in the presence of noise the system of equations is inconsistent (it has no exact solution). Finding the correct solution for both the noise-free and noisy case may be found by least squares minimization. The ‘ordinary least squares’ solution for the system $Ax = b$ comes from the problem:

$$\min_x \|Ax - b\|, \quad (\text{A-12})$$

the solution of which is found as

$$x = (A^* A)^{-1} A^* b \quad (\text{A-13})$$

with $*$ indicating the complex conjugate transpose, and *provided* that $(A^* A)^{-1}$ exists.

The reasoning is as follows, assume that a wrong trial solution x introduces errors e in the data, i.e. $b = Ax_{\text{wrong}} + e$. We want to minimize (‘least’) the dot product (‘squares’) of the errors:

$$J = e^* e = (b - Ax)^* (b - Ax), \quad (\text{A-14})$$

$$= b^* b - x^* A^* b - b^* A x + x^* A^* A x. \quad (\text{A-15})$$

We find the minimum by taking the derivative of this cost function J :

$$\frac{\partial J}{\partial x} = -2b^* A + 2x^* A^* A, \quad (\text{A-16})$$

which we set to 0 and solve for x

$$x = (A^* A)^{-1} A^* b. \quad (\text{A-17})$$

A-4-1 Posthumus

The solution by Posthumus (1993) can clearly be recognized in the above system, where we want to achieve:

$$\min_{P_{\text{up}}(f)} \left\| \begin{pmatrix} \hat{G}_P \\ \hat{G}_Z \end{pmatrix} \hat{P}_{\text{up}} - \begin{pmatrix} \hat{P}_{\text{data}} \\ \hat{V}_{z,\text{data}} \end{pmatrix} \right\|, \quad (\text{A-18})$$

and opt for the solution:

$$\hat{P}_{\text{up}}(f) = \frac{\hat{G}_P^* \hat{P}_{\text{data}} + \hat{G}_Z^* \hat{V}_{z,\text{data}}}{|\hat{G}_P|^2 + |\hat{G}_Z|^2}. \quad (\text{A-19})$$

A-4-2 Optimal Deghosting (ODG)

We can enhance the cost function to minimize the errors based on the knowledge we have on the noise covariance (i.e., if a certain measurement is very noisy, we allow a bigger magnitude error). The system is m (measurements) equals HP_{up} (ghost model times upgoing wavefield) plus N (noise). Using the noise covariance $C = N^*N$ as the error sensitivity, we obtain:

$$J = eC^{-1}e = (m - HP_{\text{up}})^* C^{-1} (m - HP_{\text{up}}), \quad (\text{A-20})$$

$$= m^*C^{-1}m - P_{\text{up}}^*H^*C^{-1}m - m^*C^{-1}HP_{\text{up}} + P_{\text{up}}^*H^*C^{-1}HP_{\text{up}}. \quad (\text{A-21})$$

Take the derivative of this function, and equate to 0 to find the minimum:

$$\frac{\partial J}{\partial P_{\text{up}}} = -2m^*C^{-1}H + 2P_{\text{up}}^*H^*C^{-1}H = 0, \quad (\text{A-22})$$

leading to the solution:

$$P_{\text{up}} = (H^*C^{-1}H)^{-1}H^*C^{-1}m. \quad (\text{A-23})$$

In case of uncorrelated noise, the covariance only has diagonal entries with the variances of the measurement noise, $\text{diag}(C) = (\sigma_p^2, \sigma_z^2)$. This simplifies the equation to:

$$P_{\text{up}} = \frac{\frac{H_p P}{\sigma_p^2} + \frac{H_z V_z}{\sigma_z^2}}{\frac{H_p^* H_p}{\sigma_p^2} + \frac{H_z^* H_z}{\sigma_z^2}} \quad (\text{A-24})$$

A-4-3 Robust optimal deghosting (RODG)

Kamil & Caprioli (2014) show that the cost function can be simplified with not the noise covariance but the data covariance $R = mm^* = hh^*P_{\text{up}} + N = \sigma_u^2 HH^* + C$, with σ_u^2 the spectrum of the upgoing wave, and H the true ghost model,

$$HR^{-1} = \frac{H^*C^{-1}}{1 + H^*C^{-1}H\sigma_u^2}. \quad (\text{A-25})$$

Substitution of H^*C^{-1} with $H^*R^{-1} = \frac{H^*C^{-1}}{1 + \sigma_u^2 H^*C^{-1}H}$ gives

$$P_{\text{up}} = \frac{H^*C^{-1}}{H^*C^{-1}H}m = \frac{H^*C^{-1}}{1 + \sigma_u^2 H^*C^{-1}H} \times \frac{1 + \sigma_u^2 H^*C^{-1}H}{H^*C^{-1}H}m = \frac{H^*R^{-1}}{H^*R^{-1}H}m. \quad (\text{A-26})$$

This means that we can replace the noise covariance matrix with the data covariance matrix. In case of incoherent noise, this means: Then the approximate solution to the system is:

$$P_{\text{up}} = \frac{\frac{H_p P}{|P|^2} + \frac{H_z V_z}{|V_z|^2}}{\frac{H_p^* H_p}{|P|^2} + \frac{H_z^* H_z}{|V_z|^2}}. \quad (\text{A-27})$$

A-4-4 Cross ghost cost function

Taking a look back at the cost function, we can eliminate P_{up} from the cost function, using the ODG result:

$$J = m^*C^{-1}m - P_{\text{up}}^*H^*C^{-1}m - m^*C^{-1}HP_{\text{up}} + P_{\text{up}}^*H^*C^{-1}HP_{\text{up}}, \quad (\text{A-28})$$

$$\begin{aligned} &= m^*C^{-1}m - ((H^*C^{-1}H)^{-1}H^*C^{-1}m)^* H^*C^{-1}m - m^*C^{-1}H(H^*C^{-1}H)^{-1}H^*C^{-1}m \\ &\quad + ((H^*C^{-1}H)^{-1}H^*C^{-1}m)^* H^*C^{-1}H(H^*C^{-1}H)^{-1}H^*C^{-1}m, \end{aligned} \quad (\text{A-29})$$

$$= m^*C^{-1}m - 2 \frac{|m^*C^{-1}H|^2}{H^*C^{-1}H} + \frac{m^*C^{-1}H}{H^*C^{-1}H} H^*C^{-1}H \frac{H^*C^{-1}m}{H^*C^{-1}H}, \quad (\text{A-30})$$

$$= m^*C^{-1}m - \frac{|m^*C^{-1}H|^2}{H^*C^{-1}H}. \quad (\text{A-31})$$

Assume that C only has elements on the diagonal, with $\text{diag}(C) = (\sigma_p^2, \sigma_z^2)$ the noise covariances, then expanding the system gives:

$$J = \frac{|P|^2}{\sigma_p^2} + \frac{|V_z|^2}{\sigma_z^2} - \frac{\left| \frac{P^*G_p}{\sigma_p^2} + \frac{V_z^*G_z}{\sigma_z^2} \right|^2}{\frac{|G_p|^2}{\sigma_p^2} + \frac{|G_z|^2}{\sigma_z^2}}, \quad (\text{A-32})$$

$$= \frac{|P|^2}{\sigma_p^2} + \frac{|V_z|^2}{\sigma_z^2} - \frac{\frac{|P|^2|G_p|^2}{\sigma_p^4} + \frac{|V_z|^2|G_z|^2}{\sigma_z^4} + \frac{P^*G_pV_zG_z^*}{\sigma_p^2\sigma_z^2} + \frac{V_z^*G_zPG_p^*}{\sigma_p^2\sigma_z^2}}{\frac{|G_p|^2}{\sigma_p^2} + \frac{|G_z|^2}{\sigma_z^2}}, \quad (\text{A-33})$$

$$\begin{aligned} &= \frac{\frac{|P|^2}{\sigma_p^2} \left(\frac{|G_p|^2}{\sigma_p^2} + \frac{|G_z|^2}{\sigma_z^2} \right) + \frac{|V_z|^2}{\sigma_z^2} \left(\frac{|G_p|^2}{\sigma_p^2} + \frac{|G_z|^2}{\sigma_z^2} \right)}{\frac{|G_p|^2}{\sigma_p^2} + \frac{|G_z|^2}{\sigma_z^2}} \\ &\quad - \frac{\frac{|P|^2|G_p|^2}{\sigma_p^4} + \frac{|V_z|^2|G_z|^2}{\sigma_z^4} + \frac{P^*G_pV_zG_z^*}{\sigma_p^2\sigma_z^2} + \frac{V_z^*G_zPG_p^*}{\sigma_p^2\sigma_z^2}}{\frac{|G_p|^2}{\sigma_p^2} + \frac{|G_z|^2}{\sigma_z^2}}, \end{aligned} \quad (\text{A-34})$$

$$= \frac{\frac{|P|^2|G_z|^2 + |V_z|^2|G_p|^2 - P^*G_pV_zG_z^* - V_z^*G_zPG_p^*}{\sigma_p^2\sigma_z^2}}{\frac{|G_p|^2}{\sigma_p^2} + \frac{|G_z|^2}{\sigma_z^2}}, \quad (\text{A-35})$$

$$= \frac{\frac{(G_zP - G_pV_z)(P^*G_z^* - V_z^*G_p^*)}{\sigma_p^2\sigma_z^2}}{\frac{|G_p|^2}{\sigma_p^2} + \frac{|G_z|^2}{\sigma_z^2}}, \quad (\text{A-36})$$

$$= \frac{\frac{|G_zP - G_pV_z|^2}{\sigma_p^2\sigma_z^2}}{\frac{|G_p|^2}{\sigma_p^2} + \frac{|G_z|^2}{\sigma_z^2}}. \quad (\text{A-37})$$

The function may be extended to more components, giving Equation (4-6).

A-5 Sensitivity analysis of the cross ghost

The following section analyzes the cross ghost function (Equation (4-9)) by adding one variable at a time, showing the *upper bound* of the cost function.

A-5-1 Unknown time delay

Consider we know $p'_z = p_z$ and $|r'_0| = |r_0|$. The cost function would reduce to:

$$J(t'_\Delta) = \left\| 2 \frac{p_z |r_0|}{\rho} \left(e^{-2\pi i f t'_\Delta} - e^{-2\pi i f t_\Delta} \right) \hat{P}_{\text{up}} \right\|_2^2. \quad (\text{A-38})$$

Say $t'_\Delta = t_\Delta + \Delta t_\Delta$, with Δt_Δ describing the deviation from the true ghost delay in the data. the cost function can be simplified towards:

$$J(\Delta t_\Delta) = \left\| \left(2 \frac{p_z |r_0|}{\rho} \right) \left(e^{-2\pi i f t_\Delta} \right) \left(e^{-2\pi i f \Delta t_\Delta} - 1 \right) \hat{P}_{\text{up}} \right\|_2^2, \quad (\text{A-39})$$

$$\leq \left\| \left(e^{-2\pi i f \Delta t_\Delta} - 1 \right) \right\|_2 \underbrace{\left\| 2 \frac{p_z |r_0|}{\rho} e^{-2\pi i f t_\Delta} \hat{P}_{\text{up}} \right\|_2^2}_{\text{constant}}, \quad (\text{A-40})$$

$$\leq \left\| \left(e^{-2\pi i f \Delta t_\Delta} - 1 \right) \right\|_2 \left\| 2 \frac{p_z |r_0|}{\rho} \hat{P}_{\text{up}} \right\|_2^2. \quad (\text{A-41})$$

Where we made use of the Cauchy-Schwartz theorem for the inequality in the L^2 norm, and realise that the L^2 norm of a complex exponential equals 1. We can now derive an upper bound to the cost function. We can observe that the last term is constant, and varying Δt_Δ is the only effect on the cost function. We may write down the L^2 norm of this varying part:

$$\left\| \left(e^{-2\pi i f \Delta t_\Delta} - 1 \right) \right\|_2 = \int_{f_{\min}}^{f_{\max}} \left| e^{-2\pi i f \Delta t_\Delta} - 1 \right|^2 df, \quad (\text{A-42})$$

$$= \int_{f_{\min}}^{f_{\max}} \left(2 - (e^{-2\pi i f \Delta t_\Delta} + e^{2\pi i f \Delta t_\Delta}) \right) df, \quad (\text{A-43})$$

$$= \int_{f_{\min}}^{f_{\max}} \left(2 - 2 \cos(2\pi f \Delta t_\Delta) \right) df, \quad (\text{A-44})$$

$$= 2(f_{\max} - f_{\min}) + \frac{\sin(2\pi f_{\min} \Delta t_\Delta)}{\pi \Delta t_\Delta} - \frac{\sin(2\pi f_{\max} \Delta t_\Delta)}{\pi \Delta t_\Delta} \quad (\text{A-45})$$

We can plot this function for varying f_{\max} and as a function of Δt_Δ (Figure A-2). We observe:

1. We know $\lim_{x \rightarrow 0} \frac{\sin(axt)}{\pi t} = a$. The function derived here thus evaluates to 0 for $\Delta t_\Delta \rightarrow 0$, as the sinc functions simplify to $2(f_{\min} - f_{\max})$. The *entire* cost function will thus be 0 in case the correct time delay is tested (as we consider an upper bound). This is regardless of the signal bandwidth. The limitations of the single component cost function (multiple peaks, and an minimum estimation limit) are thus completely gone.
2. Although the sinc term is ringy, we know it will ‘die out’ quickly and have little impact, leaving us with the constant first term $2(f_{\max} - f_{\min})$. As a proxy for this ‘dying out’, take the first zero crossing of the sinc function, found at $\Delta t_\Delta = \frac{1}{2(f_{\max} + f_{\min})}$. Increasing f_{\max} thus linearly pushes the baseline cost away from 0 for wrong time delays, and additionally causes the sensitivity to increase by narrowing the sinc function around $\Delta t_\Delta = 0$. (Figure would help here!!). In practice, the sensitivity is limited to the signal

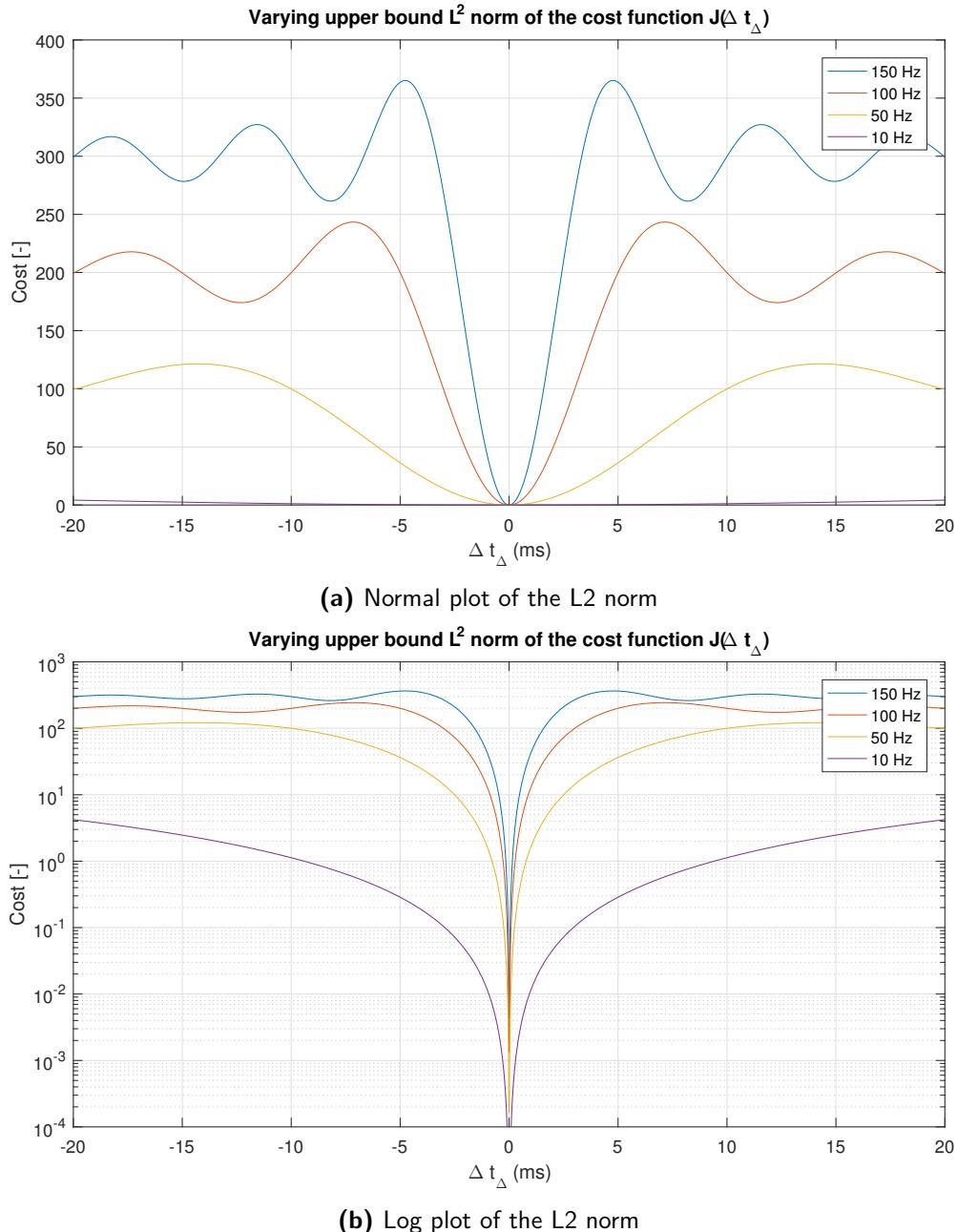


Figure A-2: The varying upper bound of the L^2 norm (Equation (A-45)), which is minimum at the correct time delay, and varies otherwise. Increasing the maximum frequency (legend) in the signal narrows the cost function around $\Delta t_\Delta = 0$. The log plot suggests that, for visual purposes, the inverse of the cost function J^{-1} will appear more striking and show the order of magnitude difference clearly.

bandwidth, so we will observe that the cost function is maximum everywhere, except for the range $(-\frac{1}{2f_{\max}}, \dots, 0, \dots, \frac{1}{2f_{\max}})$. For a signal bandwidth of $f_{\max} = 150$ Hz, this means the cost function will be around 300 everywhere, except for $\pm\frac{1}{300} = \pm 3$ ms around the true delay, where the cost function dives to 0.

To sum up: in case only the time delay is unknown, we can derive an upper bound for the cost function. This upper bound equals 0 for the true delay and is sensitive for approximately $\frac{1}{2f_{\max}}$ seconds around it. The upper bound is ‘large’ otherwise.

A-5-2 Unknown reflection coefficient

Now say both $|r'_0| = |r_0| + \Delta|r_0|$ and $t'_\Delta = t_\Delta + \Delta t_\Delta$ as the varying quantities:

$$J(\Delta t_\Delta, \Delta r_0) = \left\| \left(\frac{2p_z}{\rho} \right) \left((|r_0| + \Delta|r_0|) e^{-2\pi i f(t_\Delta + \Delta t_\Delta)} - |r_0| e^{-2\pi i f t_\Delta} \right) \hat{P}_{\text{up}} \right\|_2^2. \quad (\text{A-46})$$

We may expand:

$$J(\Delta t_\Delta, \Delta r_0) = \left\| \left(\frac{2p_z}{\rho} \right) \left((|r_0| + \Delta|r_0|) e^{-2\pi i f t_\Delta} e^{-2\pi i f \Delta t_\Delta} - |r_0| e^{-2\pi i f t_\Delta} \right) \hat{P}_{\text{up}} \right\|_2^2, \quad (\text{A-47})$$

$$= \left\| \left(\frac{2p_z |r_0|}{\rho} \right) \left(e^{-2\pi i f t_\Delta} \right) \left(\left(1 + \frac{\Delta|r_0|}{|r_0|} \right) e^{-2\pi i f \Delta t_\Delta} - 1 \right) \hat{P}_{\text{up}} \right\|_2^2, \quad (\text{A-48})$$

$$\leq \left\| \left(\left(1 + \frac{\Delta|r_0|}{|r_0|} \right) e^{-2\pi i f \Delta t_\Delta} - 1 \right) \right\|_2^2 \underbrace{\left\| \left(\frac{2p_z |r_0|}{\rho} \right) \hat{P}_{\text{up}} \right\|_2^2}_{\text{constant}}. \quad (\text{A-49})$$

We can evaluate this upper bound to find a combined cost function

$$\left\| \left(\left(1 + \frac{\Delta|r_0|}{|r_0|} \right) e^{-2\pi i f \Delta t_\Delta} - 1 \right) \right\|_2^2 = \int_{f_{\min}}^{f_{\max}} \left| \left(1 + \frac{\Delta|r_0|}{|r_0|} \right) e^{-2\pi i f \Delta t_\Delta} - 1 \right|^2 df, \quad (\text{A-50})$$

$$\begin{aligned} &= 2 \left(1 + \frac{\Delta|r_0|}{|r_0|} + \frac{\Delta|r_0|^2}{2|r_0|^2} \right) (f_{\max} - f_{\min}) + \\ &\quad \left(1 + \frac{\Delta|r_0|}{|r_0|} \right) \left(\frac{\sin(2\pi f_{\min} \Delta t_\Delta)}{\pi \Delta t_\Delta} - \frac{\sin(2\pi f_{\max} \Delta t_\Delta)}{\pi \Delta t_\Delta} \right) \end{aligned} \quad (\text{A-51})$$

Observe the close relation to the previously derived function for just Δt_Δ . We can observe:

1. If $\Delta t_\Delta = 0$ (i.e. the right time delay is found), the remaining term is $\frac{\Delta|r_0|^2}{|r_0|^2} (f_{\max} - f_{\min})$ describing the sensitivity to the correct ghost delay. Assuming the error in reflection coefficient will be approximately 5% or less, the upper bound for the norm is approximately $0.05^2 (f_{\max} - f_{\min}) = 0.0025 (f_{\max} - f_{\min})$. The order of magnitude of this cost bound is relatively small compared to that observed for the time delay which cycles around $2(f_{\max} - f_{\min})$.

2. Increasing Δt_Δ makes the sinc terms go towards 0, such that the majority of the cost function is contributed by the first term. The term $\left(1 + \frac{\Delta|t_\Delta|}{|r_0|} + \frac{\Delta|t_\Delta|^2}{2|r_0|^2}\right)$ is to first approximation equal to 1, and to second approximation equal to $1 + \frac{\Delta|t_\Delta|}{|r_0|}$. The important consequence of this is that when we are ‘far away’ from the correct Δt_Δ , the upper bound of the cost function is linear with respect to $\Delta|r_0|$, i.e., does not have a minimum at the correct $\Delta|r_0|$ at all! We can take away from this that we *can* and *must* search for the correct Δt_Δ before starting to search the correct reflection coefficient. In the ‘meantime’ we can choose an approximate reflection coefficient, e.g. $|r_0| = 0.95$.

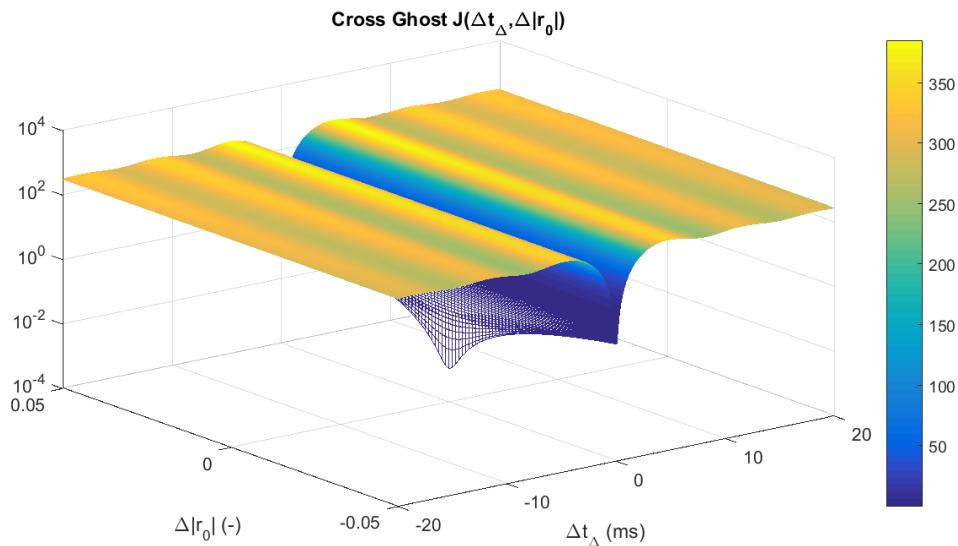


Figure A-3: Evaluating the upper bound of the L^2 norm of $J(\Delta t_\Delta, \Delta |r_0|)$, while varying both the reflection coefficient and the time delay around the true (desired) values. It's clear that the time delay has the biggest importance. Only when $\Delta t_\Delta \rightarrow 0$, we see the quadratic term in the bound become important. This motivates how we must focus on finding the correct time delay initially. Notice how the brightest yellow increases from $\Delta|r_0| = -0.05$ to $\Delta|r_0| = 0.05$, showing the linear term in the cost function for larger Δt_Δ .

A-5-3 Unknown incidence angle

Lastly, varying $p'_z = p_z + \Delta p_z$ allows us to finish the analysis. We can find an upper bound:

$$\begin{aligned}
J(\Delta t_\Delta, \Delta p_z) &= \left\| \left(\Delta p_z \left(1 - |r_0|(|r_0| + \Delta|r_0|) e^{-2\pi i f (2t_\Delta + \Delta t_\Delta)} \right) \right. \right. \\
&\quad \left. \left. + (2p_z + \Delta p_z) \left((|r_0| + \Delta|r_0|) e^{-2\pi i f (t_\Delta + \Delta t_\Delta)} - |r_0| e^{-2\pi i f t_\Delta} \right) \right) \frac{\hat{P}_{\text{up}}}{\rho} \right\|_2^2, \\
&\leq \left\| \frac{\Delta p_z}{2p_z} \left(\frac{1}{|r_0|} e^{2\pi i f t_\Delta} - (|r_0| + \Delta|r_0|) e^{-2\pi i f (t_\Delta + \Delta t_\Delta)} \right. \right. \\
&\quad \left. \left. + \left(\frac{2p_z}{\Delta p_z} + 1 \right) \left(\left(1 + \frac{\Delta|r_0|}{|r_0|} \right) e^{-2\pi i f \Delta t_\Delta} - 1 \right) \right) \right\|_2^2 \left\| \frac{2p_z|r_0|}{\rho} \hat{P}_{\text{up}} \right\|_2^2,
\end{aligned} \tag{A-52}$$

Evaluating this norm is not particularly straightforward, but still insightful.

$$\left\| \frac{\Delta p_z}{2p_z} \left(\frac{1}{|r_0|} e^{2\pi i f t_\Delta} - (|r_0| + \Delta |r_0|) e^{-2\pi i f (t_\Delta + \Delta t_\Delta)} + \left(\frac{2p_z}{\Delta p_z} + 1 \right) \left((1 + \frac{\Delta |r_0|}{|r_0|}) e^{-2\pi i f \Delta t_\Delta} - 1 \right) \right) \right\|_2^2$$

(A-54)

$$\begin{aligned} &= \int_{f_{\min}}^{f_{\max}} \left(\frac{\Delta p_z^2}{4p_z^2} \left(2 + |r_0|^2 + \Delta |r_0|^2 + 2\Delta |r_0| |r_0| + \frac{1 + \Delta |r_0|^2}{|r_0|^2} + \frac{2\Delta |r_0|}{|r_0|} \right) \right. \\ &\quad \left. + \frac{\Delta p_z}{p_z} \left(2 + \frac{2\Delta |r_0|}{|r_0|} + \frac{\Delta |r_0|^2}{|r_0|} \right) + \frac{4p_z}{\Delta p_z^2} \left(2 + \frac{2\Delta |r_0|}{|r_0|} + \frac{\Delta |r_0|^2}{|r_0|} \right) \right) df \\ &+ \int_{f_{\min}}^{f_{\max}} \left(-4 \left(1 + \frac{\Delta p_z}{p_z} + \frac{\Delta p_z^2}{4p_z^2} \right) \left(1 + \frac{\Delta |r_0|}{|r_0|} \right) \cos(2\pi f \Delta t_\Delta) \right) df \\ &+ \int_{f_{\min}}^{f_{\max}} \left(-2 \frac{\Delta p_z^2}{4p_z^2} \left(1 + \frac{\Delta |r_0|}{|r_0|} \right) \cos(2\pi f (2t_\Delta + \Delta t_\Delta)) \right) df \\ &+ \int_{f_{\min}}^{f_{\max}} \left(2 \left(\frac{\Delta p_z^2}{4p_z} + \frac{\Delta p_z}{2p_z} \right) \left(\left(\frac{1}{|r_0|} + \frac{\Delta |r_0|}{|r_0|^2} \right) \cos(2\pi f (t_\Delta + \Delta t_\Delta)) - \frac{1}{|r_0|} \cos(2\pi f t_\Delta) \right. \right. \\ &\quad \left. \left. - \left(|r_0| + 2\Delta |r_0| |r_0| + \frac{\Delta |r_0|^2}{|r_0|} \right) \cos(2\pi f (t_\Delta + 2\Delta t_\Delta)) + (|r_0| + \Delta |r_0|) \cos(2\pi f (t_\Delta + \Delta t_\Delta)) \right) \right) df \end{aligned} \quad (\text{A-55})$$

$$\begin{aligned} &= \underbrace{\left(\frac{\Delta p_z^2}{4p_z^2} \left(2 + |r_0|^2 + \Delta |r_0|^2 + 2\Delta |r_0| |r_0| + \frac{1 + \Delta |r_0|^2}{|r_0|^2} + \frac{2\Delta |r_0|}{|r_0|} \right) \right.}_{\text{constants}} \\ &\quad \left. + \frac{\Delta p_z}{p_z} \left(2 + \frac{2\Delta |r_0|}{|r_0|} + \frac{\Delta |r_0|^2}{|r_0|} \right) + \frac{4p_z}{\Delta p_z^2} \left(2 + \frac{2\Delta |r_0|}{|r_0|} + \frac{\Delta |r_0|^2}{|r_0|} \right) \right) (f_{\max} - f_{\min}) \\ &- \left(1 + \frac{\Delta p_z}{p_z} + \frac{\Delta p_z^2}{4p_z^2} \right) \left(1 + \frac{\Delta |r_0|}{|r_0|} \right) \left(\frac{\sin(2\pi f_{\min} \Delta t_\Delta)}{\pi \Delta t_\Delta} - \frac{\sin(2\pi f_{\max} \Delta t_\Delta)}{\pi \Delta t_\Delta} \right) \\ &- \frac{\Delta p_z^2}{2p_z^2} \left(1 + \frac{\Delta |r_0|}{|r_0|} \right) \left(\frac{\sin(2\pi f_{\min} (2t_\Delta + \Delta t_\Delta))}{\pi (2t_\Delta + \Delta t_\Delta)} - \frac{\sin(2\pi f_{\max} (2t_\Delta + \Delta t_\Delta))}{\pi (2t_\Delta + \Delta t_\Delta)} \right) \\ &+ \left(\frac{\Delta p_z^2}{4p_z^2} + \frac{\Delta p_z}{2p_z} \right) \left(\left(\frac{1}{|r_0|} + \frac{\Delta |r_0|}{|r_0|^2} \right) \left(\frac{\sin(2\pi f_{\min} (t_\Delta + \Delta t_\Delta))}{\pi (t_\Delta + \Delta t_\Delta)} - \frac{\sin(2\pi f_{\max} (t_\Delta + \Delta t_\Delta))}{\pi (t_\Delta + \Delta t_\Delta)} \right) \right. \\ &\quad \left. - \left(\frac{1}{|r_0|} \right) \left(\frac{\sin(2\pi f_{\min} t_\Delta)}{\pi t_\Delta} - \frac{\sin(2\pi f_{\max} t_\Delta)}{\pi t_\Delta} \right) \right. \\ &\quad \left. - \left(|r_0| + 2\Delta |r_0| + \frac{\Delta |r_0|^2}{|r_0|} \right) \left(\frac{\sin(2\pi f_{\min} (t_\Delta + 2\Delta t_\Delta))}{\pi (t_\Delta + 2\Delta t_\Delta)} - \frac{\sin(2\pi f_{\max} (t_\Delta + 2\Delta t_\Delta))}{\pi (t_\Delta + 2\Delta t_\Delta)} \right) \right. \\ &\quad \left. + (|r_0| + \Delta |r_0|) \left(\frac{\sin(2\pi f_{\min} (t_\Delta + \Delta t_\Delta))}{\pi t_\Delta + \Delta t_\Delta} - \frac{\sin(2\pi f_{\max} (t_\Delta + \Delta t_\Delta))}{\pi t_\Delta + \Delta t_\Delta} \right) \right) \end{aligned} \quad (\text{A-56})$$

We can make a simplifying choice that $r_0 = 1$ and $\Delta|r_0| = 0$, which has a relatively small effect on the overall value.

$$\begin{aligned}
 J(\Delta t_\Delta, \Delta p_z, \Delta|r_0| = 0) &= 2 \left(\frac{\Delta p_z^2}{2p_z^2} + \frac{\Delta p_z}{p_z} + 1 \right) (f_{\max} - f_{\min}) - \left(1 + \frac{\Delta p_z}{p_z} + \frac{\Delta p_z^2}{4p_z^2} \right) \left(\frac{\sin(2\pi f_{\min} \Delta t_\Delta)}{\pi \Delta t_\Delta} - \frac{\sin(2\pi f_{\max} \Delta t_\Delta)}{\pi \Delta t_\Delta} \right) \\
 &\quad - \frac{\Delta p_z^2}{2p_z^2} \left(\frac{\sin(2\pi f_{\min}(2t_\Delta + \Delta t_\Delta))}{\pi(2t_\Delta + \Delta t_\Delta)} - \frac{\sin(2\pi f_{\max}(2t_\Delta + \Delta t_\Delta))}{\pi(2t_\Delta + \Delta t_\Delta)} \right) \\
 &\quad + \left(\frac{\Delta p_z^2}{4p_z^2} + \frac{\Delta p_z}{2p_z} \right) \left(\left(\frac{\sin(2\pi f_{\min}(t_\Delta + \Delta t_\Delta))}{\pi(t_\Delta + \Delta t_\Delta)} - \frac{\sin(2\pi f_{\max}(t_\Delta + \Delta t_\Delta))}{\pi(t_\Delta + \Delta t_\Delta)} \right) \right. \\
 &\quad \quad \left. - \left(\frac{\sin(2\pi f_{\min} t_\Delta)}{\pi t_\Delta} - \frac{\sin(2\pi f_{\max} t_\Delta)}{\pi t_\Delta} \right) \right. \\
 &\quad \quad \left. - \left(\frac{\sin(2\pi f_{\min}(t_\Delta + 2\Delta t_\Delta))}{\pi(t_\Delta + 2\Delta t_\Delta)} - \frac{\sin(2\pi f_{\max}(t_\Delta + 2\Delta t_\Delta))}{\pi(t_\Delta + 2\Delta t_\Delta)} \right) \right. \\
 &\quad \quad \left. + \left(\frac{\sin(2\pi f_{\min}(t_\Delta + \Delta t_\Delta))}{\pi(t_\Delta + \Delta t_\Delta)} - \frac{\sin(2\pi f_{\max}(t_\Delta + \Delta t_\Delta))}{\pi(t_\Delta + \Delta t_\Delta)} \right) \right)
 \end{aligned} \tag{A-57}$$

The first line of this cost function bears close resemblance to the upper bound of the reflection coefficient cost function (Equation (A-51)). Ignoring other lines, the correct time delay $\Delta t_\Delta = 0$ cancels all terms except the quadratic part $\frac{\Delta p_z^2}{2p_z^2}$. The cost function however depends on five additional sinc functions. Recall that the sinc function is only of significant value when the argument is small, approximately less than $\pm\pi$. These sinc functions thus only become important when t_Δ is small. The cross ghost was evaluated for varying p_z and Δt_Δ in Figure A-4, showing that for a p_z that is 25% too small, and a time delay lower than 5 ms, the combination of the two errors particularly broadens the cost function. The function looks somewhat like the function we could expect above, strengthening some belief that the upper bound is a good approximation of the cost function.

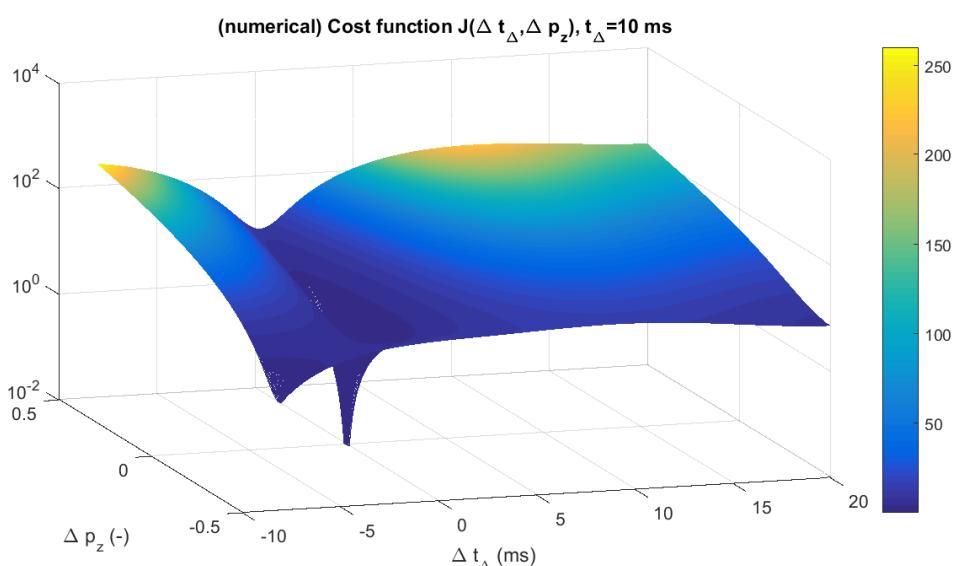


Figure A-4: Numerical evaluation of the cross ghost cost function for varying Δt_Δ and p_z , using a 40 Hz central frequency Ricker wavelet, a nominal $p_z = 0.5$ and a ghost delay of 10 ms. We observe additional minima forming for $\Delta p_z \leq 0$ at $t_\Delta + \Delta t_\Delta = 0$ and $t_\Delta + 2\Delta t_\Delta$ – both terms are present in the derived cost function.

Appendix B

Additional images

B-1 Full deghosted shots

B-1-1 Real data 1

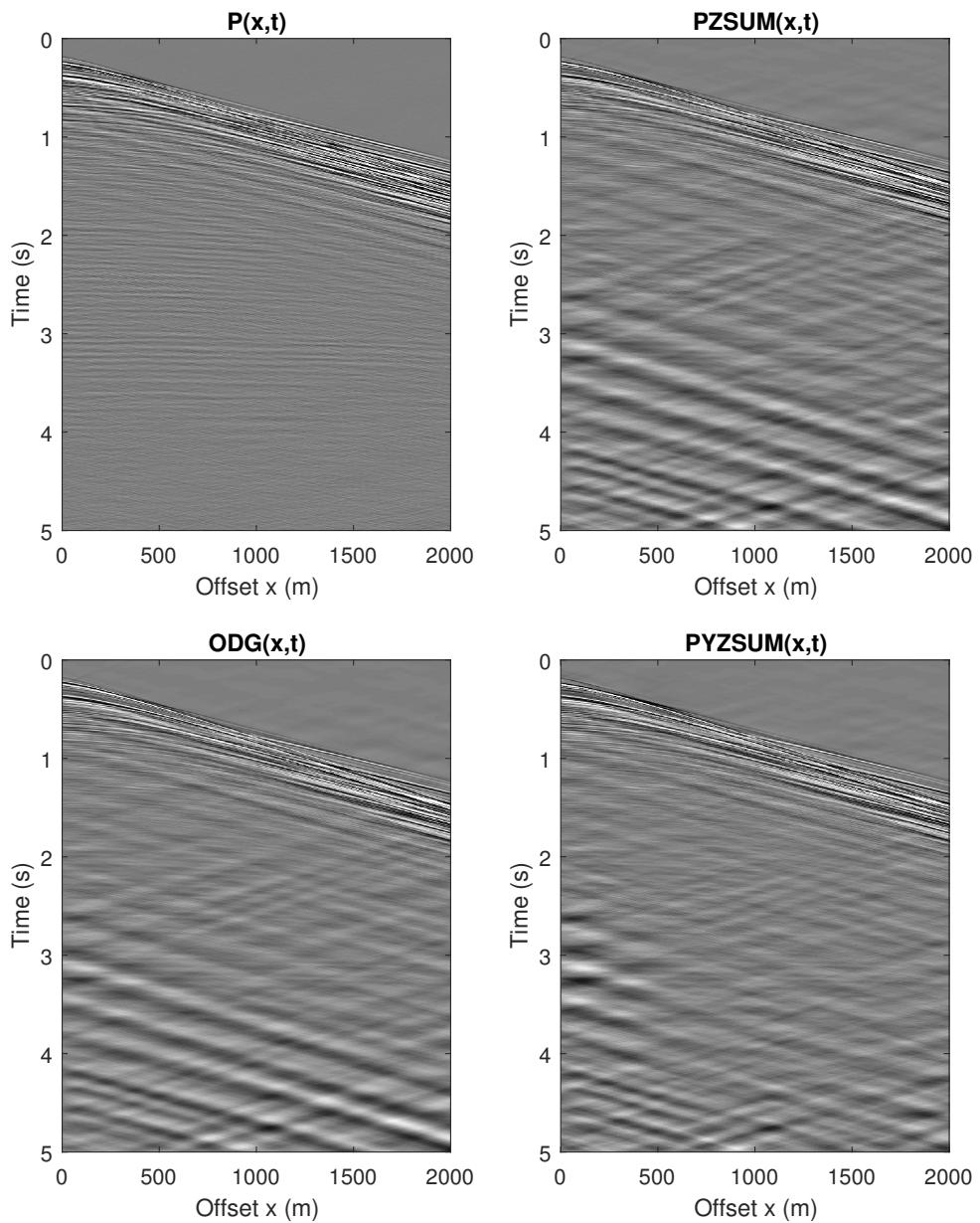


Figure B-1: Full original and deghosted shot for PYZSUM for real data set 1, with gain of t applied.

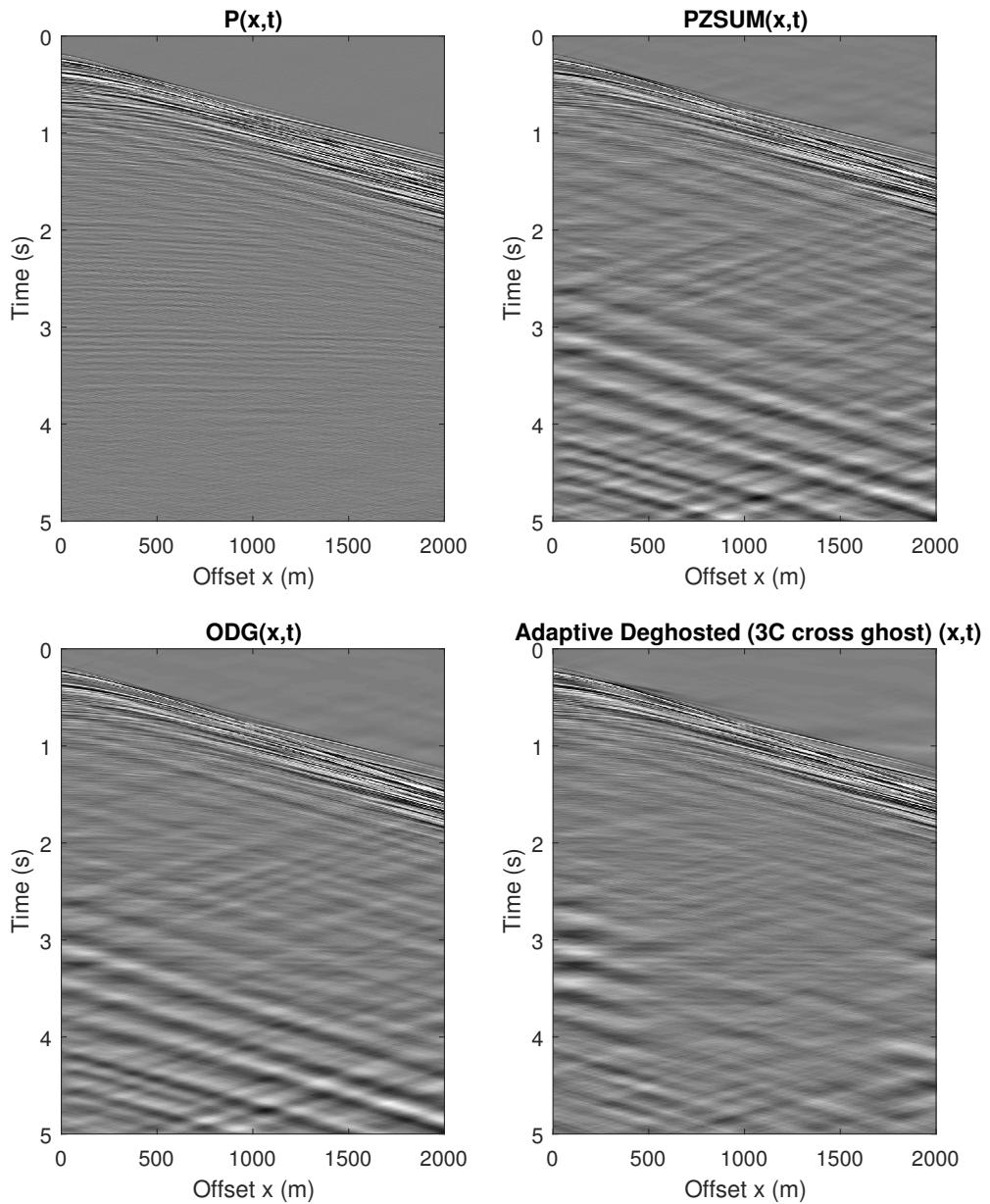


Figure B-2: Full original and deghosted shot for the 3 component cross ghost adaptive deghosting for real data set 1, with gain of t applied.

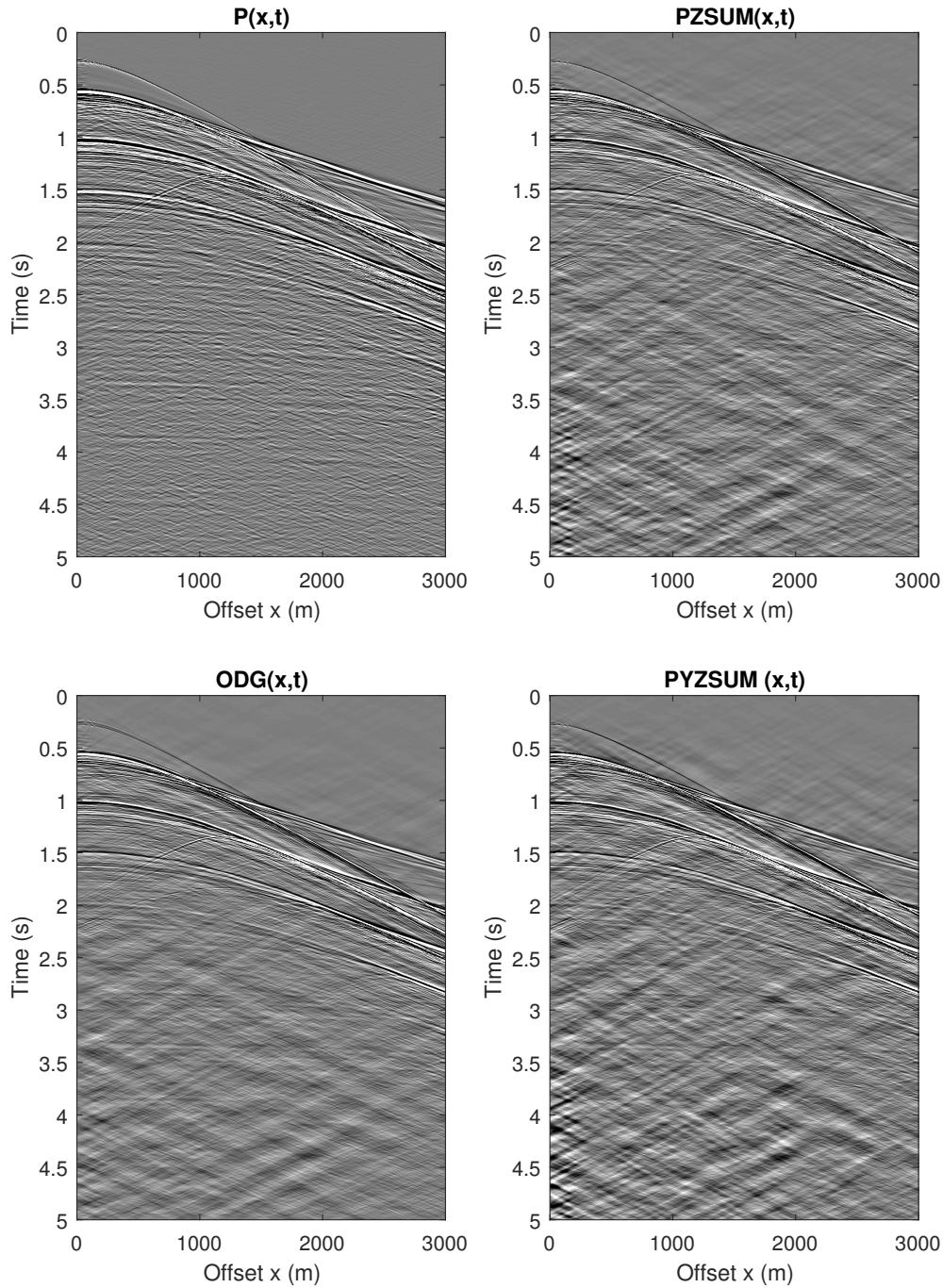
B-1-2 Real data 2

Figure B-3: Full original and deghosted shot for PYZSUM for real data set 2, with gain of t applied.

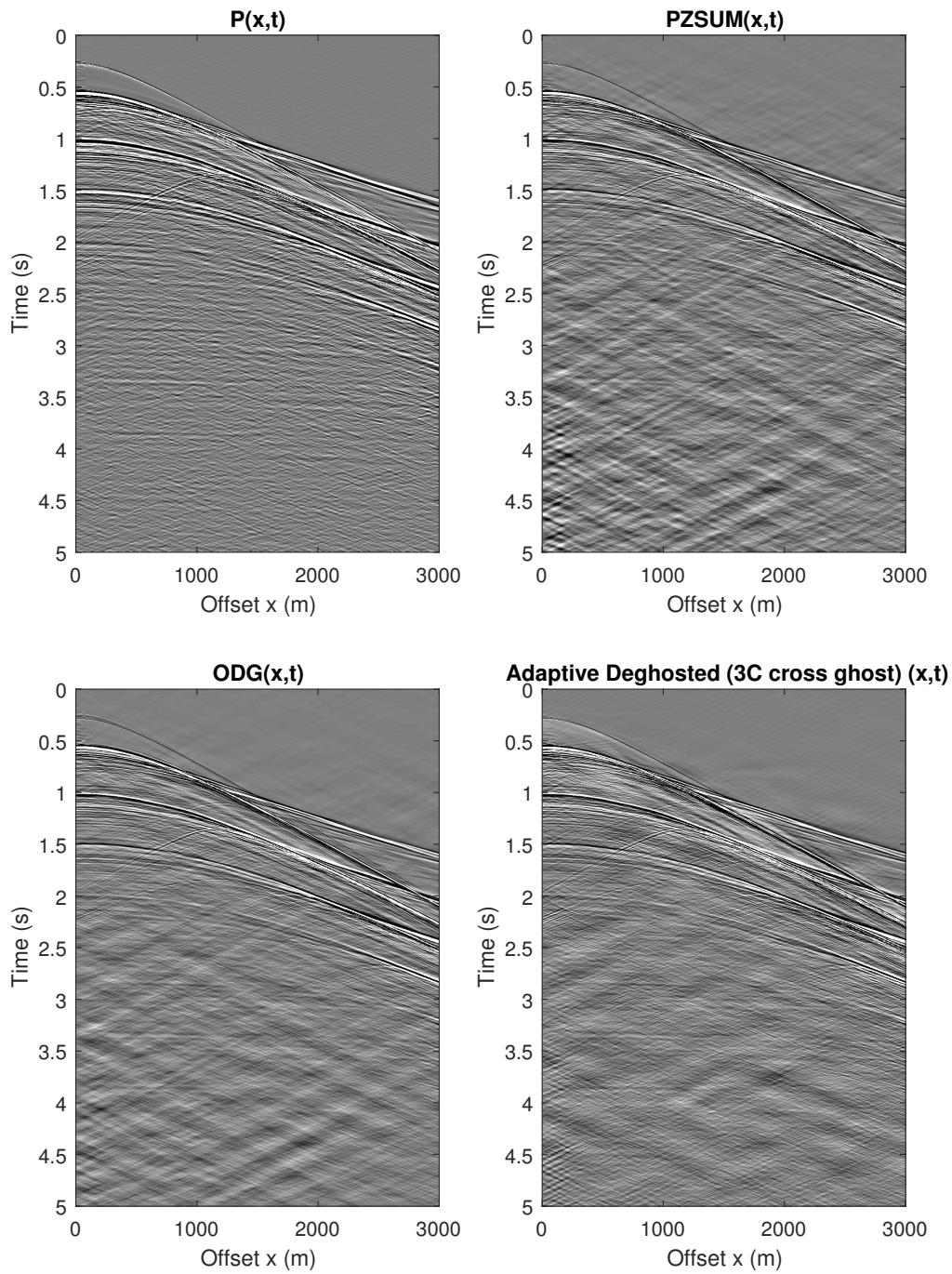


Figure B-4: Full original and deghosted shot for PYZSUM for real data set 2, with gain of t applied.

B-2 Cost functions at other time delays

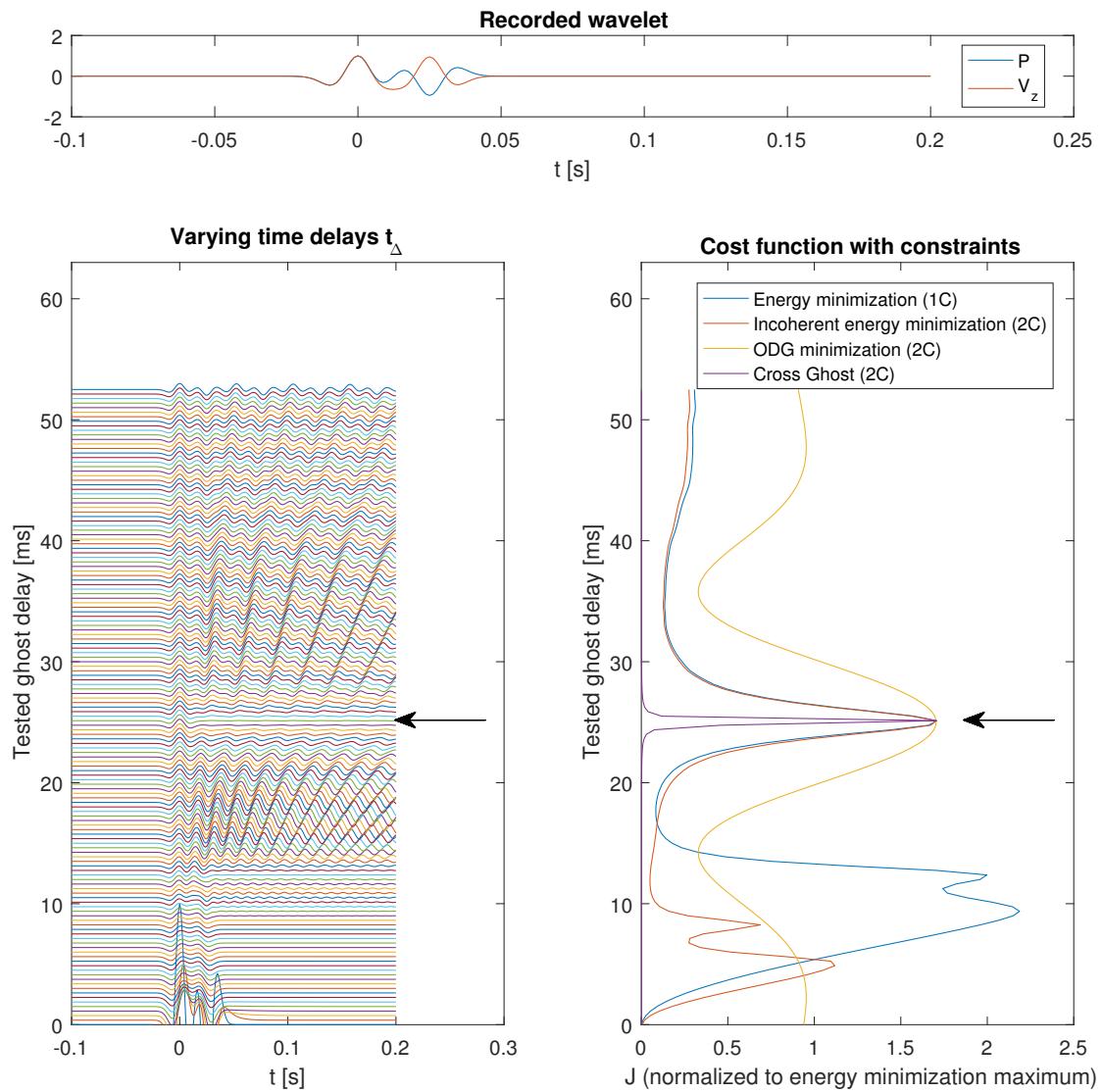


Figure B-5: The cost functions at a time delay of 25 ms, with a maximum frequency of 100 Hz.

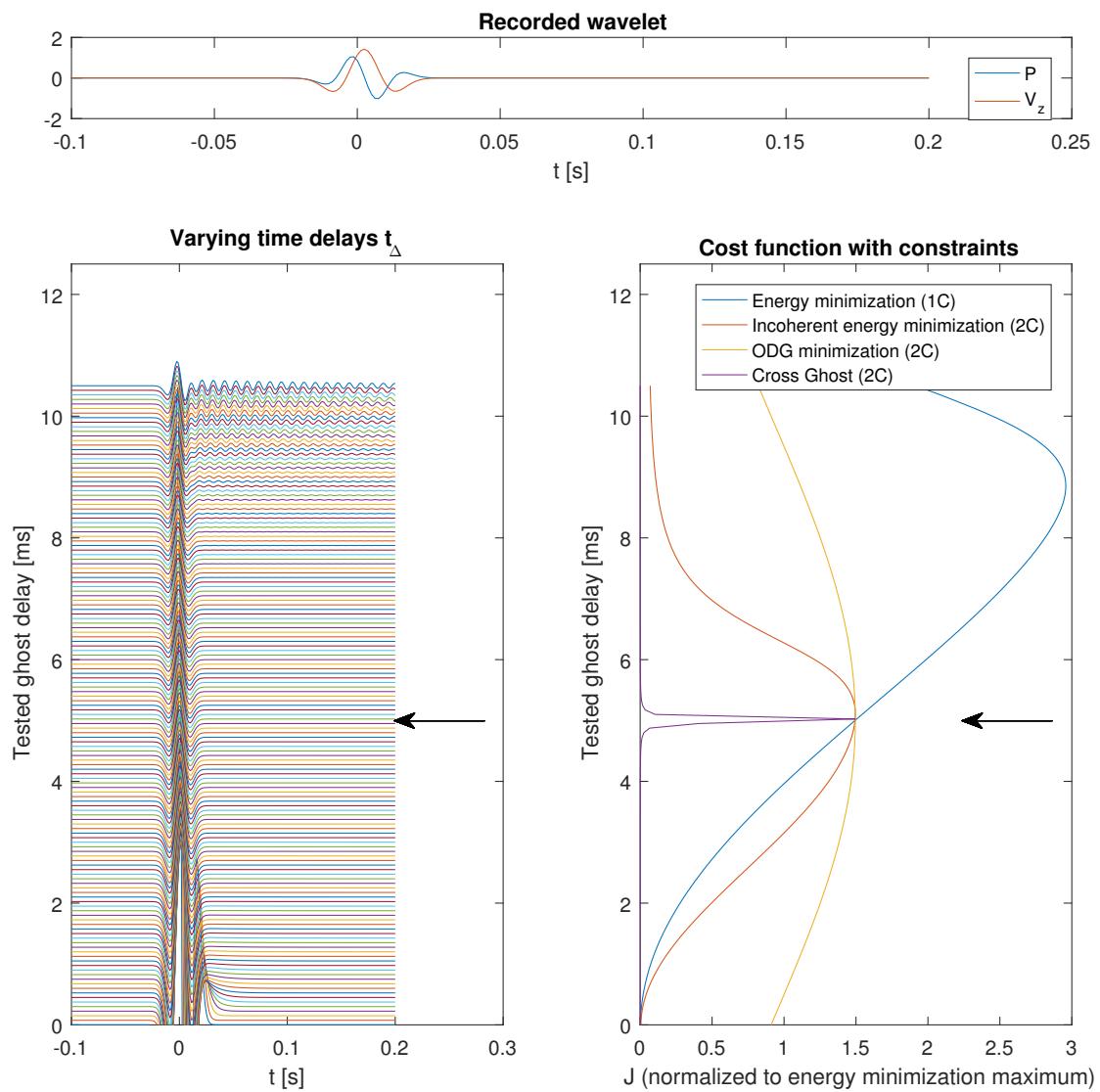


Figure B-6: The cost functions at a time delay of 5 ms, with a maximum frequency of 100 Hz. Clearly, the energy minimization has reached its estimation limit and is not useful. The 2C energy minimization just barely has a peak at the right point, but this is its estimation limit too.

