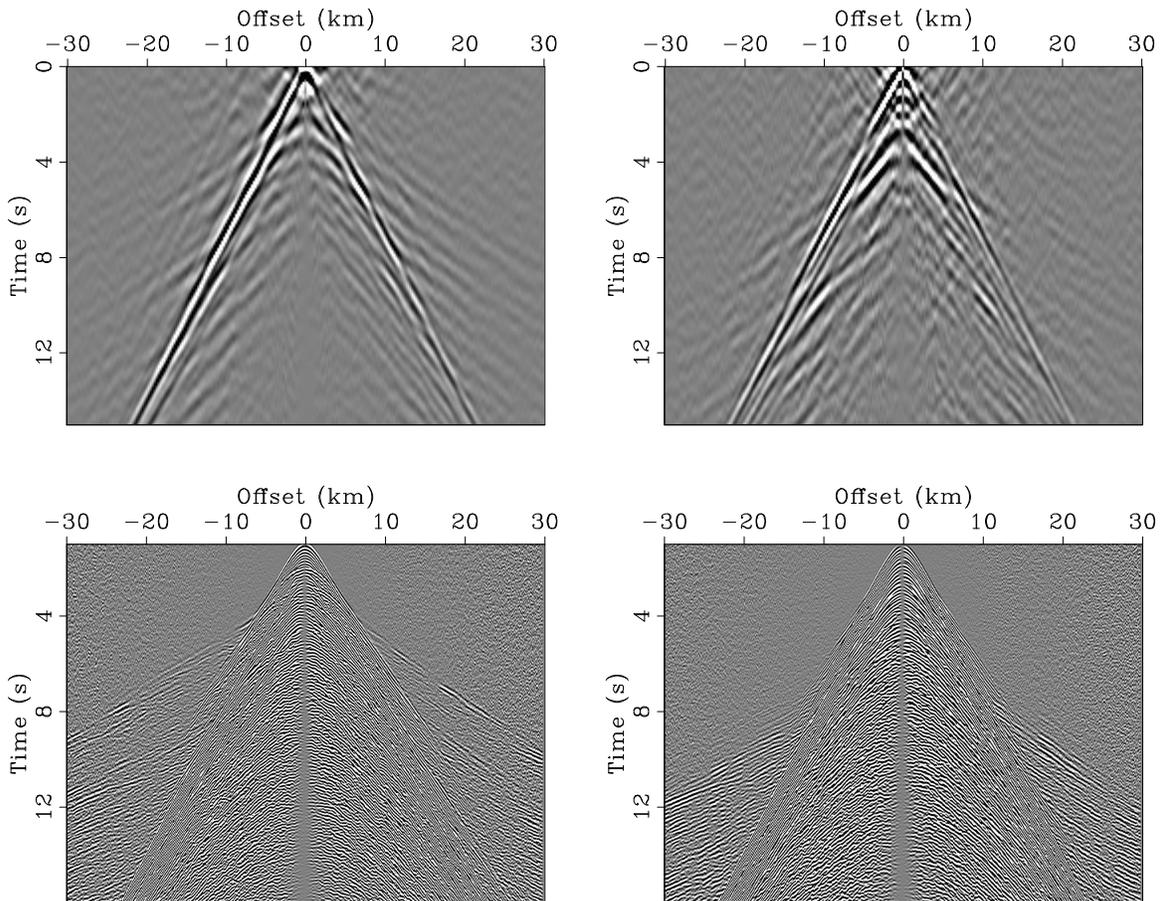


STANFORD EXPLORATION PROJECT

*Rustam Akhmadiev, Guillaume Barnier, Biondo Biondi, Ettore Biondi,
Alejandro Cabrales-Vargas, Jason Chang, Jon Claerbout, Robert Clapp, Steve Cole,
Taylor Dahlke, Nader Dutta, Gabriel Fabien-Ouellet, Stuart Farris, Joseph Jennings,
Huy Le, Stewart Levin, Eileen Martin, Tapan Mukerji, Anshuman Pradhan, Shuki Ronen,
Rahul Sarkar, Kaiwen Wang, and Siyuan Yuan*

Report Number 170, September 2017



Copyright © 2017

by the Board of Trustees of the Leland Stanford Junior University

Copying permitted for all internal purposes of the Sponsors of Stanford Exploration Project

Preface

The electronic version of this report¹ makes the included programs and applications available to the reader. The markings [ER], [CR], and [NR] are promises by the author about the reproducibility of each figure result. Reproducibility is a way of organizing computational research that allows both the author and the reader of a publication to verify the reported results. Reproducibility facilitates the transfer of knowledge within SEP and between SEP and its sponsors.

ER denotes Easily Reproducible and are the results of processing described in the paper. The author claims that you can reproduce such a figure from the programs, parameters, and makefiles included in the electronic document. The data must either be included in the electronic distribution, be easily available to all researchers (e.g., SEG-EAGE data sets), or be available in the SEP data library². We assume you have a UNIX workstation with Fortran, Fortran90, C, C++, X-Windows system and the software downloadable from our website (SEP makerules, SEPlib, and the SEP latex package), or other free software such as SU. Before the publication of the electronic document, someone other than the author tests the author's claim by destroying and rebuilding all ER figures. Some ER figures may not be reproducible by outsiders because they depend on data sets that are too large to distribute, or data that we do not have permission to redistribute but are in the SEP data library.

CR denotes Conditional Reproducibility. The author certifies that the commands are in place to reproduce the figure if certain resources are available. The primary reasons for the CR designation is that the processing requires 20 minutes or more, MPI or CUDA based code, or commercial packages such as Matlab or Mathematica.

NR denotes Non-Reproducible figures. SEP discourages authors from flagging their figures as NR except for figures that are used solely for motivation, comparison, or illustration of the theory, such as: artist drawings, scannings, or figures taken from SEP reports not by the authors or from non-SEP publications.

Our testing is currently limited to LINUX 2.6 (using the Intel compiler), but the code should be portable to other architectures. Reader's suggestions are welcome. More information on reproducing SEP's electronic documents is available online³.

¹<http://sepwww.stanford.edu/private/docs/sep170>

²<http://sepwww.stanford.edu/public/docs/sepdatalib/toc.html>

³<http://sepwww.stanford.edu/research/redoc/>

SEP170 — TABLE OF CONTENTS

Modeling and inversion

<i>Huy Le, Anshuman Pradhan, Nader Dutta, Biondo Biondi, Tapan Mukerji, and Stewart A. Levin</i> , Building pore pressure and rock physics guides to constrain anisotropic waveform inversion...	1
<i>Joseph Jennings, Biondo Biondi, Robert G. Clapp, and Shuki Ronen</i> , Waveform inversion of multicomponent blended data with polarization filters.....	13
<i>Huy Le, Stewart A. Levin, and Robert G. Clapp</i> , Synthesis, processing, and migration of a 2D data from the Gulf of Mexico	17

Passive

<i>Jason P. Chang and Biondo Biondi</i> , Classification of wave modes extracted from passive data at Moere Vest	25
<i>Eileen Martin and Biondo Biondi</i> , Time-lapse changes in ambient noise interferometry and dispersion analysis at the Stanford DAS Array.....	45
<i>Eileen R. Martin, Biondo Biondi, Gabriel Fabien-Ouellet, and Robert G. Clapp</i> , Sensitivity analysis of distributed acoustic sensing arrays.....	57
<i>Siyuan Yuan, Eileen R. Martin, Jason P. Chang, Steve Cole, and Biondo Biondi</i> , Catalog of Northern California earthquakes recorded by DAS.....	81

Velocity estimation

<i>Guillaume Barnier, Ettore Biondi, and Biondo Biondi</i> , Modified Tomographic Full Waveform Inversion using the variable projection method	97
<i>Biondo Biondi, Ettore Biondi, and Guillaume Barnier</i> , Can full waveform inversion image all scales of the velocity model?	111
<i>Taylor Dahlke</i> , Representing salt bodies with radial basis functions	123
<i>Taylor Dahlke, Biondo Biondi, and Robert Clapp</i> , Using the Hessian of a radial basis formulation for level set inversion	133
<i>Alejandro Cabrales-Vargas</i> , Implementing Wave-Equation Migration Velocity Analysis Within Linearized Waveform Inversion with Velocity Updating: Considerations and Challenges	149
<i>Rahul Sarkar and Biondo Biondi</i> , A 2D Helmholtz equation solver library based on C++ and SuiteSparse	157

Deconvolution and signal processing

<i>Stuart Farris and Rustam Akhmadiev</i> , Permian Basin High Density Land Acquisition Experimental Dataset	175
<i>Jon Claerbout and Kaiwen Wang</i> , Multichannel data: separating independent causes	189
<i>Stewart A. Levin</i> , Short Note: PEF swapping positive with negative lags	207
<i>Jon Claerbout</i> , Short Note: Time variable prediction without mathematics	209

Software

<i>Robert G. Clapp, Stuart Farris, Taylor Dahlke, and Eileen Martin</i> , C++11 non-linear solver	211
<i>Stewart A. Levin</i> , SEPlib CMake update	221
SEP phone directory	225
('SEP article published or in press, 2017')	235

Building pore pressure and rock physics guides to constrain anisotropic waveform inversion

Huy Le, Anshuman Pradhan, Nader Dutta, Biondo Biondi, Tapan Mukerji, and Stewart A. Levin

ABSTRACT

We developed a workflow that combines various sources of information, such as geomechanics, well logs, basin history, and diagenesis, to model pore pressure-velocity relation based on rock physics principles. Our workflow produces velocity templates, which can be used as constraints in any anisotropic waveform inversion process. We apply our workflow to a data set from the Gulf of Mexico. We study the diagenesis of shale, particularly, smectite-illite reaction. From well logs, we build models for velocity-porosity and density-overburden relations. Thermal history is approximated from available Bottom Hole Temperature (BHT) data and depositional history is inferred from interpreted horizons. We use mud weight data to calibrate our pore pressure-velocity transformation. A number of different pore pressure gradient scenarios result in different velocity profiles or templates. Combining with mud weight data, these templates provide bound constraints to waveform inversion. The integration and calibration of many sources of data in our workflow ensure the resulting velocity model is geologically feasible physically plausible.

INTRODUCTION

Anisotropic imaging has been shown to be necessary in many successful exploration applications, particularly in the Gulf of Mexico. Alignment of clay minerals in shales and the effect of layering both imply transverse isotropy. Additionally, salt bodies in the Gulf of Mexico can cause stress perturbations that further complicate velocity variation.

Building anisotropic velocity models for imaging is a challenge due to large uncertainties in anisotropic parameters. Conventional velocity analysis and tomography of surface seismic usually do not provide a satisfactory answer because a number of models could equally well explain the observed data. Such is also the case with full waveform inversion (FWI). All of these inversion schemes rely heavily on the assumption that the initial model is close to the true model. When this assumption does not apply, there is a high possibility of obtaining a velocity model that satisfies the imposed convergence criterion but may be geologically and physically improbable. Our workflow imposes constraints that not only satisfy the gather flattening criterion but also require the model to be geologically and physically possible.

Anisotropic velocity models can be built with forward modeling using rock physics principles, geomechanics, and basin modeling. Bachrach (2010) used differential effective medium (DEM) theory from rock physics combined with well logs and empirical models of shale diagenesis to build anisotropic velocity models. Petmecky et al. (2009) derived anisotropic velocities for imaging from a 3D basin modeler to capture the pressure, depositional, fluid

flow, and salt movement histories of a basin. Matava et al. (2016) used finite elastic deformation theory to calculate the effect of stress anomalies caused by salt movements on velocity.

Recent developments in anisotropic velocity model building show that integrating additional data, such as rock physics and pore pressures, can constrain the velocity inversion process. Dutta et al. (2015) combined rock physics and pore pressure-velocity models to create velocity bounds for tomography. These constraints not only reduce uncertainty in the tomography process, but also produce a velocity model that is able to predict physical pore pressure. This is an extra constraint that forces the vertical velocity to be within a physically expected range such as yielding a pore pressure that is bounded below by hydrostatic pore pressure and above by fracture pressure. In addition, the use of rock physics compliant velocity model enables us to estimate vertical velocity without having to rely on normal moveout analysis, which often produce poor estimates of velocity. For a review on geopressure prediction, refer to Dutta (2002). Li et al. (2016) used stochastic rock physics modeling (Bachrach, 2010) to build model covariance matrices to constrain wave equation migration velocity analysis (WEMVA). Following Dutta et al. (2015), in this paper, we present a workflow that combines rock physics, basin modeling, and pore pressure constraints to improve anisotropic full waveform inversion (FWI).

WORKFLOW

Conceptual model

Our rock physics workflow applies to the diagenesis of shale, specially, the transformation of smectite into illite as a result of burial diagenesis. Our rock model consists of, therefore, a matrix solid (smectite and illite), and a pore fluid (water). Two processes can affect pore pressure. First, as sediments deposit, mechanical compaction causes porosity to reduce. Second, when clayey rocks are buried to deeper depths and temperature reaches activation temperatures, the transformation from smectite to illite happens and is accompanied by an additional release of water that is bound in the clay system of the host rocks, resulting in further increase in pore pressure.

In our workflow, we define forward modeling as obtaining vertical velocity models from pore pressures. First, effective stress is calculated for various pore pressure gradient scenarios by subtracting pore pressure from overburden stress. Second, effective stress is then converted into porosity using a compaction-diagenetic model. Finally, porosity is used to compute velocity via an attribute model, a velocity-porosity transformation. Our forward modeling produces velocity templates corresponding to different pore pressure gradients. These templates, when combined with mud weight data, serve as a guide to our inversion process. In the reverse direction, our workflow generates pore pressure predictions from an input of velocity.

Compaction-diagenetic model

Porosity is reduced due to mechanical loading. When loading is slow enough that pore fluid is allowed to escape, pore pressure maintains in a hydrostatic equilibrium. This process is

called normal compaction. In this mode, velocity increases as porosity decreases. When loading is faster than the rate of fluid escape, abnormal pressure builds up in the pores, causing effective stress to drop. In this mode of compaction disequilibrium, porosity is reduced at a lower rate than in normal compaction. Changes in porosity due to compaction are described through changes in effective stress.

In shale, diagenesis also affects pore pressure. The transition of smectite to illite, when temperature is high enough, is followed by a release of water. When such water cannot escape, pore pressure further increases and effective stress decreases without significant loss in porosity. We follow Dutta et al. (2014) and Dutta (2016) to model both of these mechanical compaction and diagenetic processes:

$$\sigma = \sigma_0 e^{-\xi\beta}, \quad (1)$$

where:

$$\xi = \frac{\phi}{1 - \phi}, \quad (2)$$

and

$$\beta(t) = B_0 N_s(t) + B_1 [1 - N_s(t)], \quad (3)$$

with:

$$N_s(t) = N_0 e^{-\int_0^t A e^{\frac{-E}{RT(t)}} dt}. \quad (4)$$

In the above equations, σ is effective stress and σ_0 is the effective stress necessary to reduce porosity, ϕ , to zero. ξ is the ratio of pore and solid volumes. β is the diagenetic function that characterizes smectite-illite transition. $N_s(t)$ is the smectite fraction at time t and N_0 is such fraction initially. We assume $N_0 = 1$. B_0 and B_1 control the relative importance of smectite and illite in the beta function. T is temperature. A and E are Arrhenius frequency factor and activation energy, respectively (Dutta et al., 2014).

Attribute model

Velocity is inversely proportional to porosity. In our workflow, we use a velocity-porosity relation that was derived in Issler (1992):

$$\Delta\tau = \Delta\tau_m (1 - \phi)^{-X}, \quad (5)$$

with $\Delta\tau$ being slowness, $\Delta\tau_m$ being the solid matrix's slowness, and X is the acoustic factor that captures how slowness (and velocity) varies with porosity.

We combine equations 1, 2, and 5 to get an equation of slowness and effective stress:

$$\Delta\tau = \Delta\tau_m \left[1 + \frac{1}{\beta} \ln \left(\frac{\sigma_0}{\sigma} \right) \right]^{-X}. \quad (6)$$

Effective stress is stress applied to the solid matrix and defined as the difference between overburden stress, S , and pore pressure, p :

$$\sigma = S - p. \quad (7)$$

Equations 3, 4, 6, and 7 form a complete transformation from pore pressure to velocity and vice versa.

APPLICATION TO FIELD DATA

We applied our workflow to a data set acquired offshore Gulf of Mexico. We were provided with a surface seismic data, an isotropic velocity obtained from ray-based tomography, migrated images, angle gathers, a number of interpreted horizons, logs and mud weight data from six wells in the area. Figure 1 shows the wells' locations overlaid on a depth slice at two kilometers of the velocity model. Among these six wells, well SS168 has BHT data and well SS187 has a density log. Figure 2(a) and 2(b) show the source and receiver locations of the seismic data.

Thermal and depositional histories

In our workflow, the computation of smectite fraction (Equation 4) and beta function (Equation 3) requires a thermal history, $T(t)$. We approximate a simple thermal history of the study area from temperature-depth and age-depth relationships. We used BHT data at well SS168 to build a piece-wise linear temperature profile and assumed a geothermal gradient, α , that did not change in geologic time:

$$T(t) = T_0 + \alpha z(t), \quad (8)$$

where T_0 is the temperature at sea bottom, which can be calculated as a function of latitude and water depth (Beardsmore and Cull, 2001). Figure 3(a) shows six temperature profiles at our well locations and BHT data at well SS168. We also assume a constant burial rate, γ , and estimate age-depth relationship from interpreted horizons, Top Pliocene and Top Miocene (Figure 3(b)):

$$z(t) = \gamma t. \quad (9)$$

Figures 4(a) and 4(b) show smectite fractions and beta functions with depth for $B_0 = 6.5$ and $B_1 = 14$. These figures show that smectite-illite transition starts at about 2–2.5 kilometers and ends at about 6–6.5 kilometers.

Overburden stress model

Overburden stress is calculated by:

$$S = S_0 + g \int_0^z \rho(z) dz, \quad (10)$$

where S_0 is the pressure of the water column at sea bottom and $\rho(z)$ is the depth-dependent density. Using data at well SS187, we compare different density models and the corresponding overburden stresses. Specifically, we select only shale data points using a gamma log (Figure 5) and build a diagenetic model for density by least-squares fitting the equation:

$$\rho = (a_s \Delta\tau + b_s) N_s + (a_i \Delta\tau + b_i) (1 - N_s). \quad (11)$$

Here coefficients a_s , b_s , a_i , and b_i describe linear relationships between slowness, $\Delta\tau$, and densities of smectite and illite respectively. Equation 11 can be used to predict density from velocity. Figure 6 shows the fitting result. Data points are color coded by depth, indicating smectite-illite transition as temperature and depth increase (also shown in Figure 4(a)).

Figure 7(a) plots Gardner’s and diagenetic density models and the actual density log at well SS187. We observe that diagenetic model well captures the low-frequency trend down to four kilometers deep and starts to deviate slightly in deeper sections. Figure 7(b) shows the comparison of different overburden stress models. Despite a slight difference among density models below four kilometers, overburden stress models well agree with the empirical model (Dutta (2017) private communication):

$$S = az^2 + bz + cz_0, \quad (12)$$

where $a = 0.0000585$, $b = 2.75$, $c = 1.493$, and z_0 is the water depth. Here stress is measured in psi and depth in meter. For simplicity, we will use this empirical model for overburden calculation.

Calibration

In our workflow, a number of important parameters need to be determined:

1. $\Delta\tau_m$ is the matrix’s slowness and X is the acoustic formation-factor exponent (Equation 5). Following Issler (1992), we find these two parameters by fitting the sonic transit time and porosity data from well logs. Figure 8 shows the fitting results for different wells. Among five wells used for fitting, well SS160 gives the most reasonable parameters:

$$\begin{aligned} \Delta\tau &= 2.13 \times 10^{-4} \text{ s/m}, \\ X &= 1.97. \end{aligned} \quad (13)$$

We use these values in our workflow. Other wells’ parameters seem either too high or too low.

2. σ_0 is the effective stress that can reduce porosity to zero (Equation 1). B_0 and B_1 determine relative contributions of smectite and illite in beta function (Equation 3). These parameters were chosen so that the sonic converted pore pressures are bounded below by hydrostatic pressure and above by mud weights. Figures 9(a), 10(a), 11(a), 12(a), 13(a), and 14(a) show the results for:

$$\begin{aligned} \sigma_0 &= 26000 \text{ psi}, \\ B_0 &= 6.5, \\ B_1 &= 14. \end{aligned} \quad (14)$$

Pore pressures and velocity templates

Figures 9(a), 10(a), 11(a), 12(a), 13(a), and 14(a) show the sonic and seismic converted pore pressure profiles together with mud weights, overburden, and fracture stresses at the wells’ locations. Here we take fracture stress to be 97% of overburden stress. Figures 9(b), 10(b), 11(b), 12(b), 13(b), and 14(b) show the corresponding velocity templates. We observed that the seismic velocities match generally well with the sonic velocities. Additionally, pore pressure profiles show a deviation from hydrostatic pressure at 3-4 kilometers. This deviation in pore pressures is also reflected on the velocity templates by a velocity reduction.

Moreover, the depth at which this pressure deviation and velocity reduction happen agrees with where smectite-illite transition starts (Figure 4(a)).

Well ST143 (Figure 12) and well ST168 (Figure 13) show a sharp decrease in pore pressure and an increase in velocity at about six kilometers. This is caused by the salt bodies that these two wells came in contact with. Salt bodies' generation and movement can lead to stress and velocity anomalies that our current workflow does not address.

CONCLUSIONS

We developed a workflow that produces velocity templates which can be used to guide an anisotropic waveform inversion process. Our workflow uses seismic, well logs, and geomechanical data to model depositional, thermal histories, and shale diagenesis. Combined with rock physics principles, we build a transformation that can predict pore pressure from velocity. Incorporated as constraints to an waveform inversion, this assures that the inverted velocity model is physically plausible. This is the focus of our ongoing work.

ACKNOWLEDGEMENTS

We would like to thank Schlumberger MultiClient for providing us the seismic data and IHS Energy Log Services Inc. for the well logs.

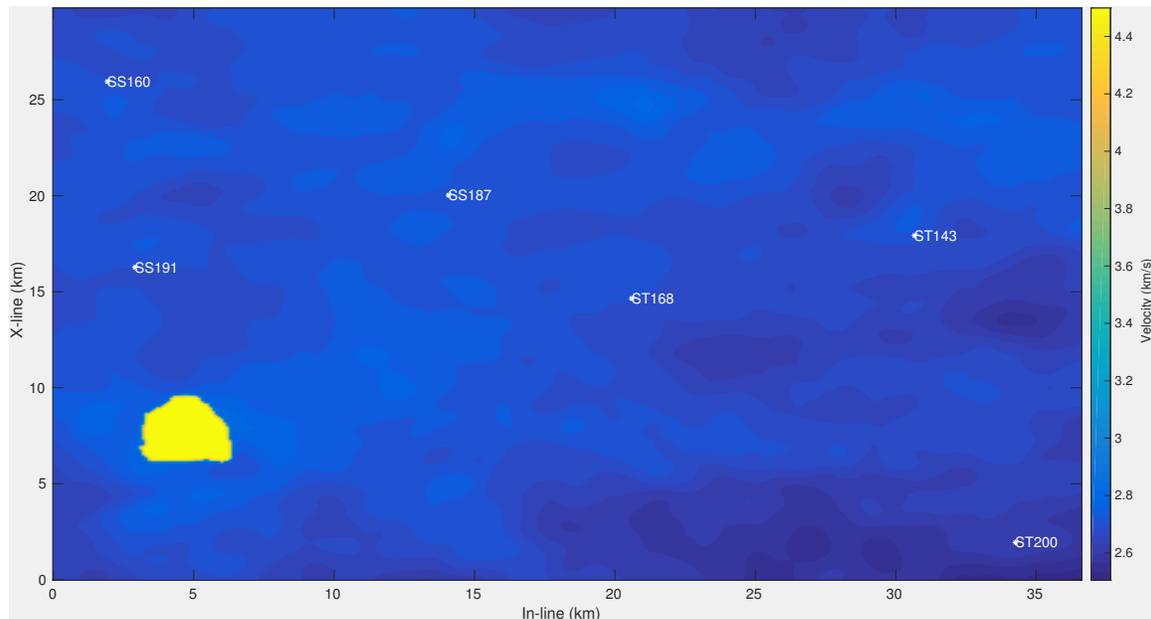


Figure 1: Depth slice at 2 km of the velocity model and well locations. [ER]

huyle2/. velocitywells

REFERENCES

Bachrach, R., 2010, Applications of deterministic and stochastic rock physics modeling to anisotropic velocity model building: SEG Annual International Meeting, Expanded

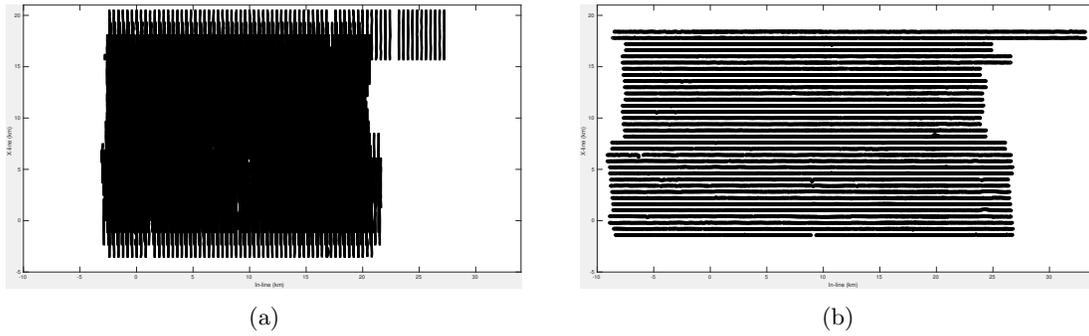


Figure 2: Source (left) and receiver (right) locations of the provided seismic data. [ER] huyle2/. s4ph4s1,s4ph4r1

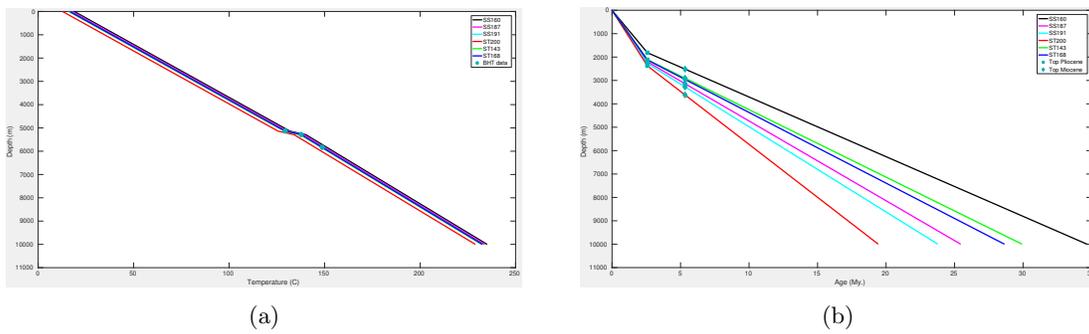


Figure 3: Temperature-depth (left) and geologic age-depth (right) relationships. [ER] huyle2/. temps,ages

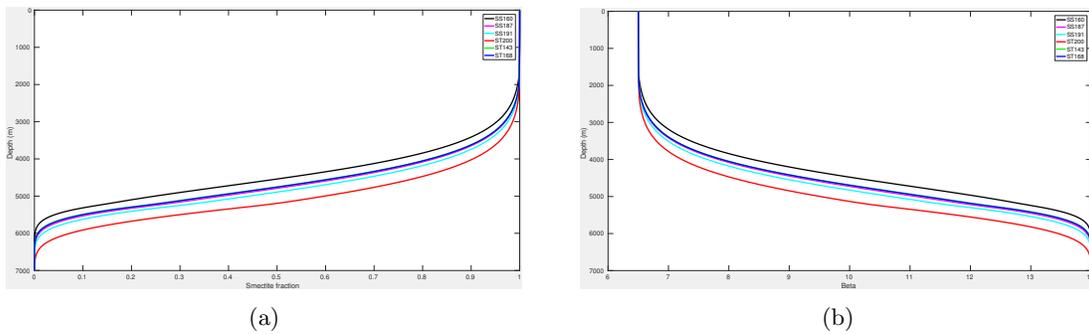


Figure 4: Smectite fractions (left) and beta functions (right) at well locations. [ER] huyle2/. smectite,beta

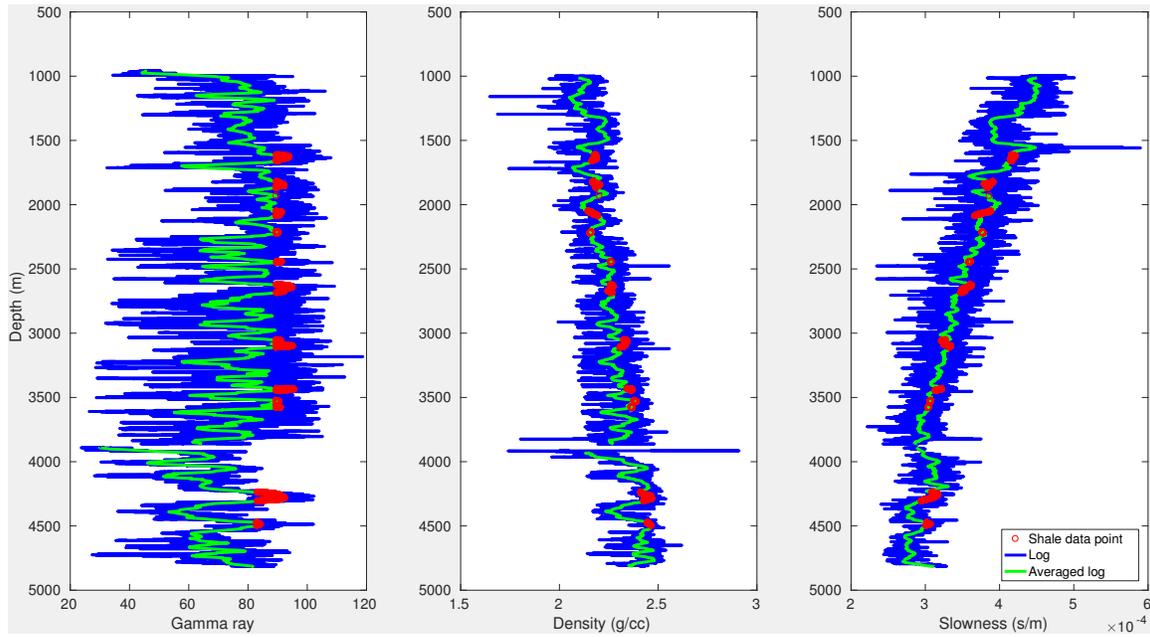


Figure 5: Shale data point selection at well SS187. [ER] huyle2/. shalerhodtSS187

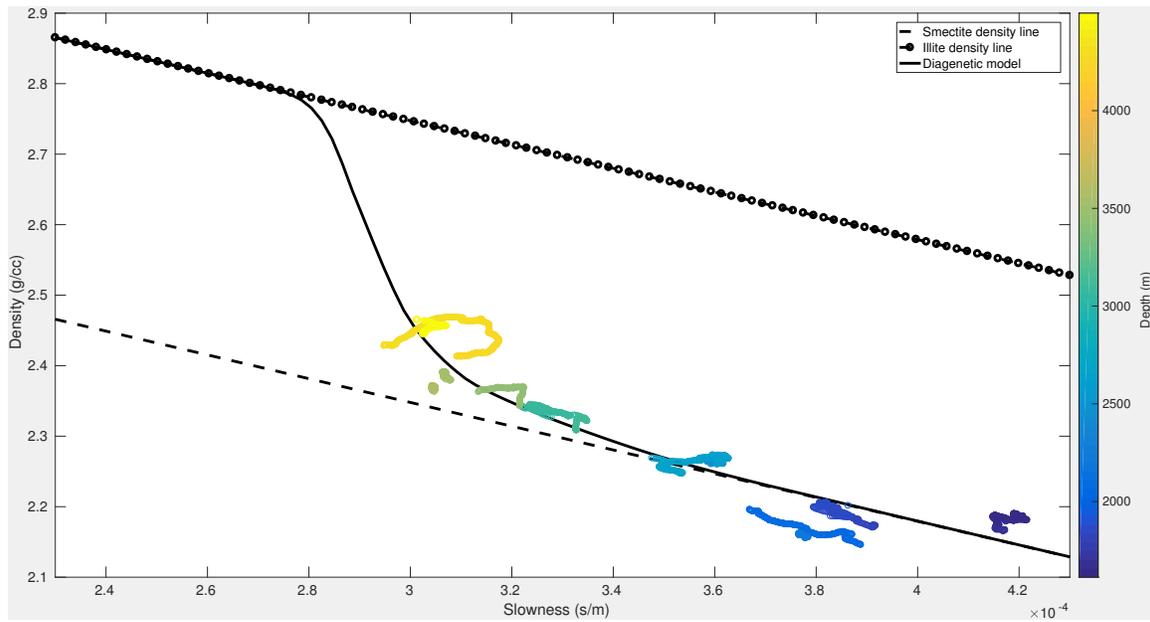


Figure 6: Diagenetic model for density at well SS187. [ER] huyle2/. rhodtSS187

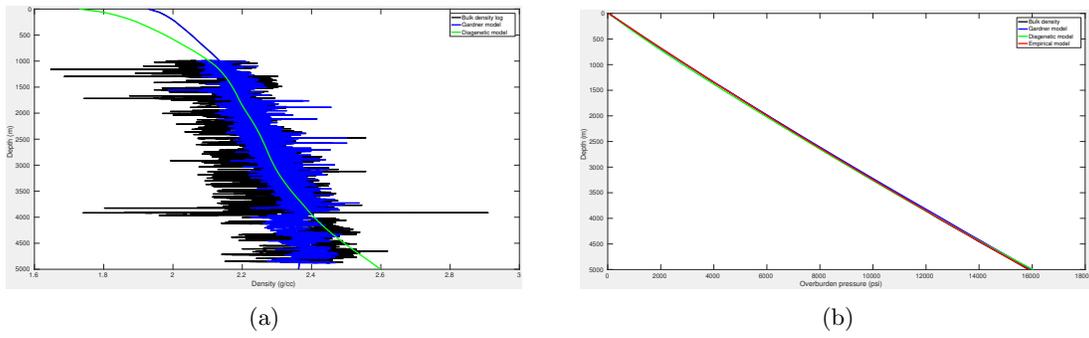


Figure 7: Different density models (left) and overburden models (right) at well SS187. [ER] huyle2/. densityModelsSS187,overburdenModelsSS187

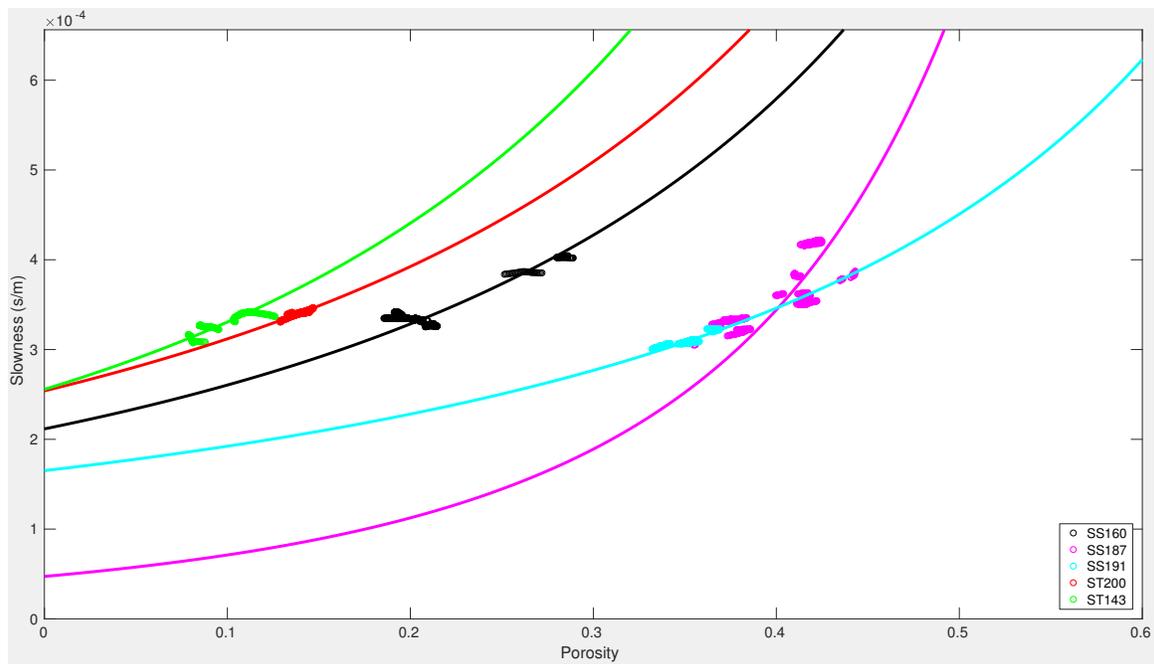


Figure 8: Least-squares fitting for Issler's model. [ER] huyle2/. dtmx

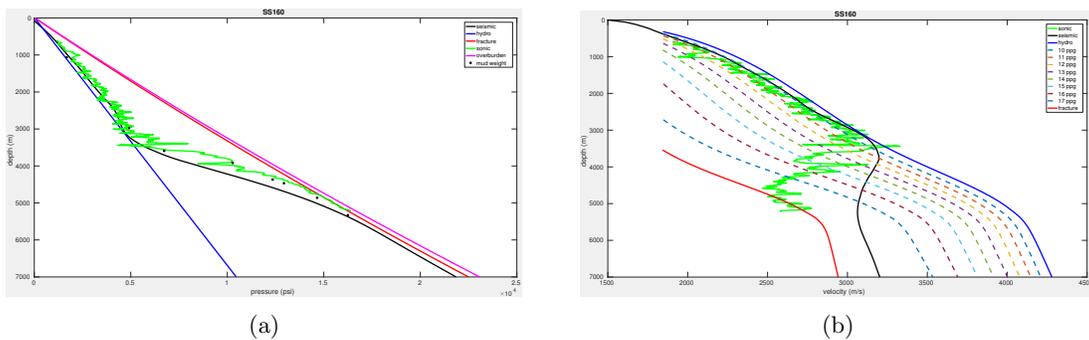


Figure 9: Pore pressure profile (left) and velocity template (right) at well SS160. [ER] huyle2/. pp-SS160,rpt-SS160

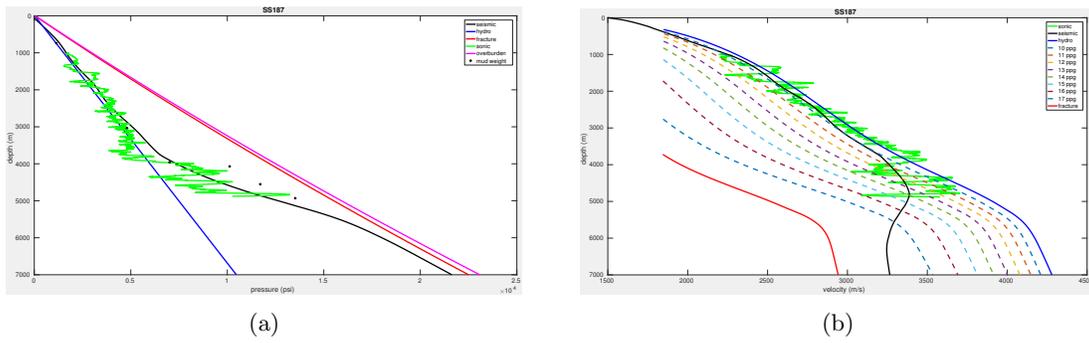


Figure 10: Pore pressure profile (left) and velocity template (right) at well SS187. [ER] huyle2/. pp-SS187,rpt-SS187

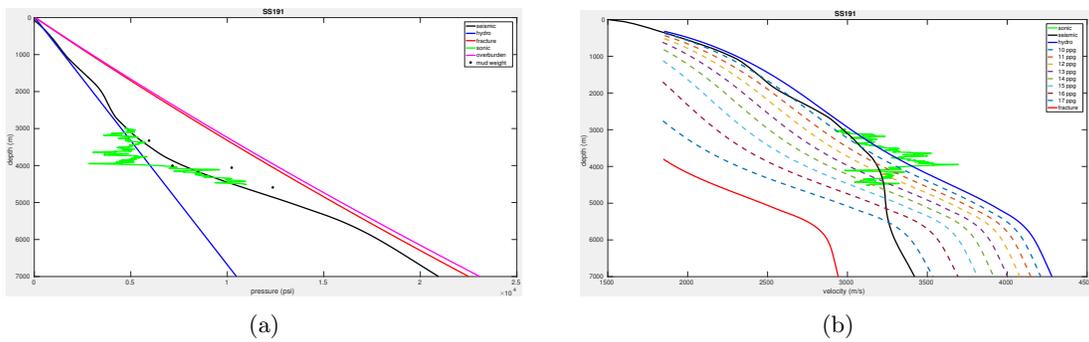


Figure 11: Pore pressure profile (left) and velocity template (right) at well SS191. [ER] huyle2/. pp-SS191,rpt-SS191

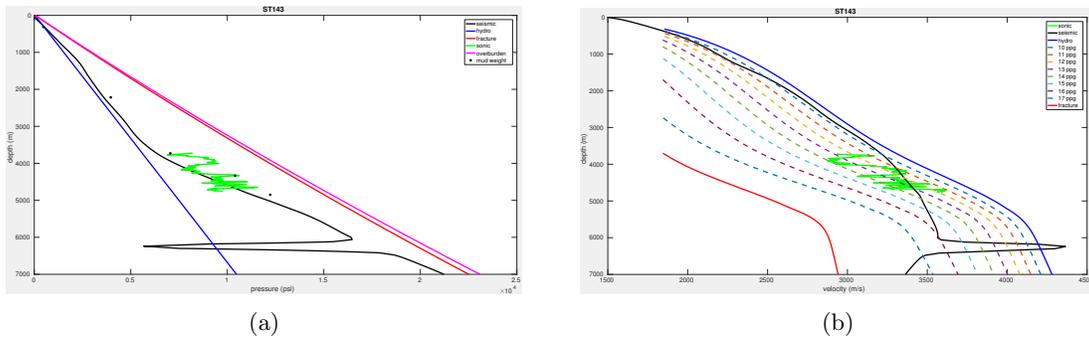


Figure 12: Pore pressure profile (left) and velocity template (right) at well SS143. [ER] huyle2/. pp-ST143,rpt-ST143

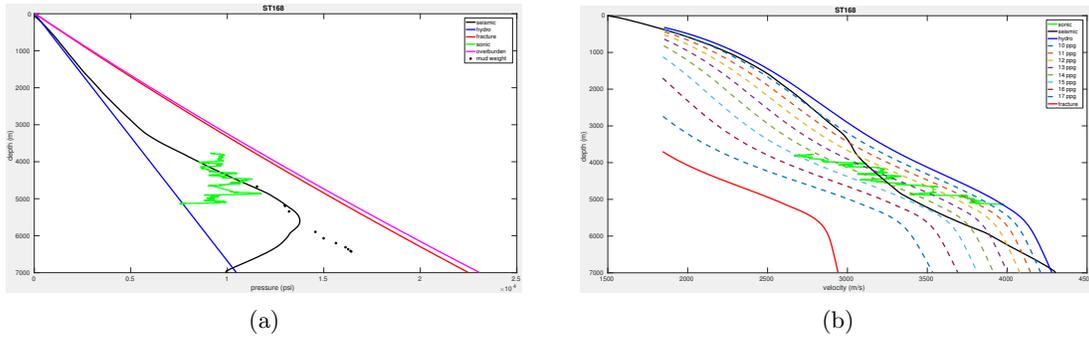


Figure 13: Pore pressure profile (left) and velocity template (right) at well ST168. [ER] huyle2/. pp-ST168,rpt-ST168

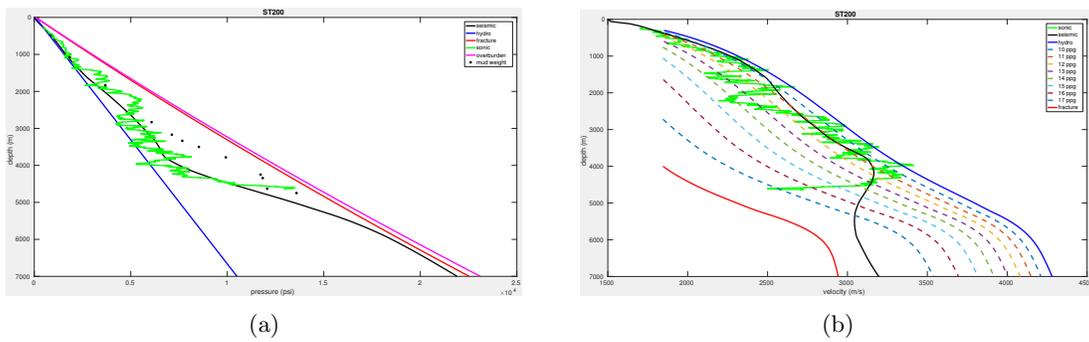


Figure 14: Pore pressure profile (left) and velocity template (right) at well ST200. [ER] huyle2/. pp-ST200,rpt-ST200

- Abstracts, 2436–2440, Society of Exploration Geophysicists.
- Beardsmore, G. R. and J. P. Cull, 2001, *Crustal heat flow*: Cambridge University Press.
- Dutta, N., B. Deo, Y. K. Liu, Krishna, Ramani, J. Kapoor, and D. Vigh, 2015, Pore-pressure-constrained, rock-physics-guided velocity model building method: Alternate solution to mitigate subsalt geohazard: *Interpretation*, **3**, SE1–SE11.
- Dutta, N. C., 2002, Geopressure prediction using seismic data: Current status and the road ahead: *Geophysics*, **67**, 2012–2041.
- , 2016, Effect of chemical diagenesis on pore pressure in argillaceous sediment: The Leading Edge, **35**, 523–527.
- Dutta, N. C., S. Yang, J. Dai, S. Chandrasekhar, F. Dotiwala, and C. V. Rao, 2014, Earth-model building using rock physics and geology for depth imaging: *The Leading Edge*, **33**, 1136–1152.
- Issler, D. R., 1992, A new approach to shale compaction and stratigraphic restoration, Beaufort-Mackenzie basin and Mackenzie corridor, Northern Canada: *AAPG Bulletin*, **76**, 1170–1189.
- Li, Y., B. Biondi, R. Clapp, and D. Nichols, 2016, Integrated VTI model building with seismic data, geological information, and rock-physics modeling-Part 1: Theory and synthetic test: *Geophysics*, **81**, C177–C191.
- Matava, T., R. Keys, D. Foster, and D. Ashabraner, 2016, Isotropic and anisotropic velocity-model building for subsalt seismic imaging: *The Leading Edge*, **35**, 240–245.
- Petmecky, R. S., M. L. Albertin, and N. Burke, 2009, Improving sub-salt imaging using 3d basin model derived velocities: *Marine and Petroleum Geology*, **26**, 457–463.

Waveform inversion of multicomponent blended data with polarization filters

Joseph Jennings, Biondo Biondi, Robert G. Clapp, and Shuki Ronen

ABSTRACT

We present a new algorithm for directly imaging blended data via waveform inversion. The algorithm makes use of the directional information contained within multicomponent blended data by introducing polarization filters at each iteration. We show that with the introduction of these polarization filters, the waveform inversion results contain significantly fewer artifacts than those obtained with conventional waveform inversion of blended data.

INTRODUCTION

The main challenge associated with waveform inversion of blended data is the artifacts that appear due to the cross talk between sources. The current state of the art is to regularize the inversion by imposing constraints that smooth along the structure present in the inverted model parameters (Tang et al., 2009; Xue et al., 2014). While this technique is effective at suppressing the spurious artifacts, it compromises resolution in the earth model.

To overcome this issue, we present a modified waveform inversion algorithm that makes use of the directional information contained in the multicomponent data that are generally acquired during simultaneous source surveys with ocean-bottom node receivers. We incorporate this additional information into the waveform inversion by introducing polarization filters at each iteration. We estimate these filters and then apply them to the blended data to give approximate deblended data. We then compute the waveform inversion gradient on these approximately deblended data which provides gradients with much fewer artifacts and therefore better model updates. In a linearized waveform inversion example for acoustic multicomponent blended data, we show that the inverted model resulting from this algorithm has significantly fewer artifacts.

THEORY

The conventional objective function for waveform inversion of blended data can be written as follows:

$$J(\mathbf{m}) = \frac{1}{2} \|\tilde{\mathbf{f}}(\mathbf{m}) - \tilde{\mathbf{d}}\|_2^2, \quad (1)$$

where \mathbf{m} is an earth model (e.g., velocity model), $\tilde{\mathbf{f}}$ is the non-linear blended wave equation modeling operator and $\tilde{\mathbf{d}}$ is the recorded blended data. The minimization of equation 1 results in an optimal \mathbf{m} such that the synthetically modeled blended data $\tilde{\mathbf{f}}(\mathbf{m})$ fits the true recorded data in the least-squares sense. Waveform inversion of multicomponent blended

data with polarization filters minimizes the following modified blended waveform inversion objective function:

$$J(\mathbf{m}) = \frac{1}{2} \|\mathbf{f}(\mathbf{m}) - \mathbf{P}\tilde{\mathbf{d}}\|_2^2, \quad (2)$$

where \mathbf{m} is again an earth model, \mathbf{f} is an independent modeling operator, $\tilde{\mathbf{d}}$ is multicomponent blended data and \mathbf{P} contains polarization filters.

These polarization filters are estimated from the independently modeled data (as a result of $\mathbf{f}(\mathbf{m})$) and then are used in order to deblend the data by minimizing the following objective function:

$$J(\mathbf{d}_1, \mathbf{d}_2) = \frac{1}{2} \|\mathbf{d}_1 + \mathbf{d}_2 - \tilde{\mathbf{d}}\|_2^2 + \frac{1}{2} \|\mathbf{P}_2(\mathbf{d}_1 - \tilde{\mathbf{d}})\|_2^2 + \frac{1}{2} \|\mathbf{P}_1(\mathbf{d}_2 - \tilde{\mathbf{d}})\|_2^2, \quad (3)$$

where \mathbf{d}_1 and \mathbf{d}_2 are the deblended data and \mathbf{P}_1 and \mathbf{P}_2 are the polarization filters estimated from the independently modeled data. Note that minimizing equation 3 is equivalent to $\mathbf{P}\tilde{\mathbf{d}}$. Once the data are approximately deblended via the minimization of this objective function, they are then used to compute the waveform inversion gradient and subsequently update the model. This process of deblending and imaging at each iteration is repeated until convergence.

RESULTS

We tested the algorithm for a linearized waveform inversion problem with synthetic multicomponent acoustic blended data generated from a dipping reflector and syncline model. The results after four iterations of conventional blended linearized waveform inversion and waveform inversion with polarization filters are shown in Figures 1(a) and 1(b) respectively. Note that the result of waveform inversion with polarization filters has a significant reduction in artifacts.

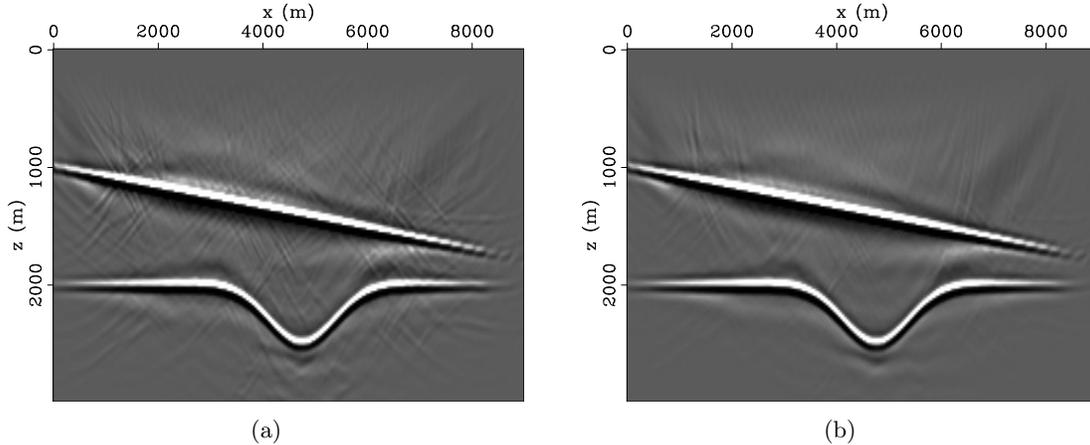


Figure 1: Results after four iterations of (a) conventional linearized waveform inversion of multicomponent blended data and (b) linearized waveform inversion of multicomponent blended data with polarization filters. Note the significant reduction of artifacts in the inverted model obtained in panel (b) vs the model obtained in panel (a). [CR]

joseph1/. dmlblwiiterss4,dmlwipfitterss4

CONCLUSION AND FUTURE WORK

We have shown that using the directional information contained in multicomponent blended data can improve imaging of blended data. Our waveform inversion of multicomponent blended data with polarization filters, that makes use of this directional information by filtering the blended data at each iteration, provided better waveform inversion results than conventional waveform inversion. In the future, we plan to apply this algorithm on multicomponent acoustic data acquired in the Mediterranean Sea by Kietta (Haumonté, 2016). Figure 2 shows the windowed direct arrival and reflections of a single four component shot gather acquired during this survey. As these data are not blended, we will first blend them and then use these blended data as input to our modified waveform inversion algorithm. In addition to working with these data, we plan to extend our current numerical implementation to a fully elastic waveform inversion to apply our algorithm to blended multicomponent ocean-bottom node data.

ACKNOWLEDGEMENTS

The authors would like to thank Kietta for permission to show the data. They also would like to thank the SEP sponsors for their intellectual and financial support.

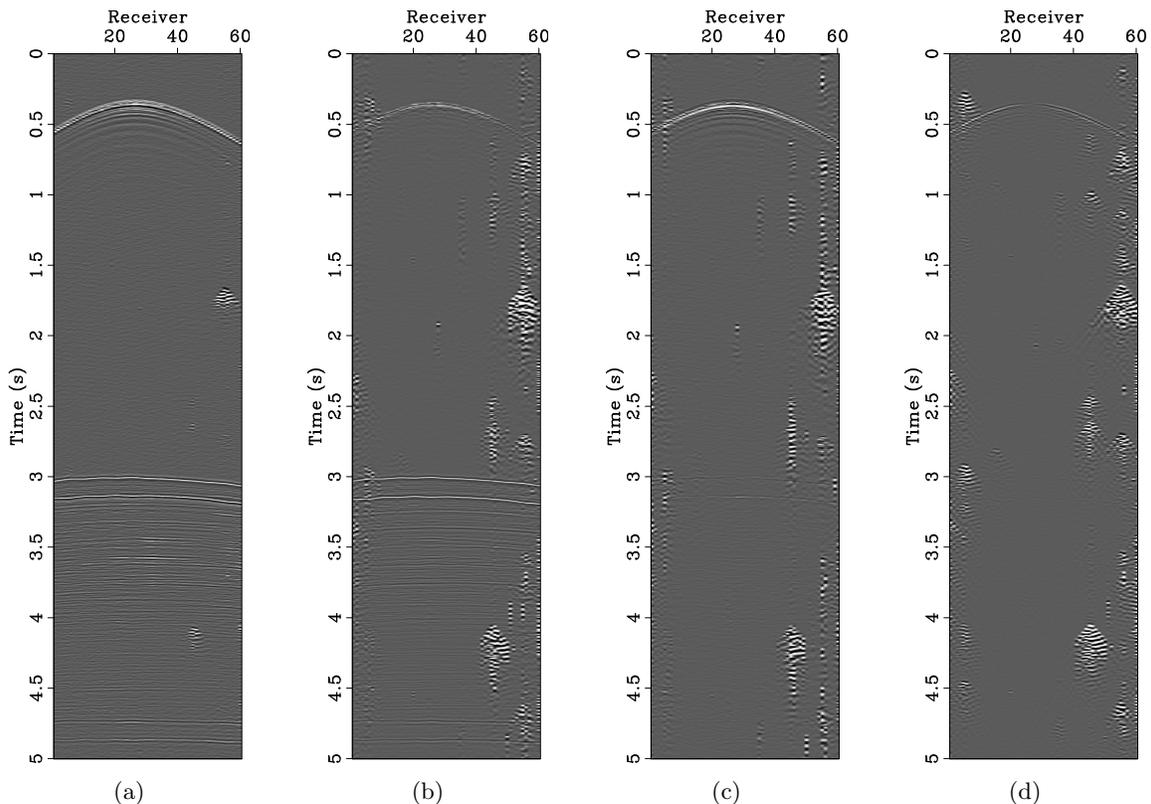


Figure 2: A single multicomponent acoustic shot gather acquired in the Mediterranean sea by Kietta. (a) Hydrophone component, (b) vertical component, (c) crossline component and (d) inline component. [CR] joseph1/. Hydro1138,Vert1138,XL1138,IL1138

REFERENCES

- Haumonté, L., 2016, In situ test results obtained in the mediterranean sea with a novel marine seismic acquisition system (freecable), *in* SEG Technical Program Expanded Abstracts 2016, 51–55, Society of Exploration Geophysicists.
- Tang, Y., B. Biondi, et al., 2009, Least-squares migration/inversion of blended data: Presented at the 2009 SEG Annual Meeting.
- Xue, Z., Y. Chen, S. Fomel, J. Sun, et al., 2014, Imaging incomplete data and simultaneous-source data using least-squares reverse-time migration with shaping regularization: Presented at the 2014 SEG Annual Meeting.

Synthesis, processing, and migration of a 2D data from the Gulf of Mexico

Huy Le, Stewart A. Levin, and Robert G. Clapp

ABSTRACT

As a step toward applying our anisotropic waveform inversion methodology developed and presented in previous reports, we synthesized a 2D seismic data set from a 3D cross-shooting data acquired offshore Gulf of Mexico. The data's cross-shooting geometry poses a challenge to 2D processing and migration. Since the source lines and receiver lines are perpendicular, out-of-plane events could degrade image quality. Additionally, there are a number of salt bodies in the data area, which, at some locations, rise up to sea bottom with steep flanks causing imaging problems with short-offset data. We migrated the resulting 2D data with an isotropic velocity. Despite all the limitations, we were able to image reflectors down to four kilometers depth. However, segments of the salt bodies' boundaries could hardly be traced due to low signal to noise ratio in deeper portions of the data.

INTRODUCTION

In previous reports, we implemented a time-domain anisotropic full-waveform inversion (FWI) method using second-order pseudo-acoustic wave equations in transverse isotropic media and applied it to synthetic models and data (Le, 2016a,b). Our next step is to test our implementation on a field data set. Among data donated to SEP by our sponsors, the E-Dragon data caught our attention. The data area is rich in shale and layered in many places, which makes it suitable for anisotropic imaging and inversion. Alignment of clay minerals in shale and layering combine to create a medium of at least transverse isotropy. Moreover, salt bodies' movement could potentially cause stress anomalies, which further lowers the medium's symmetry to, e.g., orthorhombic. Isotropic imaging applied to areas that display anisotropy can result in poor focus and wrong placement of reflectors.

Anisotropic model building is a challenging problem because of the large number of unknown parameters, their trade-offs, and uncertainty. In order to constrain anisotropic inversion, additional sources of information should be incorporated. E-Dragon data come together with well logs, mud weights, and attribute cubes, such as shale volume, pore pressure, and effective stress, which makes it possible to explore different types of constraints. The availability of data from E-Dragon has promoted the work of a former SEP student, Elita Li (Li et al., 2016), who used geological and rock physics constraints in Wave Equation Migration Velocity Analysis (WEMVA). In another paper in this report, we develop a workflow that use geomechanical information and basin modeling for constraining anisotropic FWI.

ACQUISITION AND PROCESSING

The data set we chose was acquired in the Gulf of Mexico at four millisecond sampling, using ocean bottom cables (OBC). The area where it was recorded has a shallow water depth, approximately 36 meters on average. We were provided with P-Z summed data. Figures 1(a) and 1(b) respectively show the source locations and receiver locations. The source lines are perpendicular to the receiver lines. Source line spacing is 400 meters and source spacing is 50 meters, while receiver line spacing is 600 meters and receiver spacing is 50 meters. Maximum offset is about six kilometers.

To synthesize a 2D data set from the original 3D data, we chose a subset of the 3D data with midpoints within a one-kilometer swath, covering two receiver lines. These receiver lines are in close proximity to one of the wells in the survey area and overlay interesting salt bodies that we hope to image. Figure 2(a) shows locations of the midpoints, two receiver lines, and corresponding sources.

Assuming structures and velocity do not vary significantly in the cross-line direction, the chosen sources and receivers were rotated about their midpoints to align in-line. Figure 2(b) shows locations of the sources and receivers after rotation. Rotated sources and receivers densely cover a patch one kilometer wide and 30 kilometers long, which we will regard as our 2D data. The rotation is trusted to not change reflection moveouts because the water depth at this area is particularly shallow. As a result, no differential moveout correction was applied. The resulting data were then sorted into 50-meter bins and stacked. This produces 536 shots, of which only one quarter is used for migration. We also applied a low-pass 30 Hz filter to the data. Figures 3(a) and 3(b) show two sample shots. Some reflection hyperbolas are identifiable in the first four seconds of the data. Little coherent signal shows in the later parts of the data.

MIGRATION

The data came with an isotropic velocity model that was obtained with ray-based tomography. Figure 4(a) shows a 2D section of the velocity model along our synthesized 2D line. The velocity section shows monotonic variation with depth down to about four kilometers, but changes laterally in deeper regions, especially around salt bodies. Those are areas we expect our 2D approximation and rotation break down. Figure 4(b) shows the same 2D velocity section, in which the salt bodies were already removed and filled with surrounding sediment velocity. This is our migration velocity.

We performed isotropic migration on the synthesized data using a Ricker wavelet of 10 Hz central frequency. Figure 5 shows the migrated image. As expected, we observed a number of continuous and relatively flat reflectors down to four kilometers. The image degrades below that. This is due to low data quality and inaccuracy of our 2D assumption. This is also where the salt bodies locate and velocity significantly changes laterally. Consequently, we were able to image only segments of the salt bodies at around seven kilometer depth.

Figures 6 and 7 show angle gathers at various locations. A number of reflectors with downward curvatures can be identified. This indicates our migration velocity might be faster than true velocity. This is because isotropic ray-based tomography tends to capture normal moveout velocity, which, in areas of positive ϵ , is greater than vertical velocity. This

leaves room for improvement with anisotropic velocity models that we hope to build.

CONCLUSIONS

We synthesized a 2D data set from cross-shooting 3D data from the Gulf of Mexico. We chose an area that has interesting salt structures and is close to wells that can be used later in the inversion process. With an isotropic velocity model, we were able to image sedimentary layers above the salt bodies. Even though the image degrades in the deeper half, segments of the salt boundaries were visible. The angle gathers display downward curvatures on a number of reflectors, indicating that the migration velocity is higher than true velocity. This is where our anisotropic FWI tools can help to build a better velocity model and improve these angle gathers.

ACKNOWLEDGEMENT

We would like to thank Schlumberger MultiClient for providing us the field data. We also thank Professor Biondo Biondi, Huy Le's thesis advisor, for many insightful discussions.

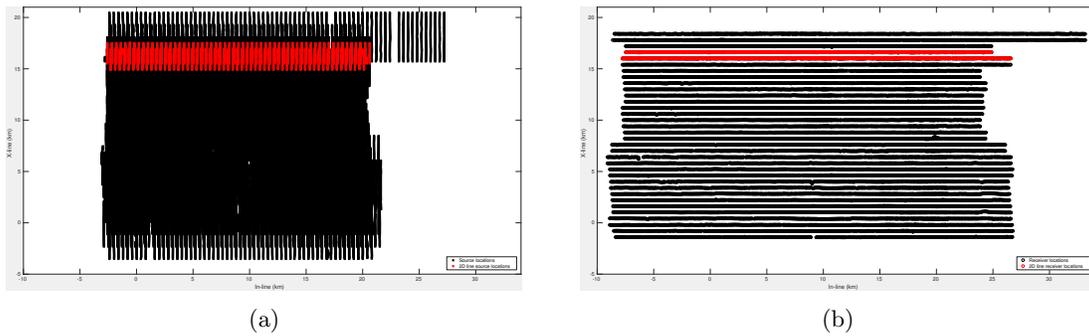


Figure 1: Source (left) and receiver (right) locations of the original 3D data set and location of the picked 2D line (in red). [ER] huyle1/. s4ph4s,s4ph4r

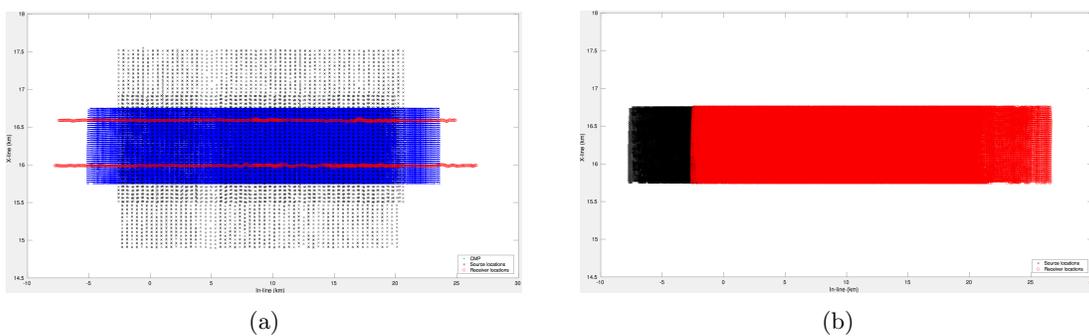


Figure 2: Source, receiver, and midpoint locations before (left) and after (right) rotation. [ER] huyle1/. s4ph4line0cmpr2,s4ph4line0rotated2

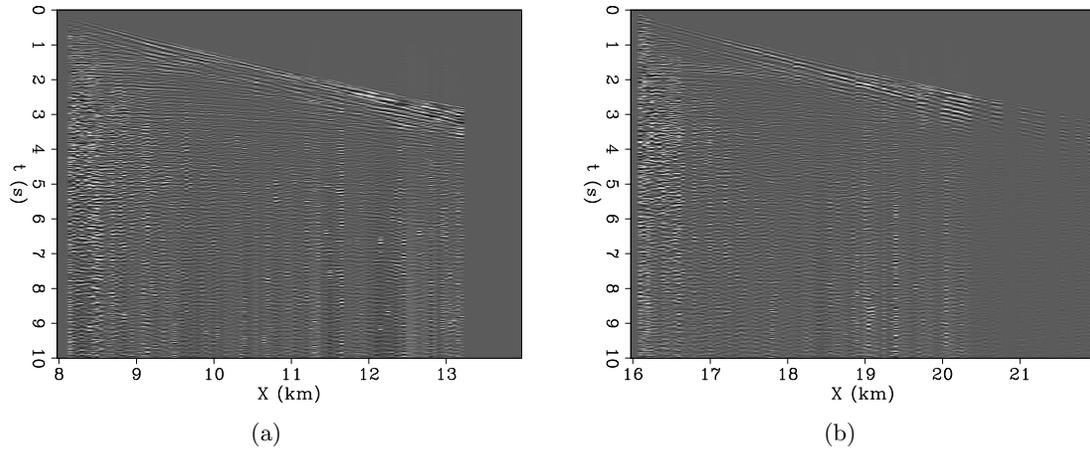


Figure 3: Two sample shot gathers. [ER] huyle1/. shot1,shot2

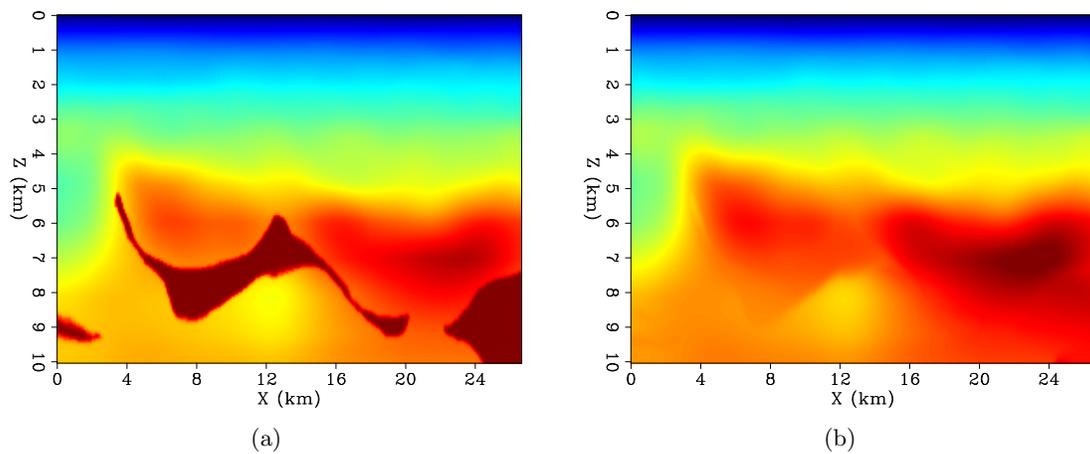


Figure 4: On the left is the provided isotropic velocity model, obtained by ray-based tomography. On the right is the velocity model in which the salt bodies have been replaced with sediments. [ER] huyle1/. bgv,smoothbgvnosalt

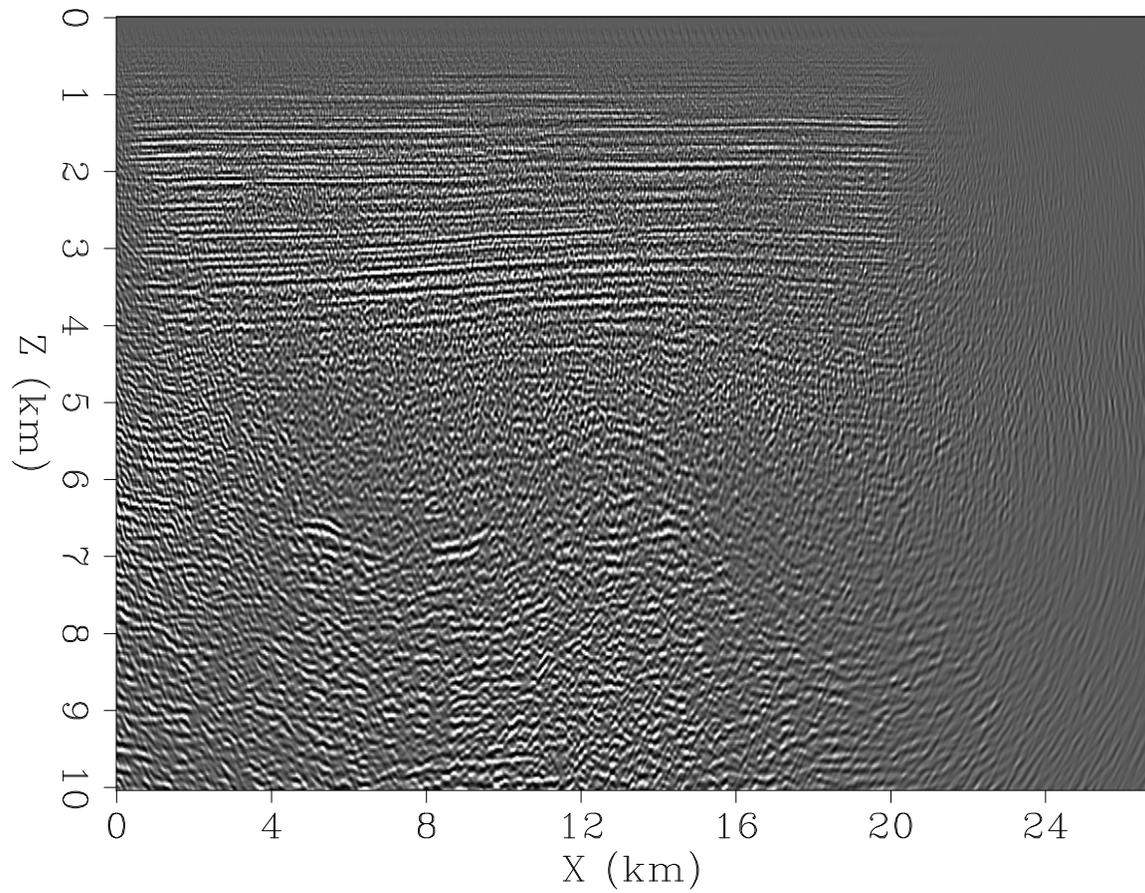


Figure 5: Migrated image with isotropic velocity model shown in Figure 4(b) shows sedimentary layers above four kilometers and segments of the salt boundaries at about seven kilometers. [CR] huyle1/. image

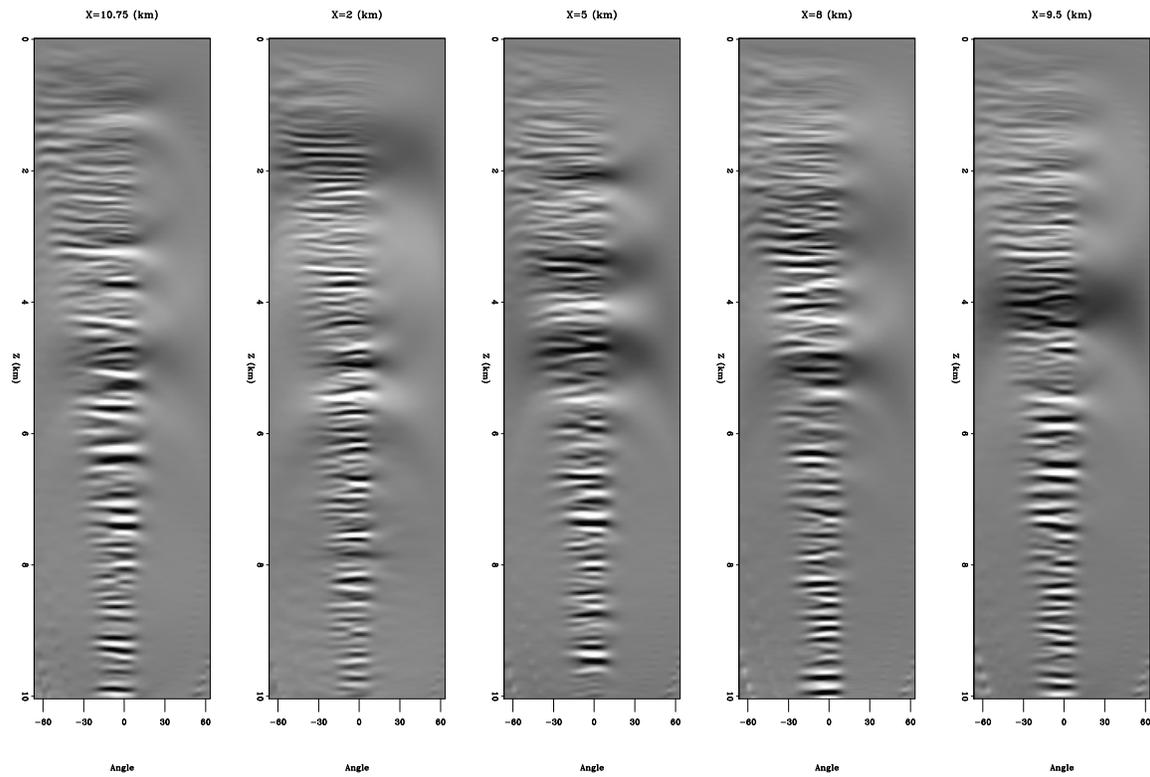


Figure 6: Angle gathers show downward curvatures at some locations, for example at five kilometers depth at $X = 10.75$ km and 4.5 km depth at $X = 2$ km. [CR]

huyel/. anglegather1

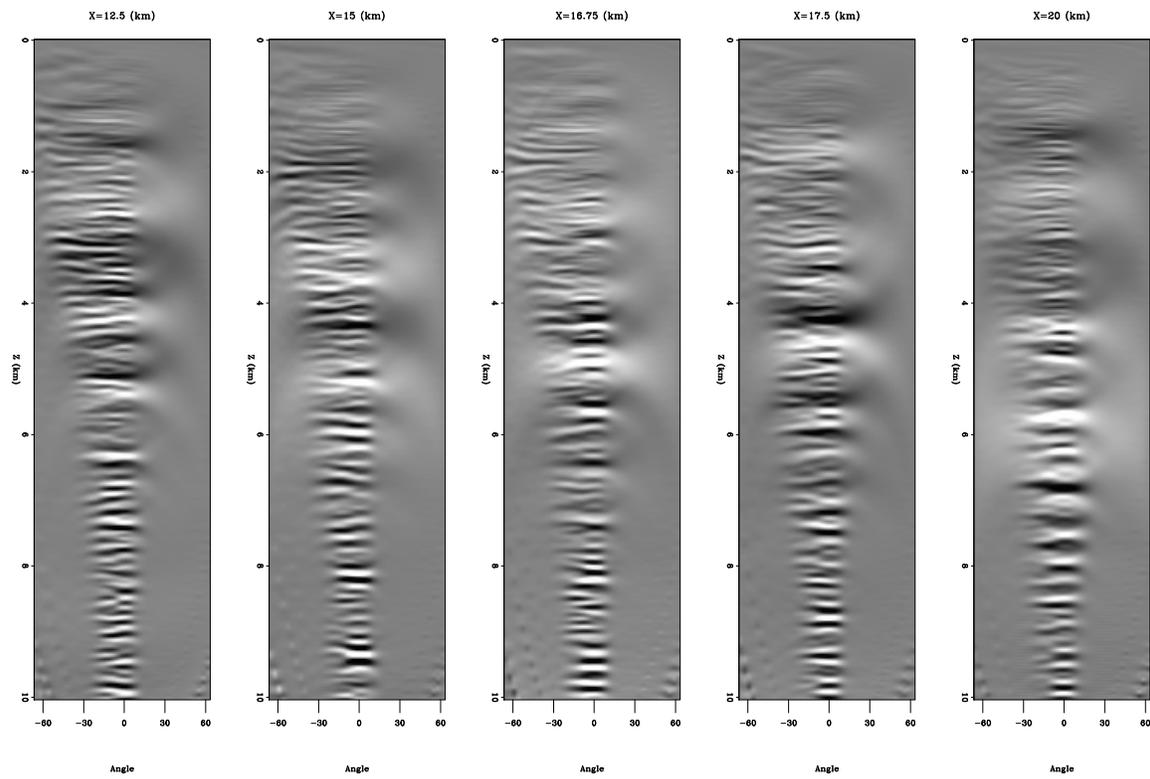


Figure 7: More angle gathers at various locations. [CR] huyle1/. anglegather2

REFERENCES

- Le, H., 2016a, Anisotropic full waveform inversion: SEP-Report, **163**, 155–162.
 ———, 2016b, Practical issues in anisotropic full waveform inversion: SEP-Report, **165**, 1–15.
 Li, Y., B. Biondi, R. Clapp, and D. Nichols, 2016, Integrated VTI model building with seismic data, geological information, and rock-physics modeling-Part 1: Theory and synthetic test: *Geophysics*, **81**, C177–C191.

Classification of wave modes extracted from passive data at Moere Vest

Jason P. Chang and Biondo Biondi

ABSTRACT

By performing multi-component seismic interferometry on passive seismic data from the Moere Vest ocean-bottom node survey, we observe three distinct wave modes from the resulting virtual source gathers: Scholte waves, guided acoustic waves, and critical refractions. Scholte waves are characterized by their dispersive nature, very low frequency content (below 0.5 Hz), slow propagation velocity (500 – 700 m/s), and relative clarity in the vertical-vertical correlations. Guided acoustic waves trapped between the sea surface and sea bottom are characterized by their dispersive nature, group velocity of approximately 1500 m/s, and relative clarity in the hydrophone-hydrophone correlations. Critical refractions in the vertical-vertical correlations are characterized by their non-dispersive linear moveout and high propagation velocity (4000 – 5000 m/s). Comparison of these events to critical refractions off the top of basalt in vertical-component receiver gathers from active data reveals similar arrival times and moveout velocities. To enhance the clarity of these refractions, we create virtual super-source gathers for both vertical-vertical and vertical-radial correlations. We find that the critical refraction arrives later in the vertical-radial gathers than in the vertical-vertical gathers, which is also observed in vertical- and radial-component receiver gathers from active data. Forward wavefield propagation suggests that these critical refractions could be generated by low-frequency Scholte waves scattering off the horst-and-graben structure of the top of basalt. Overall, these results lay the foundation for passive subsurface imaging using wave modes beyond interface waves.

INTRODUCTION

The application of seismic interferometry to recordings of Earth’s ambient seismic noise field has been shown to successfully estimate the Green’s function between pairs of receivers. While extraction of the interface-wave portion of the estimated Green’s function has been successful across many spatial scales in both land and marine environments (e.g., Shapiro et al., 2005; Bensen et al., 2008; de Ridder and Dellinger, 2011; Chang et al., 2016), there have been far fewer studies that have successfully extracted non-interface-wave portions. In land environments, Roux et al. (2005) and Gerstoft et al. (2008) retrieved diving P-waves at the regional scale. Additionally, Poli et al. (2012) retrieved P-wave reflections off the mantle transition zone, while Ruigrok et al. (2011) and Zhan et al. (2010) retrieved P- and S-wave reflections off the Moho, respectively. At the exploration scale, Nakata et al. (2015) were able to extract diving P-waves from a dense array at Long Beach, California, while Nakata et al. (2011) and Draganov et al. (2013) were able to extract reflection events in separate locations. In shallow marine environments, studies have been able to extract direct

P-waves traveling in the water column (e.g., Mordret et al., 2013; Chang, 2017), as well as sea-surface reflected and sea-bottom reflected events (e.g., Brooks and Gerstoft, 2009).

In this report, we focus on extracting both interface waves and non-interface waves from passive seismic data recorded by the Moere Vest deep-water ocean-bottom node (OBN) array. The work shown here builds on the initial analysis of the continuous recordings performed by de Ridder (2014). We begin with an overview of the Moere Vest OBN array. We then provide an overview of passive seismic interferometry (or ambient-noise cross-correlation technique) and apply it to passive seismic data below 2 Hz. The resulting virtual source gathers reveal three distinct wave modes:

- Scholte waves, which are commonly observed in ambient noise studies using OBN data;
- guided acoustic waves, which have previously only been observed at these water depths and frequencies in Hatchell and Mehta (2010); and
- critical refractions, which have previously never been observed in passive seismic data at this spatial scale.

Because of the rarity of two of these wave modes in passive seismic studies, we outline evidence supporting each of these classifications. We then simulate wave propagation through a simple elastic model to test a hypothesis for the generation of critical refractions from passive energy.

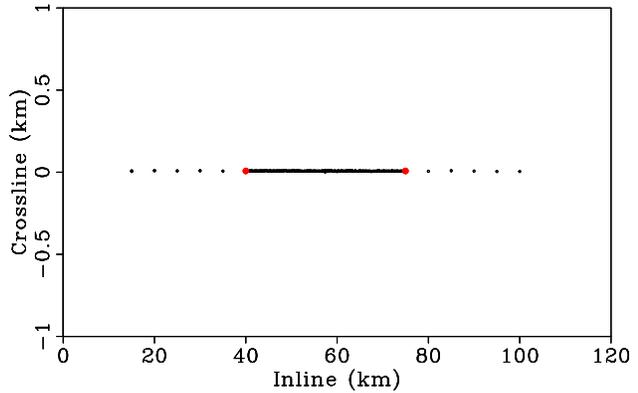
MOERE VEST CONTINUOUS RECORDINGS

The Moere Vest data set, provided by Seabed Geosolutions, consists of 179 four-component ocean-bottom nodes (OBNs) deployed in a line along the seabed in the North Sea and above a block of basalt (Figure 1). The array is notable for its spatial extent (85 km maximum offset between nodes) and deep-water environment (approximately 1.6–1.9 km depth). The nodes consist of three different geometries: 141 nodes spaced 250 m apart and spanning 35 km that form the regular spread; 12 nodes spaced 5 km apart at each end of the regular spread to increase the total length of the array to 85 km; and 26 nodes spaced 2 m apart in two parallel lines along the regular spread. Each node records continuously at 2-ms sampling for approximately 10 days, with all nodes simultaneously recording for 7 of those days.

Within the 7 days of simultaneous recordings were periods of active seismic shooting. This is apparent in the log-spectrogram of the vertical-geophone recording for the node on the western edge of the regular spread, where the spectral comb signature of active sources is dominant above 2 Hz (Figure 2(a)). While the effect of active seismic shooting does not appear to interfere with the frequencies of interest in this study (below 2 Hz), we remove them from further analysis to ensure that the results shown here are derived from strictly passive seismic data. Removal of times of active seismic surveys results in about 3 days of simultaneous passive recordings. The resulting log-spectrogram indicates strong microseism energy below 2 Hz (Figure 2(b)). We look to extract coherent wave modes from this energy using passive seismic interferometry.

Figure 1: Map of the Moere Vest OBN array. Red points indicate nodes used for spectrograms and virtual source locations. [CR]

`jsonpc1/. map-all`



PASSIVE SEISMIC INTERFEROMETRY

Passive seismic interferometry produces an estimate of the Green’s function between two receivers by cross-correlating their continuous and simultaneous recordings of ambient seismic noise (Wapenaar et al., 2010). Here, we apply processing steps similar to those outlined in Bensen et al. (2007) to the hydrophone and geophone components. First, we rotate the horizontal components of each geophone such that one component is inline with the array and the other component is orthogonal to the array using the method from Alves (2017). We then divide each continuous recording into two-hour windows with 50% overlap (Seats et al., 2012) and remove the time windows containing active seismic shooting (resulting in 78 time windows spanning over 3 days). Next, we perform passive seismic interferometry by calculating the average of the whitened coherency between each pair of nodes across all time windows. This procedure is also referred to as calculating the cross-coherence, and it typically produces a broader-band signal than standard cross-correlation while maintaining phase information. In the frequency domain, the procedure is generally expressed as:

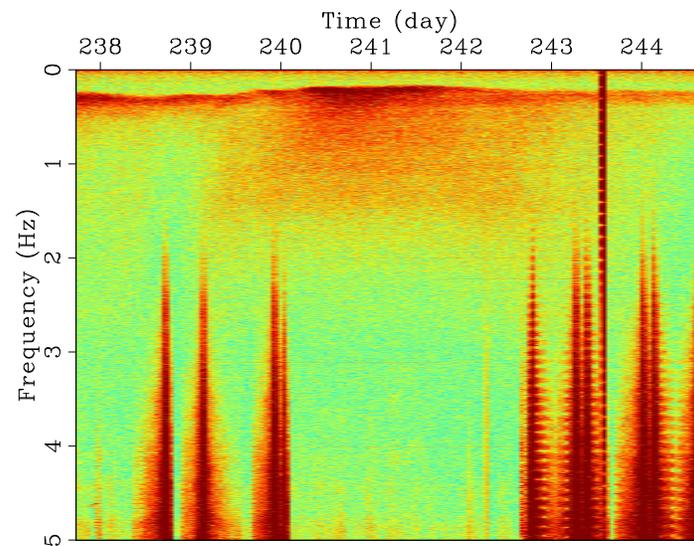
$$[G(x_B, x_A, \omega) + G^*(x_B, x_A, \omega)] = \left\langle \left(\frac{U(x_B, \omega)}{\{|U(x_B, \omega)|\}} \right) \left(\frac{U^*(x_A, \omega)}{\{|U(x_A, \omega)|\}} \right) \right\rangle, \quad (1)$$

where G is the Green’s function between two receiver locations (x_A, x_B) , $U(x, \omega)$ is the Fourier transform of the wavefield at a given receiver location x , $*$ is the complex conjugate, $\langle \cdot \rangle$ is an averaging operation, $|\cdot|$ is the magnitude of the spectrum, and $\{\cdot\}$ is a 0.003 Hz running window average used for normalizing the signal.

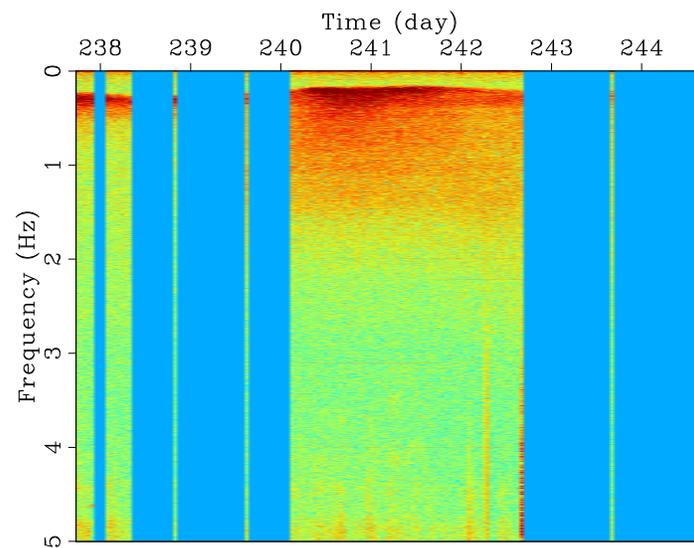
By cross-correlating the recording from one node with recordings from all other nodes, we can create a virtual source gather at any node in the array. For this study, we focus on correlations from the hydrophones and vertical and radial components of the geophones. Because of the effects of attenuation, we only investigate frequencies below 2 Hz, which is reasonable for such a wide-aperture array.

WAVES MODES IN VIRTUAL SOURCE GATHERS

Virtual source gathers derived from passive seismic data reveal three different wave modes: Scholte waves, guided acoustic waves, and critical refractions. We outline evidence for each of these classifications in the following subsections.



(a)



(b)

Figure 2: Log-spectrograms for the vertical-component geophone indicated by the western red point in Figure 1. (a) Spectrogram for the entire overlapping period of continuous recordings. Active seismic shooting is distinguished by high amplitudes and comb patterns above 2 Hz. (b) Spectrogram with times of active seismic shooting removed. Warmer colors correspond to higher spectral power. All results in this report use only times without active seismic shooting. [CR] `jasonpc1/. sp-full,sp-keep`

Scholte waves

Scholte waves are interface waves that travel along fluid-solid interfaces. Figure 3 shows virtual source gathers for frequencies between 0.2 – 0.4 Hz for two virtual source locations: one at the western edge of the regular array and one at the eastern edge of the regular array (red dots in Figure 1). The slow event propagating in each gather at a group velocity of approximately 450 m/s (most apparent in the left column of Figure 3) appears to be a Scholte wave for a number of reasons.

First, this event is much stronger in the vertical-vertical (*ZZ*) correlations (left column; Figure 3) than in the hydrophone-hydrophone (*PP*) correlations (right column; Figure 3). Since geophones are coupled to the ground while hydrophones are coupled to the water, we expect any Scholte wave propagating along the seabed to be stronger in *ZZ* correlations than in *PP* correlations. The stronger, faster-propagating event in the *PP* correlations is likely an acoustic guided wave (addressed in the following subsection). Second, interface waves typically dominate Earth’s ambient seismic noise field at these very low frequencies (microseism band), and as a result Scholte waves are commonly the dominant arrival in passive seismic data collected at the ocean bottom (e.g., Mordret et al., 2013; de Ridder and Biondi, 2015). Third, Scholte waves are commonly observed to propagate at velocities on the order of 100 m/s (e.g., de Ridder and Dellinger, 2011; de Ridder et al., 2015), which is on the order of the group velocities seen here.

Another common characteristic of Scholte waves is that they are dispersive. We create a dispersion image by performing slant stacks for different intercept times and moveout velocities on a virtual source gather, and then performing a Fourier transform along the time axis to produce an image in the frequency-velocity domain. The dispersion image associated with the *ZZ* virtual source gather in Figure 3(a) is shown in Figure 4. The phase velocity appears to decrease as frequency increases, which is a typical characteristic of Scholte waves (shorter wavelengths sample slower, shallower depths). Given this dispersion image, it appears that the wavelengths vary approximately between 1250 m at 0.4 Hz and 3000 m at 0.2 Hz, which are long enough to potentially reach the top of basalt approximately 1 km below the sea bottom.

The previously outlined evidence suggests that these low-velocity, low-frequency events are Scholte waves. From the *ZZ* correlations (left column; Figure 3), it is clear that the probable Scholte-wave energy is strongest at positive time lags when the virtual source is in the west of the array and at negative time lags when the virtual source is in the east of the array. This indicates that the Scholte-wave energy is propagating primarily from west to east. These Scholte waves can potentially be used to estimate shear-wave velocity depth profiles.

Guided acoustic waves

Guided acoustic waves consist of constructively-interfering, post-critical P-wave reflections trapped between the sea surface and seabed (Burg et al., 1951). They are commonly observed over a broad range of frequencies in active seismic marine surveys where the water depth is shallow and the sea bottom is hard (e.g., Shtivelman, 2004; Klein et al., 2005; Boiero et al., 2013).

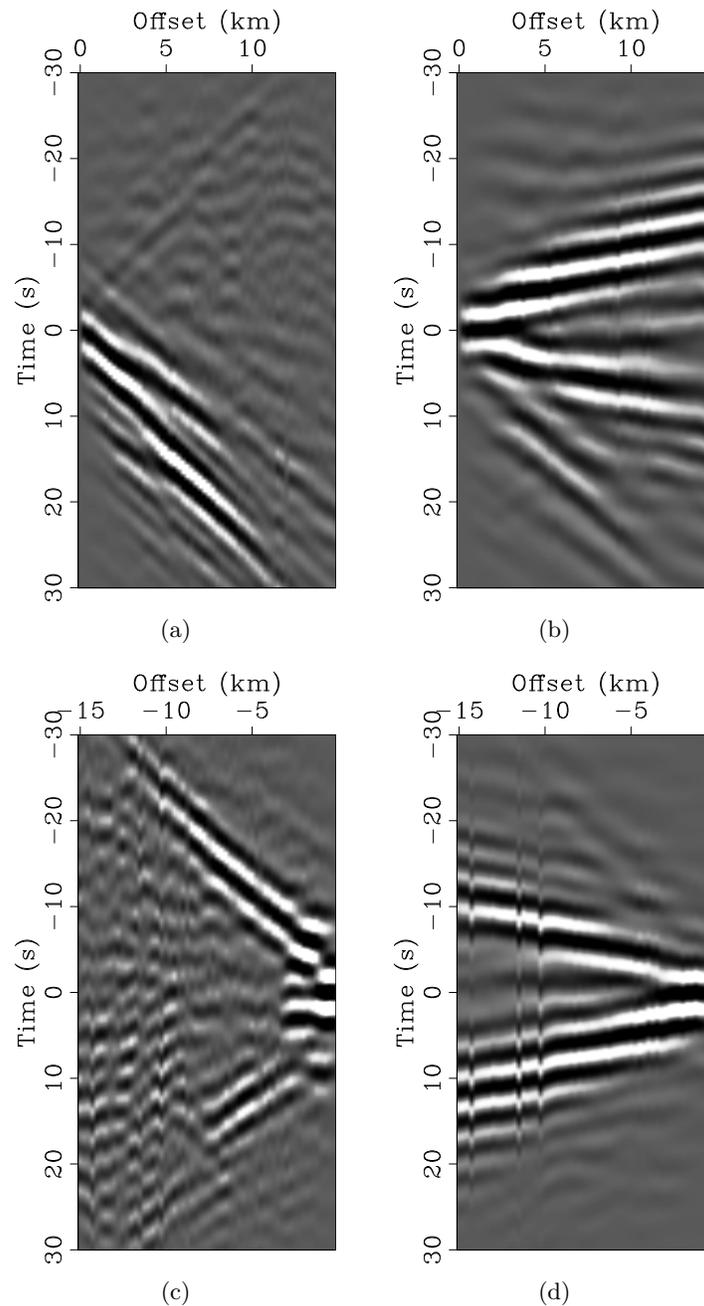


Figure 3: Virtual source gathers for frequencies between 0.2 – 0.4 Hz. Positive offsets are toward the east while negative offsets are toward the west. Top row: virtual source at the western red dot in Figure 1. Bottom row: virtual source at the eastern red dot in Figure 1. Left column: vertical-vertical geophone correlations. Right column: hydrophone-hydrophone correlations. The strong events in the vertical-vertical geophone correlations are Scholte waves, and their asymmetry suggests they are primarily propagating west to east. Though the same Scholte waves are apparent in the hydrophone-hydrophone correlations, they are faint compared to the faster-propagating acoustic guided waves. [CR] `jasonpc1/.scholte-40-zz,scholte-40-hh,scholte-75-zz,scholte-75-hh`

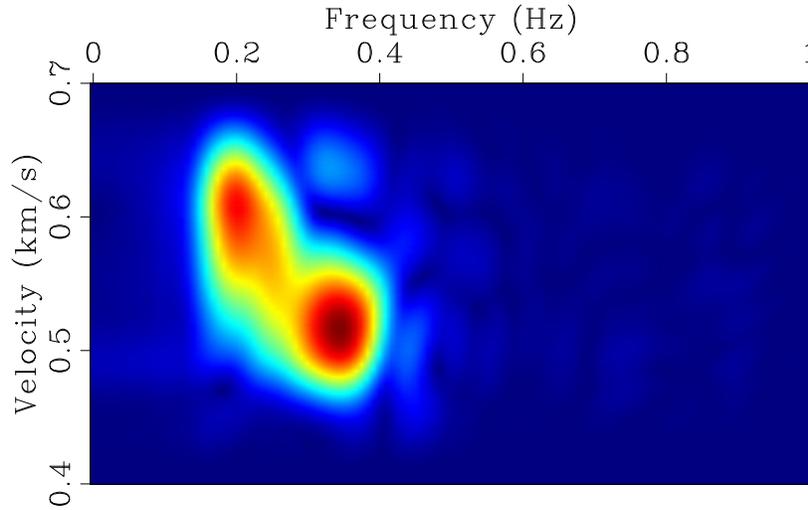


Figure 4: Dispersion image for the virtual source gather centered at the western node for the vertical-vertical components (Figure 3(a)). Warmer colors correspond to greater coherence of the dispersion estimate. [CR] [jasonpc1/. dispersion-z-scholte](#)

Figure 5 shows virtual source gathers for frequencies between 0.4 – 1.5 Hz for the same two virtual source locations as in Figure 3. The strong, consistent arrivals in each of the panels are likely guided acoustic waves for a number of reasons. First, the group velocity of these arrivals is approximately 1500 m/s, which is the velocity of acoustic waves in water. Given the frequency range here, the velocity is much too high to be associated with Scholte waves. Furthermore, the velocity of these events appears to be the same as the dominant events in the PP correlations at frequencies below 0.4 Hz (right column; Figure 3). Second, the signal-to-noise ratio for this event appears to be higher in the PP correlations (right column; Figure 5) than in the ZZ correlations (left column; Figure 5), and this same event dominates the PP correlations at frequencies below 0.4 Hz (right column; Figure 3). Since hydrophones are coupled to the water and geophones are coupled to the ground, we expect any waves propagating in the water to be stronger in the PP correlations than in the ZZ correlations.

In terms of kinematics, acoustic guided waves are dispersive and their phase velocities exceed water velocity (Shtivelman, 2004). The background image in Figure 6 shows that the dispersion image derived from the PP virtual source gather in Figure 5(b) shares these traits. The event is clearly dispersive, as phase velocity for a given mode exceeds water velocity as frequency decreases and approaches water velocity as frequency increases. Additionally, this event displays multiple modes, which has been observed in acoustic guided waves in a number of studies (e.g., Klein et al., 2005; Boiero et al., 2013; Wang et al., 2016).

To further support the hypothesis that these events are acoustic guided waves, We compare the observed dispersion image to theoretical dispersion curves calculated using the following approximation from Hatchell and Mehta (2010) for guided waves in a liquid layer

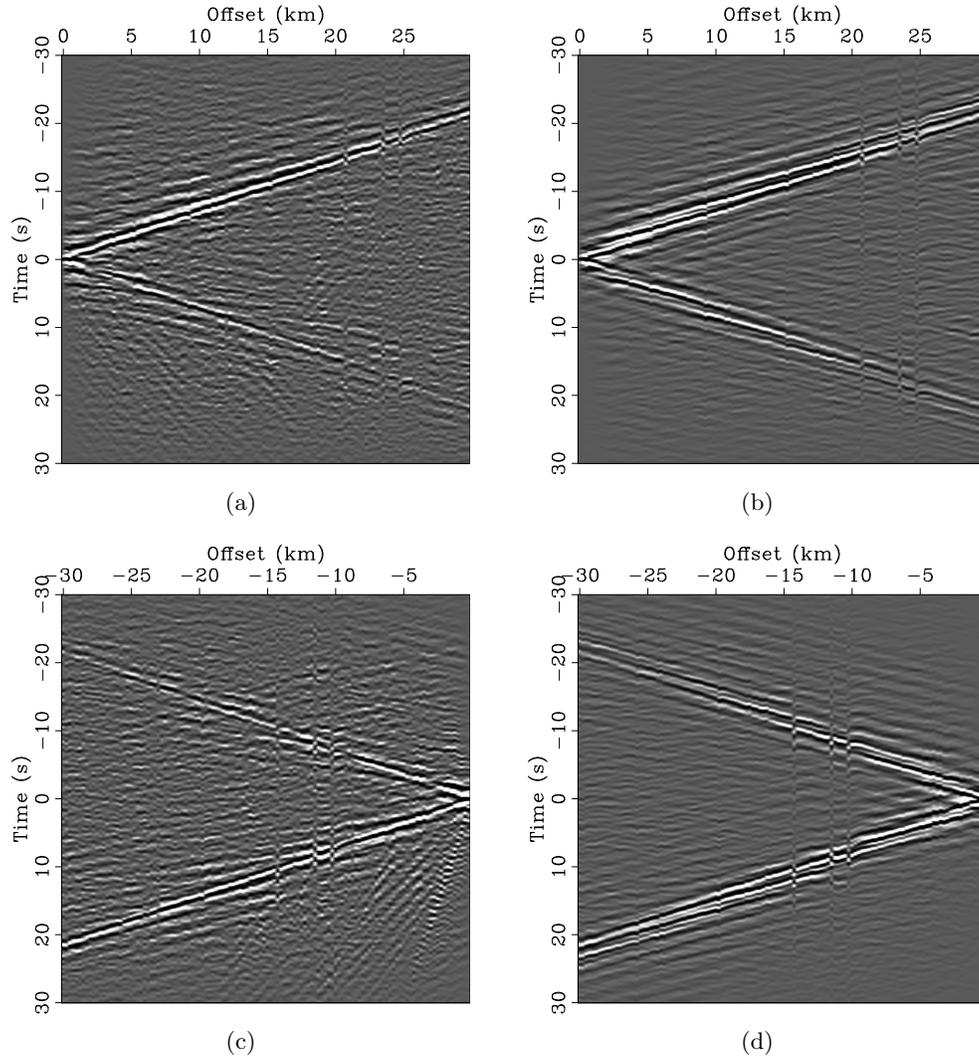


Figure 5: Virtual source gathers for frequencies between 0.4 – 1.5 Hz. Positive offsets are toward the east while negative offsets are toward the west. Top row: virtual source at the western red dot in Figure 1. Bottom row: virtual source at the eastern red dot in Figure 1. Left column: vertical-vertical geophone correlations. Right column: hydrophone-hydrophone correlations. The strong events in all virtual source gathers are guided acoustic waves propagating at approximately 1500 m/s. Note that the signal-to-noise ratio is much higher in the hydrophone-hydrophone correlations. [CR] `jasonpc1/. guided-40-zz,guided-40-hh,guided-75-zz,guided-75-hh`

over a solid:

$$\tan \left[\frac{2\pi h f}{c} \sqrt{\frac{c^2}{v_1^2} - 1} \right] = -\frac{\rho_2 \sqrt{\frac{c^2}{v_1^2} - 1}}{\rho_1 \sqrt{1 - \frac{c^2}{v_2^2}}}, \quad (2)$$

where h is water depth, ρ_1 and ρ_2 are the densities of water and seafloor, respectively, f is frequency, v_1 is water velocity, v_2 is P-wave velocity of the seafloor, and c is guided-wave phase velocity. Solutions for phase velocity c and frequency f given $h = 1800$ m, $\rho_1 = 1000$ kg/m³, $\rho_2 = 1800$ kg/m³, $v_1 = 1500$ m/s, and $v_2 = 2300$ m/s (reasonable parameters given Alves (2017)) are shown in white in Figure 6. The theoretical and observed dispersion curves match quite well, further suggesting that these events are guided acoustic waves.

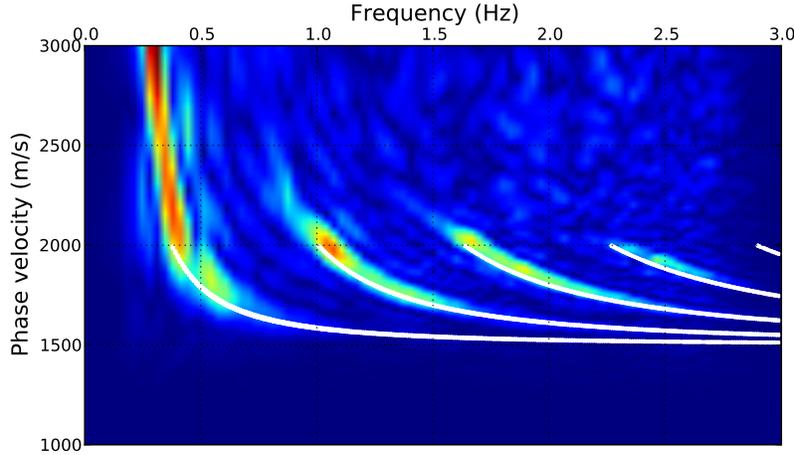


Figure 6: Dispersion image for the virtual source gather centered at the western node for the hydrophone-hydrophone correlations (Figure 5(b)). A gain proportional to frequency is applied to enhance the higher modes. Warmer colors correspond to greater coherence of the dispersion estimate. The overlying white lines indicate theoretical dispersion curves using Equation 2 and the parameters outlined in the text. Note the similarity between the observed and theoretical dispersion curves. [CR] `jasonpc1/. dispersion-h-guided`

While commonly observed in shallow marine seismic surveys, guided acoustic waves have rarely been observed in passive seismic data at these frequencies, this spatial scale, and these water depths. One such case was Hatchell and Mehta (2010), who used passive seismic interferometry to recover guided acoustic waves from continuous seismic data recorded by OBNs 1000 m deep and with maximum offset of 9600 m. As seen with the theoretical dispersion curve calculation, though these acoustic guided waves propagate in the water, they interact with the seabed and are thus sensitive to elastic properties of the subsurface material (Klein et al., 2005). It is possible to go beyond the simple half-space P-wave velocity model estimated here by inverting these acoustic guided wave dispersion curves for P-wave velocity depth profiles of the subsurface (Klein et al., 2005; Boiero et al., 2013).

Critical refractions

Critical refractions occur where there are interfaces with sharp impedance contrasts. At Moere Vest, Alves (2017) identified refractions off the top of basalt using the active data. Here, we examine the passive data for similar critical refractions. The top row of Figure 7 shows time-symmetrized ZZ virtual source gathers for frequencies between 0.5 – 2.0 Hz for the same two virtual source locations as in Figures 3 and 5. Though faint, there appear to be very fast linear events in both gathers (the stronger events are the acoustic guided waves). The same events in the PP virtual source gathers (not shown here) are even fainter. Because vertical geophones are coupled to the ground and not water, this event is likely propagating in the subsurface. This observation, along with the apparent velocity of the event and knowledge of the subsurface basalt, hints that this event is potentially a critical refraction.

The simplest way to determine whether these events from passive seismic data are critical refractions is to compare them to active-source data. Before doing so, differences in source location must be accounted for: virtual sources are located at nodes on the sea bottom, while active sources are located near the sea surface. To get two gathers to correspond to the same location, we reference the virtual source gather from passive data and the receiver gather from active data to a common node. To get the gathers to start at the same time, we shift the receiver gather from active data by 1 s. Given that water depth at Moere Vest is approximately 1600 – 1900 m, this shift removes the initial traveltimes from sea-surface source to sea-bottom node and effectively places the active sources near the sea bottom. Though the angle of incidence of the initial path through the water column depends on source-receiver offset, we assume that it does not vary significantly from normal incidence due to ray bending from the high impedance contrast between the seabed and water.

Active-source vertical-geophone receiver gathers centered at the western and eastern nodes of the regular spread are shown in the bottom row of Figure 7. We bandpass for frequencies between 2 – 6 Hz, apply a linear gain proportional to offset, and heavily clip the gathers to enhance the critical refractions. Comparison of these events to the fast linear events in the corresponding ZZ virtual source gathers in the top row of Figure 7 reveals similar arrival times and moveout velocities. The similarity not only includes the first linear arrivals, but also the later linear arrivals, which may be multiple-related critical refractions.

To enhance the stacking power of the potential critical refraction events in the passive data, it is common to stack over all virtual source gathers to create a super-source gather (Lin et al., 2013; Nakata et al., 2015). The trade-off is that all spatial information is lost, as the process effectively assumes a 1D medium. The ZZ super-source gather (Figure 8(a)) reveals stronger critical refractions than in the individual virtual source gathers (top row; Figure 7). These events are apparent at both negative and positive times, as well as both positive and negative offsets, which indicates that the potential refractions are propagating in both directions along the array. The symmetry of these events in time and space suggest that they are not generated by localized sources, and that their fast velocities are not just apparent velocities from waves hitting the array broadside. The vertical-radial (ZR) super-source gather (Figure 8(b)) also displays hints of a refraction-like event that was not apparent in corresponding individual virtual source gathers (not shown in this report). Compared to the ZZ super-source gather, the fast linear events arrive later in time and are only apparent at positive time lags. The latter observation could provide insight into how

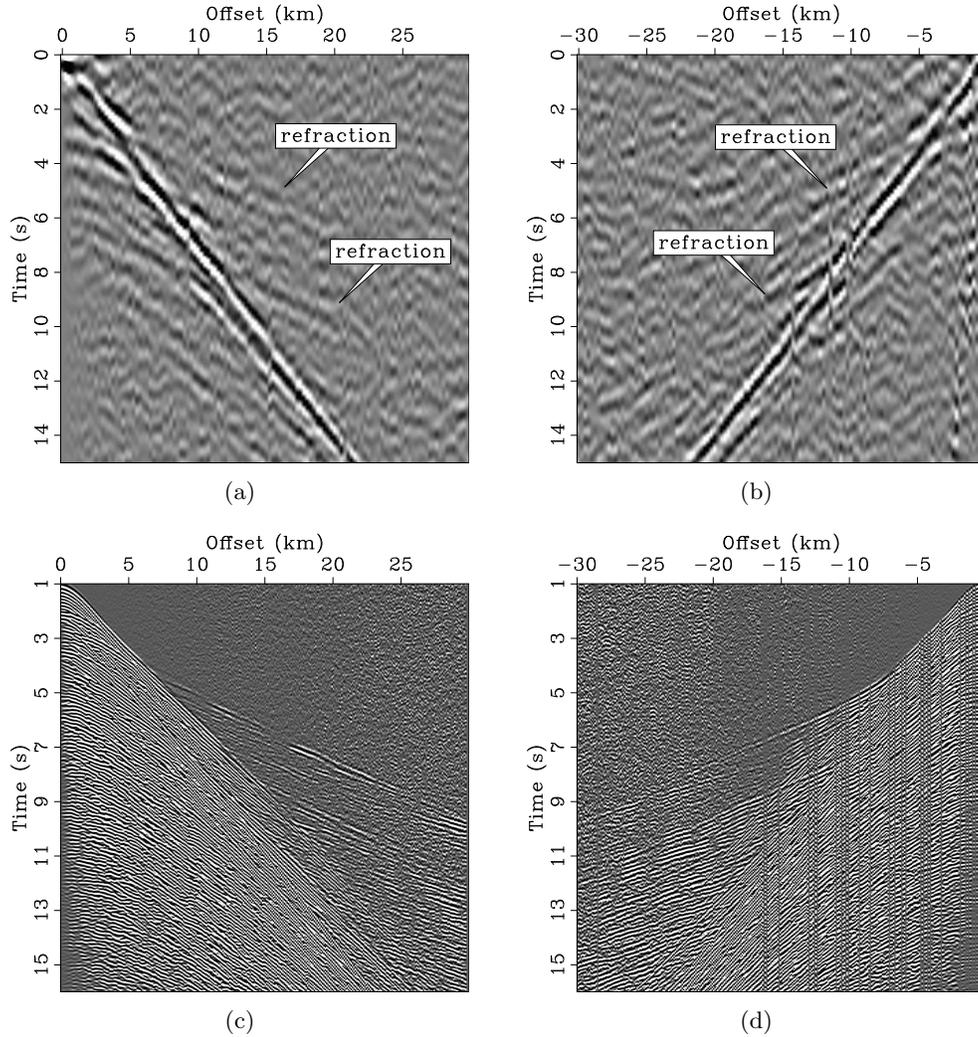


Figure 7: Comparison of virtual source gathers from passive data and receiver gathers from active-source data . Left column: western node in Figure 1. Right column: eastern node in Figure 1. Top row: virtual source gathers between vertical-vertical components for frequencies between 0.5 – 2.0 Hz. Gathers are symmetrized over zero-time lag to enhance fast linear events. Bottom row: vertical-component receiver gathers from active data for frequencies between 2 – 6 Hz. A strong clip and a gain proportional to offset are applied to enhance refractions, and the gathers are shifted in time by 1 s so they are comparable to the virtual source gathers. The arrival times of the refraction events in the passive and active gathers coincide well. [CR] `jasonpc1/. refraction-40-zzA,refraction-75-zzA,active-40-zz-half,active-75-zz-half`

these potential critical refractions are generated in Earth’s ambient seismic noise field, as they suggest that refractions only appear when passing the vertical geophone before the radial geophone (and not vice versa).

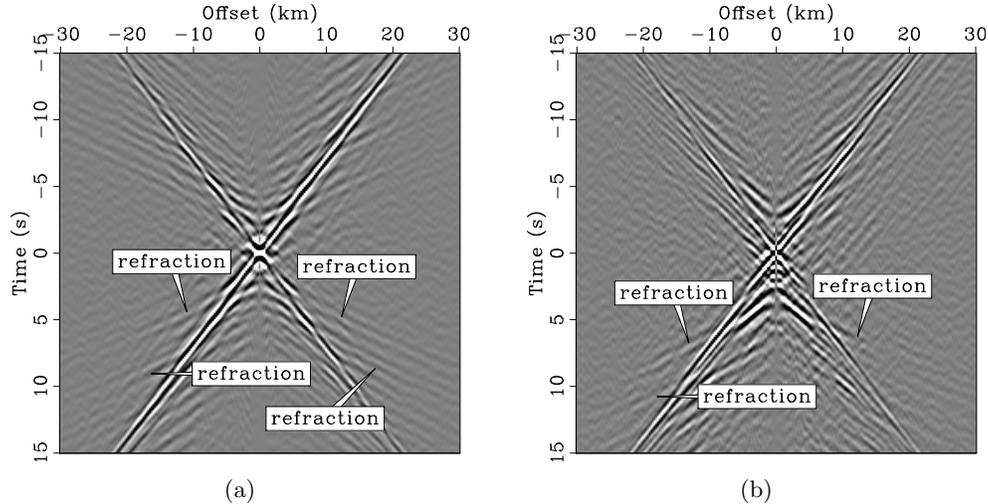


Figure 8: Virtual super-source gathers for frequencies between 0.5 – 2.0 Hz. (a) Vertical-vertical geophone correlations. (b) Vertical-radial geophone correlations. Note the difference in the first refraction arrival times between the two gathers, particularly at positive time lags. [CR] `jasonpc1/. super-zz-unsymA,super-zr-unsymA`

To further examine the difference in arrival times of the potential refraction in the super-source gathers, we compare their positive time lags to vertical- and radial-component receiver gathers from active data (Figure 9). Though the super-source gathers do not correspond to a specific spatial location and these particular receiver gathers are centered at the western node of the regular array, it is an adequate comparison for investigating the relative arrival times of the critical refraction between different geophone components. The arrival times of the potential refraction event in the ZZ super-source gather and vertical-component receiver gather are similar (left column; Figure 9). Using these arrival times as a reference, it is clear that the fast linear events arrive later in the ZR super-source gather and the radial-component receiver gather by similar amounts of time (right column; Figure 9). Moronfroyer et al. (2016) hypothesized that the difference in arrival times of the refraction in the two orthogonal components was due to differences in subsurface P-wave and S-wave velocities. The vertical component recorded the faster P-wave refraction while the radial component recorded the slower converted S-wave refraction. Regardless of the true reason for the differences in refraction arrival times, the fact that fast linear events in the ZZ and ZR super-source gathers display similar arrival time differences as in the active data is another piece of evidence supporting the classification of these events as critical refractions.

To compare the velocities of the apparent refractions in the passive and active data, we perform linear moveout on the gathers in Figure 9. Results using a moveout velocity of 4400 m/s are shown in Figure 10. The apparent refraction in the super-source gathers and receiver gathers (at least at positive offsets) appear to be flattened, thus providing another piece of evidence that we are recovering critical refractions in the passive data. Note that

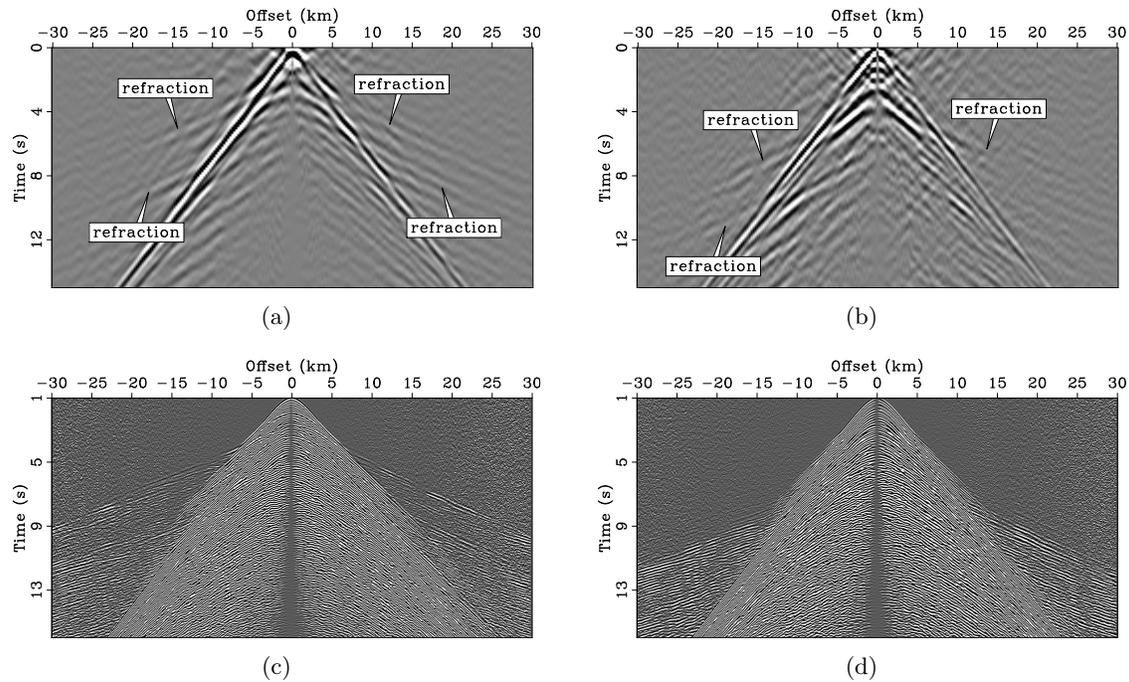


Figure 9: Comparison of virtual super-source gathers from passive data to receiver gathers from active-source data. Positive time lags of virtual super-source gather for frequencies between 0.5 – 2.0 Hz: (a) vertical-vertical components, and (b) vertical-radial components. Receiver gathers from active-source data centered at the western node in Figure 1 for frequencies between 2 – 6 Hz: (c) vertical component, and (d) radial component. A strong clip and a gain proportional to offset are applied to enhance refractions, and the gathers are shifted in time by 1 s so they are comparable to the virtual source gathers. Note the later refraction arrival times when looking at gathers involving the radial component. [CR]

`jasonpc1/. super-zz-symA,super-zr-symA,active-40-zz,active-40-zr`

the refractions in the receiver gathers are not quite flat at negative offsets (where sources are west of the node); they require moveout with a higher velocity to be flattened. This could be because the basalt tilts slightly downward toward the west, and apparent refraction velocities are typically slower shooting down-dip (toward the west) than up-dip (toward the east).

This trend is not observed in the super-source gathers. This is potentially because the negative offsets in the super-source gathers are sensitive to a different part of the basalt than the negative offsets of the receiver gather centered at the western node of the regular array. Because of the relatively large number of nodes in the regular section of the array, both offsets of the super-source gathers are biased towards the refraction response off the basalt directly beneath that particular subset of the array. While the positive offsets of the receiver gather centered at the western node of the regular array sense the same refraction off the basalt beneath the regular array, the negative offsets sense the refraction off the basalt that is west of the regular array. As a result, the negative offsets of the receiver gather are not flattened at 4400 m/s while the positive offsets and super-source gathers are.

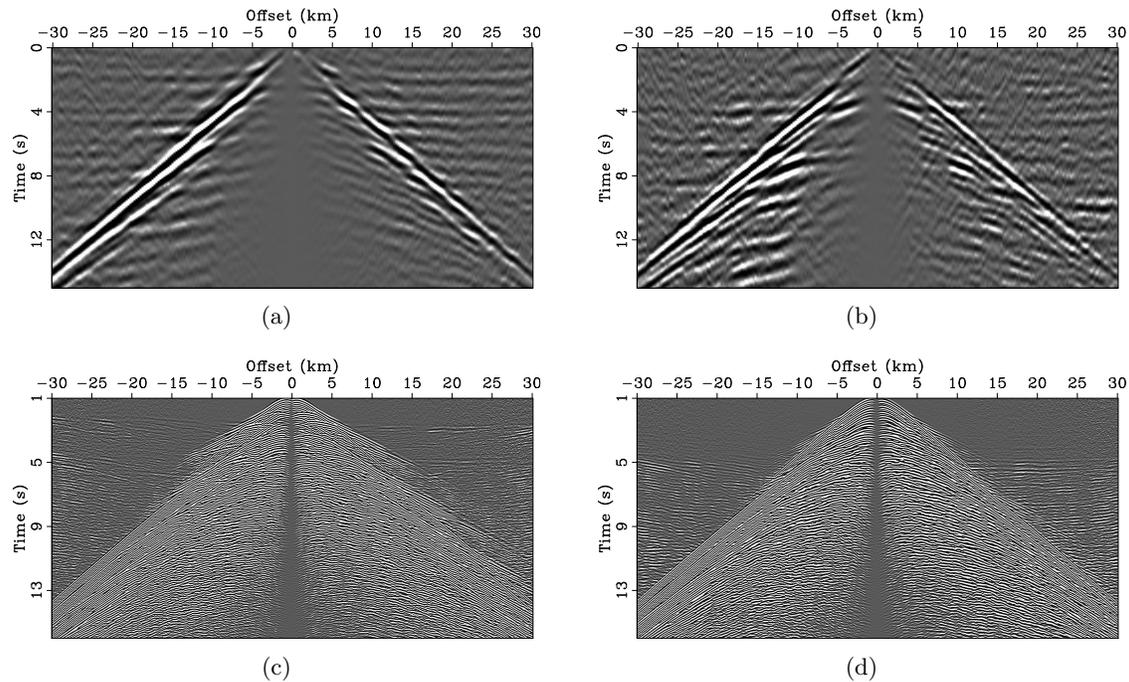


Figure 10: Linear move-out of the gathers in Figure 9 with a velocity of 4400 m/s. Note how the refraction events in all gathers are flattened. [CR] jasonpc1/. super-zz-sym-lmo,super-zr-sym-lmo,active-40-zz-lmo,active-40-zr-lmo

Overall, there is evidence that we are extracting critical refractions from passive data, particularly when comparing these events to active data. To the author's knowledge, these types of events from passive data have never been observed before. They are observed at Moere Vest for a couple of potential reasons. One is that the array provides sufficiently long offsets for potential refractions to become well-separated from other events, such as guided acoustic waves here. Another reason is that there is a basalt body in the subsurface, which is likely to produce the high-impedance interface needed for critical refractions to be generated. We investigate one hypothesis for the generation of critical refractions in passive

energy in the next section.

MODELING OF CRITICAL REFRACTIONS

While we appear to observe critical refractions at Moere Vest, the mechanism for their generation is not yet understood. Knowing that Earth’s ambient seismic field is dominated by interface waves, one hypothesis is that we are observing Scholte waves that are interacting with the top of basalt and subsequently converting or scattering into critical refractions and diffractions. Given it appears that the Scholte-wave wavelengths vary approximately between 1250 m at 0.4 Hz and 3000 m at 0.2 Hz, it is possible that they penetrate deep enough to sense the basalt’s horst-and-graben structure approximately 1 km beneath the sea bottom. To test this hypothesis, we simulate wavefield conversions using a staggered-grid elastic finite-difference wave propagation code (Fabien-Ouellet et al., 2017) on a simplified 2D elastic model of Moere Vest consisting of three layers: a water layer, an overburden layer, and a basalt layer with a 100-m thick step feature (Figure 1). Specific values for the elastic parameters were estimated from Alves (2017) and can be found in Table 1. The model spans 1500 cells in the horizontal direction and 400 cells in the vertical direction, and has a grid spacing of 100 m. Because we are interested in observing the interaction of low-frequency Scholte waves with the simple step feature of the top of basalt, we place an explosive source with 0.4 Hz central frequency 50 km away from the step feature and 100 m below the sea bottom so that the Scholte wave is separated from other wave modes (refractions and direct waves) by the time it reaches the step feature. We also place absorbing boundary conditions along all sides of the model, as we want to reduce interference from waves reflecting off the sea surface. We run the propagation for 12500 time steps of size 0.008 s.

Layer	Thickness [m]	V_p [m/s]	V_s [m/s]	Density [kg/m ³]
Water	1500	1500	0	1000
Overburden	1000	2300	1400	1800
Basalt	infinite	5000	3000	2500

Table 1: Model parameters for the simple three-layer model with a 100-m thick step feature at the top of basalt shown in Figure 1.

Figure 12(a) shows the resulting vertical-velocity source gather recorded by receivers along the water bottom. While there are a number of arrivals in the gather, many of them can be grouped together by apparent moveout velocity and are likely to be related to trying to excite a very low-frequency source in a relatively thin layer. To help identify these events, we include a snapshot of the vertical-velocity wavefield at 24 s in Figure 13. The fastest group of arrivals are likely related to the critical refraction off the top of basalt as they are traveling at approximately 5000 m/s (150000 m/30 s), which is the P-wave velocity of the basalt layer. The slower bound of this group of arrivals is likely the direct P-wave as it is propagating at approximately 2500 m/s (150000 m/60 s), which is close to the P-wave velocity of the overburden. Additionally, the corresponding event at 60 km in Figure 13 extends through the entirety of the basalt, further suggesting that this is a direct P-wave. The slower group of arrivals is likely related to Scholte waves. Though their moveout velocities appear to be approximately 1400 m/s (140000 m/100 s), which is the S-wave velocity in the overburden, the corresponding wavefield trapped near the surface around 24 km in Figure 13 suggests that these arrivals are related to interface waves.

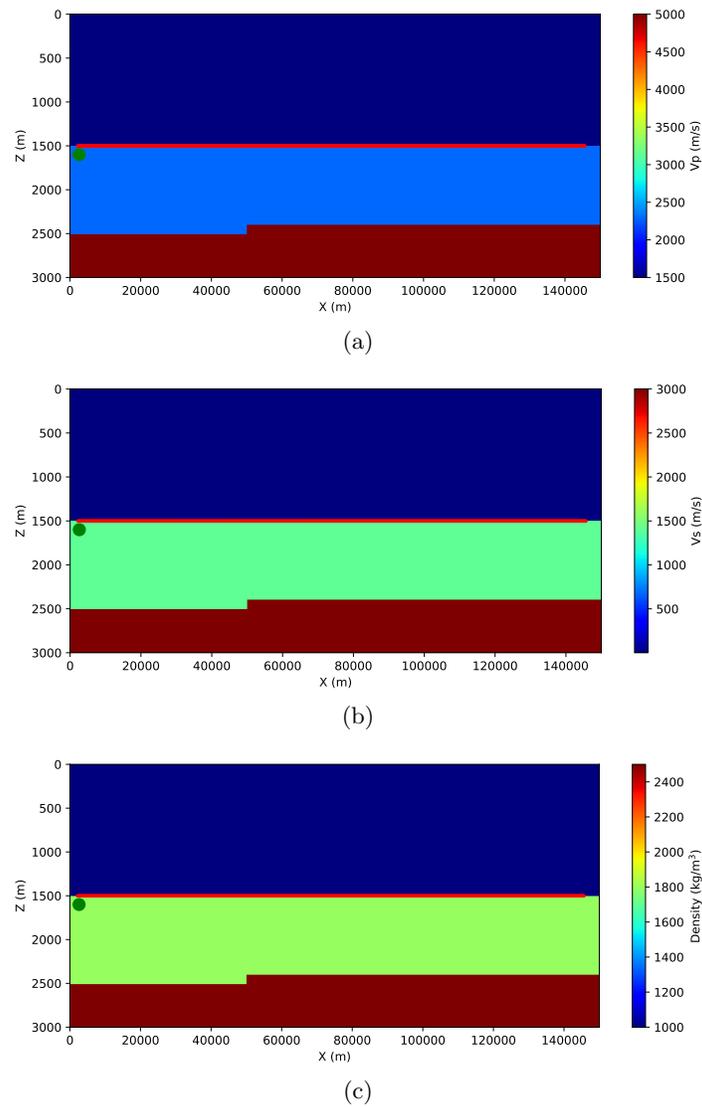


Figure 11: Model used for generating synthetic source gathers. The simple three-layer model consists of a water layer, an overburden, and a basalt body with a step feature. (a) P-wave velocity. (b) S-wave velocity. (c) Density. Red dots represent receiver locations, while green dots represent source locations. Note that vertical and horizontal axes are not to scale in order to enhance the step feature in the basalt. [CR] `jasonpc1/. step-vp,step-vs,step-rho`

Assuming the slower packet of arrivals is related to Scholte waves, there is apparent forward- and back-scattered energy at the location of the step feature (50 km). To better examine the scattered events, we subtract the source gather associated with the basalt with the step feature (Figure 12(a)) from a source gather associated with the basalt without the step feature (not shown here). While there is similar forward- and back-scattering energy associated with the faster refraction packet and the slower Scholte-wave packet (Figure 12(b)), we focus on the scattered events associated with the Scholte wave. As expected, there is back-scattered energy with a similar moveout velocity as the incident Scholte wave; this is likely a back-scattered Scholte wave. More importantly, there is both forward- and back-scattered energy with similar moveout velocity as the earlier-arriving incident critical refractions. This seems to suggest that a Scholte wave interacting with a step feature as thin as 100 m can produce forward- and back-scattered critical refractions. Thus, it is possible that a low-frequency Scholte wave interacting with the horst-and-graben structure of the top of basalt could produce critical refractions along both directions of the array.

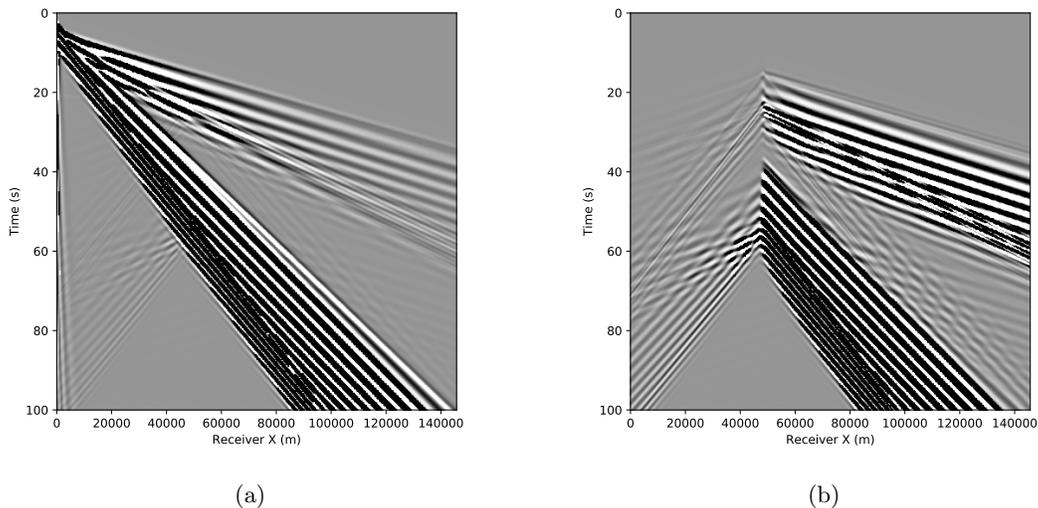


Figure 12: (a) Vertical-velocity source gather associated with the model in Figure 1. (b) Difference between vertical-velocity source gathers with and without the step feature at the top of basalt. Note the back-scattered energy from the step feature appears to be comprised of both interface waves and refractions. The source wavelet is a Ricker wavelet with 0.4 Hz central frequency. [CR] `jasonpc1/. step-vz,diff-vz`

CONCLUSIONS

Using seismic interferometry, we found that the continuous seismic recordings from the Moere Vest OBN array contained apparent Scholte waves, guided acoustic waves, and critical refractions. While Scholte waves are commonly extracted from Earth's ambient seismic noise field, the latter two wave modes are not. Acoustic guided waves are commonly observed in shallow-water active seismic surveys, but had rarely been observed in passive data at low frequencies and deep depths. At Moere Vest, we found that the dispersion image

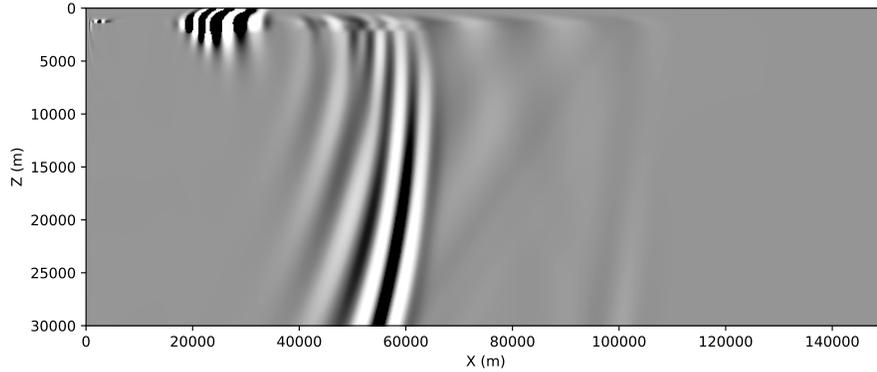


Figure 13: Snapshot of the vertical-velocity wavefield through the model in Figure 1 at 24 s. Note the apparent refraction at 100 km, the apparent direct P-wave at 60 km, and the apparent Scholte wave at 24 km. Though the top of basalt is at 2.5-km depth, the displayed depth goes down to 30 km to more easily observe the characteristics of the wavefield. Note that the horizontal and vertical axes are not to scale. [CR] `jasonpc1/. step-mov-vz`

for the passive acoustic guided wave had similar characteristics as those from typical active seismic surveys. Much like Scholte waves, these dispersion curves can be inverted for depth profiles of seismic velocities. Even more rare are critical refractions, which had never been observed in passive seismic data before. We found that the arrival times of potential critical refractions from passive data compared well to those from active data. It is likely that the length of the array, the presence of a high-impedance interface at the top of basalt, and the horst-and-graben structure of the top of basalt allows these events to be visible.

Future work will focus on understanding the mechanism of these critical refractions and using them for subsurface imaging. Of primary interest is further understanding how Earth’s ambient seismic field could generate these events. While the forward modeling of the wavefield here supports the primary hypothesis that we are observing deep-penetrating Scholte waves that converted into critical refractions off the horst-and-graben structured top of basalt, there are other possible hypotheses to test. One such hypothesis is that these critical refractions are due to the interaction of acoustic guided waves with the top of basalt. In addition to further modeling the critical refractions, we also intend to use these wave modes to estimate subsurface properties. For the Scholte and acoustic guided waves, the associated dispersion curves could be inverted to obtain depth profiles of $V_p - V_s$ ratios. For the critical refractions, we will need to investigate processing techniques (such as phase-weighted stacking or creating a selection filter as in Nakata et al. (2015)) to enhance these arrivals in individual virtual source gathers in hopes they may then be used for body-wave imaging.

ACKNOWLEDGEMENTS

The authors would like to thank Seabed Geosolutions for use of this data set. We would also like to thank Robert Clapp, Shuki Ronen, Gabriel Fabien-Ouellet, and Gustavo Alves for helpful discussions and insight. This work used the XStream computational resource,

supported by the National Science Foundation Major Research Instrumentation program (ACI-1429830).

REFERENCES

- Alves, G., 2017, Elastic Full Waveform Inversion of the Moere Vest data: SEP-Report, **168**, 63–76.
- Bensen, G., M. Ritzwoller, M. Barmin, A. Levshin, F. Lin, M. Moschetti, N. Shapiro, and Y. Yang, 2007, Processing seismic ambient noise data to obtain reliable broad-band surface wave dispersion measurements: *Geophysical Journal International*, **169**, 1239–1260.
- Bensen, G., M. Ritzwoller, and N. Shapiro, 2008, Broadband ambient noise surface wave tomography across the United States: *Journal of Geophysical Research: Solid Earth*, **113**.
- Boiero, D., E. Wiarda, and P. Vermeer, 2013, Surface-and guided-wave inversion for near-surface modeling in land and shallow marine seismic data: *The Leading Edge*.
- Brooks, L. A. and P. Gerstoft, 2009, Green’s function approximation from cross-correlations of 20–100 Hz noise during a tropical storm: *The Journal of the Acoustical Society of America*, **125**, 723–734.
- Burg, K., M. Ewing, F. Press, and E. Stulken, 1951, A seismic wave guide phenomenon: *Geophysics*, **16**, 594–612.
- Chang, J. P., 2017, The search for P-waves at Forties: SEP-Report, **168**, 35–50.
- Chang, J. P., S. A. de Ridder, and B. L. Biondi, 2016, High-frequency Rayleigh-wave tomography using traffic noise from Long Beach, California: *Geophysics*, **81**, B1–B11.
- de Ridder, S., 2014, Analysis of moere vest obsns as continuous data: SEP-Report, **155**, 71–80.
- de Ridder, S., B. Biondi, and D. Nichols, 2015, Elliptical-anisotropic eikonal phase velocity tomography: *Geophysical Research Letters*, **42**, 758–764.
- de Ridder, S. and J. Dellinger, 2011, Ambient seismic noise eikonal tomography for near-surface imaging at Valhall: *The Leading Edge*, **30**, 506–512.
- de Ridder, S. A. and B. L. Biondi, 2015, Ambient seismic noise tomography at ekofisk: *Geophysics*.
- Draganov, D., X. Campman, J. Thorbecke, A. Verdel, and K. Wapenaar, 2013, Seismic exploration-scale velocities and structure from ambient seismic noise (> 1 Hz): *Journal of Geophysical Research: Solid Earth*, **118**, 4345–4360.
- Fabien-Ouellet, G., E. Gloaguen, and B. Giroux, 2017, Time-domain seismic modeling in viscoelastic media for full waveform inversion on heterogeneous computing platforms with OpenCL: *Computers & Geosciences*, **100**, 142–155.
- Gerstoft, P., P. M. Shearer, N. Harmon, and J. Zhang, 2008, Global P, PP, and PKP wave microseisms observed from distant storms: *Geophysical Research Letters*, **35**.
- Hatchell, P. and K. Mehta, 2010, Ocean bottom seismic (obs) timing drift correction using passive seismic data, *in* SEG Technical Program Expanded Abstracts 2010, 2054–2058, Society of Exploration Geophysicists.
- Klein, G., T. Bohlen, F. Theilen, S. Kugler, and T. Forbriger, 2005, Acquisition and inversion of dispersive seismic waves in shallow marine environments: *Marine Geophysical Researches*, **26**, 287–315.
- Lin, F.-C., D. Li, R. W. Clayton, and D. Hollis, 2013, High-resolution 3D shallow crustal structure in Long Beach, California: Application of ambient noise tomography on a dense

- seismic array: *Geophysics*, **78**, Q45–Q56.
- Mordret, A., M. Landès, N. Shapiro, S. Singh, P. Roux, and O. Barkved, 2013, Near-surface study at the valhall oil field from ambient noise surface wave tomography: *Geophysical Journal International*, ggt061.
- Moronfoyer, A. T., S. Ronen, S. L. Klemperer, and G. Alves, 2016, Upper-crustal structure of the Møre margin, offshore-norway: 2016 AGU Fall Meeting, San Francisco, –.
- Nakata, N., J. P. Chang, J. F. Lawrence, and P. Boué, 2015, Body wave extraction and tomography at long beach, california, with ambient-noise interferometry: *Journal of Geophysical Research: Solid Earth*, **120**, 1159–1173.
- Nakata, N., R. Snieder, T. Tsuji, K. Larner, and T. Matsuoka, 2011, Shear wave imaging from traffic noise using seismic interferometry by cross-coherence: *Geophysics*, **76**, SA97–SA106.
- Poli, P., M. Campillo, H. Pedersen, et al., 2012, Body-wave imaging of Earths mantle discontinuities from ambient seismic noise: *Science*, **338**, 1063–1065.
- Roux, P., K. G. Sabra, P. Gerstoft, W. Kuperman, and M. C. Fehler, 2005, P-waves from cross-correlation of seismic noise: *Geophysical Research Letters*, **32**.
- Ruigrok, E., X. Campman, and K. Wapenaar, 2011, Extraction of P-wave reflections from microseisms: *Comptes Rendus Geoscience*, **343**, 512–525.
- Seats, K. J., J. F. Lawrence, and G. A. Prieto, 2012, Improved ambient noise correlation functions using Welch’s method: *Geophysical Journal International*, **188**, 513–523.
- Shapiro, N. M., M. Campillo, L. Stehly, and M. H. Ritzwoller, 2005, High-resolution surface-wave tomography from ambient seismic noise: *Science*, **307**, 1615–1618.
- Shtivelman, V., 2004, Estimating seismic velocities below the sea-bed using surface waves: *Near Surface Geophysics*, **2**, 241–247.
- Wang, J., R. R. Stewart, N. I. Dyaur, and M. L. Bell, 2016, Marine guided waves: Subbottom property estimation and filtering using physical modeling data: *Geophysics*.
- Wapenaar, K., D. Draganov, R. Snieder, X. Campman, and A. Verdel, 2010, Tutorial on seismic interferometry: Part 1–basic principles and applications: *Geophysics*, **75**, 75A195–75A209.
- Zhan, Z., S. Ni, D. V. Helmberger, and R. W. Clayton, 2010, Retrieval of Moho-reflected shear wave arrivals from ambient seismic noise: *Geophysical Journal International*, **182**, 408–420.

Time-lapse changes in ambient noise interferometry and dispersion analysis at the Stanford DAS Array

Eileen Martin and Biondo Biondi

ABSTRACT

Ambient noise interferometry allows us to extract signals that mimic active source surveys without the cost and permitting requirements of a true active survey for near-surface imaging. In many environments, seismic velocities in the near surface may change seasonally, reacting to temperature and saturation, and even subsidence. We analyze time-lapse changes in virtual source response estimates extracted from ambient seismic noise recorded at the Stanford Distributed Acoustic Sensing Array (SDASA-1) between September 2016 and August 2017. Our analysis indicates that only one week of noise is enough to yield stable virtual source response estimates when compared with the estimate from the same full month of noise. The virtual source response estimates we extract throughout one year appear to show an improvement in signal-to-noise-ratio during months when the ground is more saturated. The Rayleigh wave dispersion images show velocities in the same range as active source geotechnical surveys (Thomas et al., 2013). Further, their Rayleigh wave dispersion images suggest changes in near surface velocity tied to those saturation changes. But these apparent velocity changes are also accompanied by power spectrum changes, so further investigation into the ambient noise field is needed before these velocity shifts can be interpreted with certainty.

INTRODUCTION

For the purpose of near-surface characterization, we are interested in processing ambient noise to avoid the cost, time, and permitting requirements involved in active surveys. Ambient noise interferometry has successfully been used with point sensors to create data mimicking active surveys at the scale of a city (Chang et al., 2016) and time-lapse surveys at the reservoir scale (de Ridder, 2014). In urban areas, the long-term use of traditional sensors, nodes or otherwise is logistically and economically prohibitive for many applications: permission and permits for installation must be obtained separately for each location, wired sensors can only span a short distance, node batteries must be replaced regularly (often monthly), few cost-effective sensors have wireless data communication capabilities, and failed or stolen sensors must be replaced.

Distributed acoustic sensing (DAS) installed in existing telecommunications conduits, similar to the Stanford Distributed Acoustic Sensing Array (SDASA-1), may provide a cost-effective alternative on the receiver side: the cost per sensor is below \$1/meter, higher density can be achieved through changes in the interrogator unit, a single power source and communication point is required for the entire system of potentially thousands of sensors, and sensors can be left in place indefinitely in a conduit as secure as existing utilities with

little to no maintenance cost. However, DAS systems measure strain rates, a tensor quantity, distributed over subsets of fiber. This leads to some potentially challenging theoretical issues that we must account for in processing (Martin et al., 2017).

Passive Rayleigh wave interferometry has been applied recently to trenched-fiber DAS data for imaging at geotechnical scales (Zeng et al., 2017; Martin et al., 2016), and it appears to yield the proper velocities, but this yields no information between lines in the array. In hopes of increasing our ray coverage throughout the space between fiber lines, we have extended the calculation of virtual source response estimates to pairs of fiber channels throughout 2D arrays, and have specifically applied this theory at SDASA-1 (Martin and Biondi, 2017). In this report, we test whether the virtual source response estimates throughout the 2D array show significant changes throughout the first year of recording.

First, we establish that stacking roughly one week of cross-correlations is enough to ensure that virtual source response estimates throughout the array are stable relative to the signal extracted from one month of noise. Next, we look at one year of minimally pre-processed cross-correlations responding to two virtual source locations (one east-west oriented and one north-south oriented) on SDASA-1. Virtual source response estimates from epochs when the ground is more saturated show a clear increase in signal-to-noise ratio at longer distances. We calculate dispersion images for the Rayleigh wave portion of the virtual source response estimates (channels colinear to the virtual source channel), showing how much energy travels at each velocity for each frequency, and clarifying the apparent velocity changes throughout this one year period. Still, the changes in apparent velocities are accompanied by changes in the power spectrum of the background noise field, so further analysis of the ambient noise field is needed before the picked velocities can be interpreted.

CONVERGENCE ANALYSIS

To test how long we need to average cross-correlations to get stable virtual source response estimates, we study the similarity of short-term averages from varying window lengths to month-long averages throughout September 2016. Note that convergence of virtual source response estimates does not necessarily mean a more accurate signal, just a more repeatable one. As in (Seats et al., 2012), for any virtual source v receiver r pair, we use the normalized zero time-lag correlation between the long term average virtual source response estimate, $l(v, r, \tau)$, where τ is time lag, and any shorter virtual source response estimate, $s(v, r, \tau; t, t+w)$ averaging cross-correlations for windows that start and end between time t and $t+w$. Thus, if the virtual source response estimates are compactly supported on $(-\tau, \tau)$, their correlation coefficient is:

$$R_C(v, r; t, t+w) = \frac{\int_{-\tau}^{\tau} l(v, r, \tau') \cdot s(v, r, \tau'; t, t+w) d\tau'}{\left(\int_{-\tau}^{\tau} l^2(v, r, \tau') d\tau'\right)^{1/2} \left(\int_{-\tau}^{\tau} s^2(v, r, \tau; t, t+w) d\tau'\right)^{1/2}} \quad (1)$$

Note that in practice R_C is an approximation with sums over discretized signal vectors instead of integrals of continuous functions. If $R_C(v, r; t, t+w) \approx 1$ for nearly all window start times, t , this suggests little is gained by continuing to average (v, r) cross-correlations averaged over a longer window, w .

As seen in Figures 1(a) to 1(g), we calculated R_C to see how it varied throughout our 30 days of cross-correlations. We calculated s for all receivers responding to channel 75 as a virtual source for each contiguous subset of w hours throughout the 30 days. This was tested for $w = 6, 12, 24, 48, 96, 192,$ and 384 hour sliding windows. Note that windows with $w > 1$ overlap, so when $w = 6$, there is a window from midnight to 6 am on the first day, a window from 1 am to 7 am on the first day, and so on. The horizontal stripes in R_C plots for 6 and 12 hour windows repeatedly show lower R_C values during the daytime, even in parts of the array farther from roads. These daily variations are barely picked up by the auto-correlation of channel 75, indicating that processing decisions based on convergence require a measure of convergence throughout the array, not just auto-correlations. Overall improvement could be quantified by an R_C matrix with higher entropy (more even convergence throughout the array and over time), higher average values, or higher minimum values. One possibility is not using any daytime noise, but this increases the recording time required. By the time one week of data has been integrated, we have stable cross-correlations.

CALCULATING TIME-LAPSE CHANGES AND RESULTS

Previously, we showed that coherence virtual source response estimates could be extracted throughout the array out of one week of data (Martin and Biondi, 2017). Using similar processing, we extended those analyses to calculate a monthly correlation using six days of data from each month from September 2016 to April 2017, including all 24 hours of the day. Even when averaging over multiple days of data, ambient noise pre-processing decisions can cause significant biases in the estimated Green's functions retrieved by (Fichtner, 2014). Thus, we did minimal pre-processing: data were divided into 5 minute windows with 50% overlap, were bandpassed from 0.5 -24 Hz, were thresholded to +/- 1, cross-correlated, then stacked for each hour. After saving each hour's average cross-correlations throughout the week, we normalized the cross-correlations by their L2 norms. Finally these virtual source response estimates were stacked over six days, yielding a virtual source response estimate.

Cross-correlation changes

Our convergence analysis indicates that just four days of data are be enough to yield stable virtual source response estimates when compared to estimates from the same full month of data. Still, to be on the safe side with respect to convergence, we show cross correlations for one week of data from each month in Figures 2(a) to 3(l). Although these correlations may be relatively constant in the one month time scale, changes do emerge over a longer time scale, indicating either significant change in the noise field or velocity profile. Data collected in November through April yielded higher signal to noise ratios at longer distances than data in June, July, August, September and October. This can be seen comparing Figures 2(h) and 3(h) to Figures 2(a) and 3(a), which are the virtual source response estimates to the same virtual source channels shown in September in Martin and Biondi (2017). We hypothesize that as the ground became more saturated with rain starting in the late fall, the coupling between the conduits and soil improved, which meant an improvement in sensitivity on both the virtual source and the receiver side. We are continuing to investigate time-lapse changes in this challenging environment with many transient noise sources. Even for static virtual source response estimates, the issues with pre-processing related biases are especially

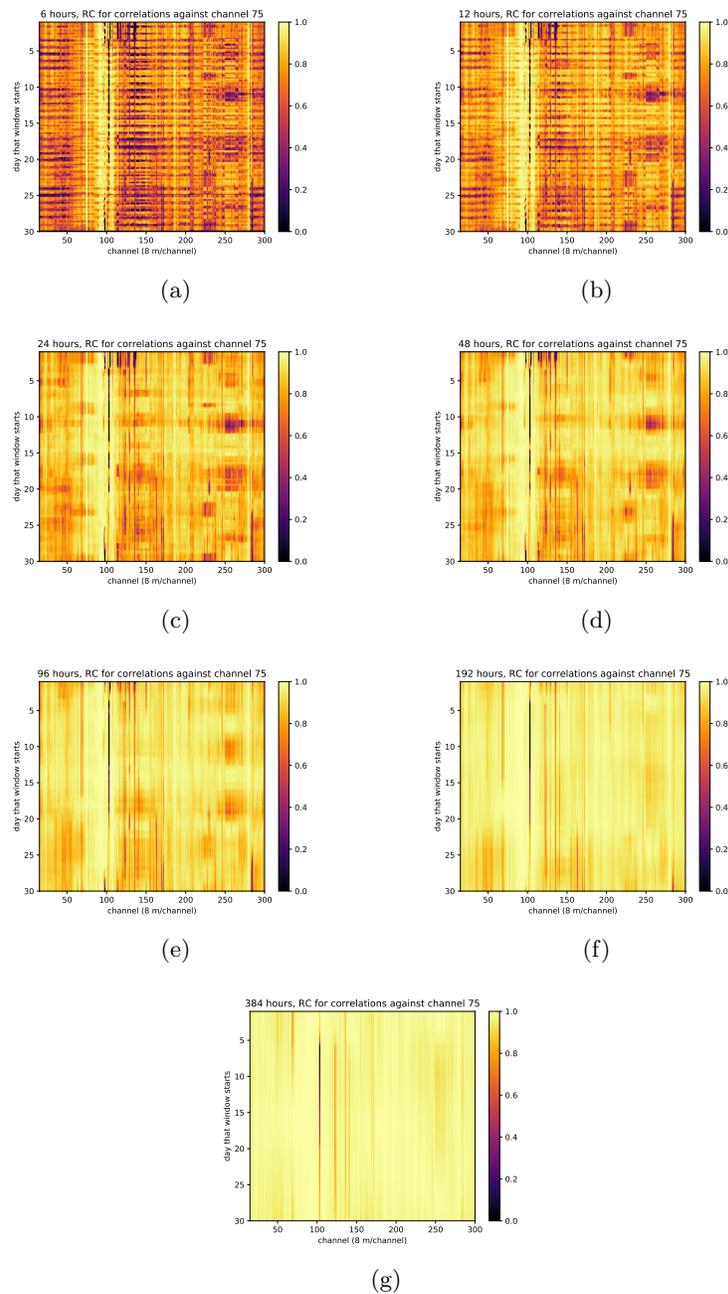


Figure 1: Calculations of R_c for cross-correlations between channel 75 and the rest of the array throughout September 2017 for (a) 6 hour windows, (b) 12 hour windows, (c) 24 hour windows, (d) 48 hour windows, (e) 96 hour windows, (f) 192 hour windows, (g) 384 hour windows show convergence on nearly all receiver channels after one week (indicated by R_c near 1). [CR] eileen1/. RC6,RC12,RC24,RC48,RC96,RC192,RC384

pertinent in environments with transient noise sources, so we are working towards scalable methods for automatic identification and filtering of these sources in urban environments (Huot et al., 2017).

Dispersion image changes

We calculate Rayleigh wave dispersion images from virtual source response estimates limited to channels along the same line as these, as seen in Figures 4(a) to 5(l). These dispersion images were calculated via tau-p transforms followed by a Fourier transform along the tau axis. These dispersion images tell us how much energy is traveling at each velocity for a given frequency, summarizing the Rayleigh wave speeds in each of the cross-correlation plots. Energy in the 10-20 Hz range tends to travel at faster velocities in the summer months in response to channel 75, and this trend sort of continues in response to channel 35 (but some months this energy is not well-focused enough to differentiate). At lower frequencies, 3-10 Hz, energy travels faster in the wetter months in response to channel 35, but there is no clear trend in response to channel 75. Below 3 Hz and above 20 Hz there is no focused peak in these dispersion images.

Due to its apparently higher signal-to-noise ratio, we focus on results from April, seen for the whole month in Figure 6(a) to 6(d). To better distinguish signal from noise, we do not normalize each frequency so that the value at the peak velocity in the dispersion image is 1. These show velocities and frequencies in the correct range for geotechnical surveys, suggesting this may be a reasonable tool for earthquake hazard analysis. At 5 Hz the main velocity near channel 61 is 700 m/s, and such a wave would be sensitive to features in the top 45 to 70 meters. Over at channel 141, the peak velocity for 5 Hz waves is closer to 400 m/s, which would be sensitive to features at a scale of 25 to 40 meters. At both locations, the peak at 5 Hz is continuous over a range of frequencies and is likely the fundamental mode. At 10 Hz, the main velocity near channel 61 is around 400 m/s, which would be sensitive to features at a scale of roughly 13 to 20 meters. Farther north, at channel 141, the peak velocity for 10 Hz is closer to 300 m/s, which would be sensitive to features in the top 10 to 15 meters. For earthquake hazard analysis, engineers must estimate surface wave and S-wave velocity in the top 30 meters of the subsurface. Thus, the frequencies at which we extract signals are in the right range for geotechnical studies, and the ability to cut the cost of these surveys could enable more widespread near surface stability studies.

DISCUSSION

The findings here indicate the likely potential of DAS interferometry to detect near-surface changes, but further work must be done to ensure that spatially and temporally heterogeneous noise sources are not the cause of these changes. In this report we used very minimal preprocessing, but new tools developed within SEP need to be incorporated into ambient noise analysis to remove nearby vehicles (Huot et al., 2017) and earthquake recordings (Yuan et al., 2017). Furthermore, it has been documented that even power spectrum changes can cause false apparent velocity changes, so we need to test the robustness of our velocity changes to spectral whitening and cross-coherence (Daskalakis et al., 2016). Further, we intend to build on these results by performing surface wave inversion.

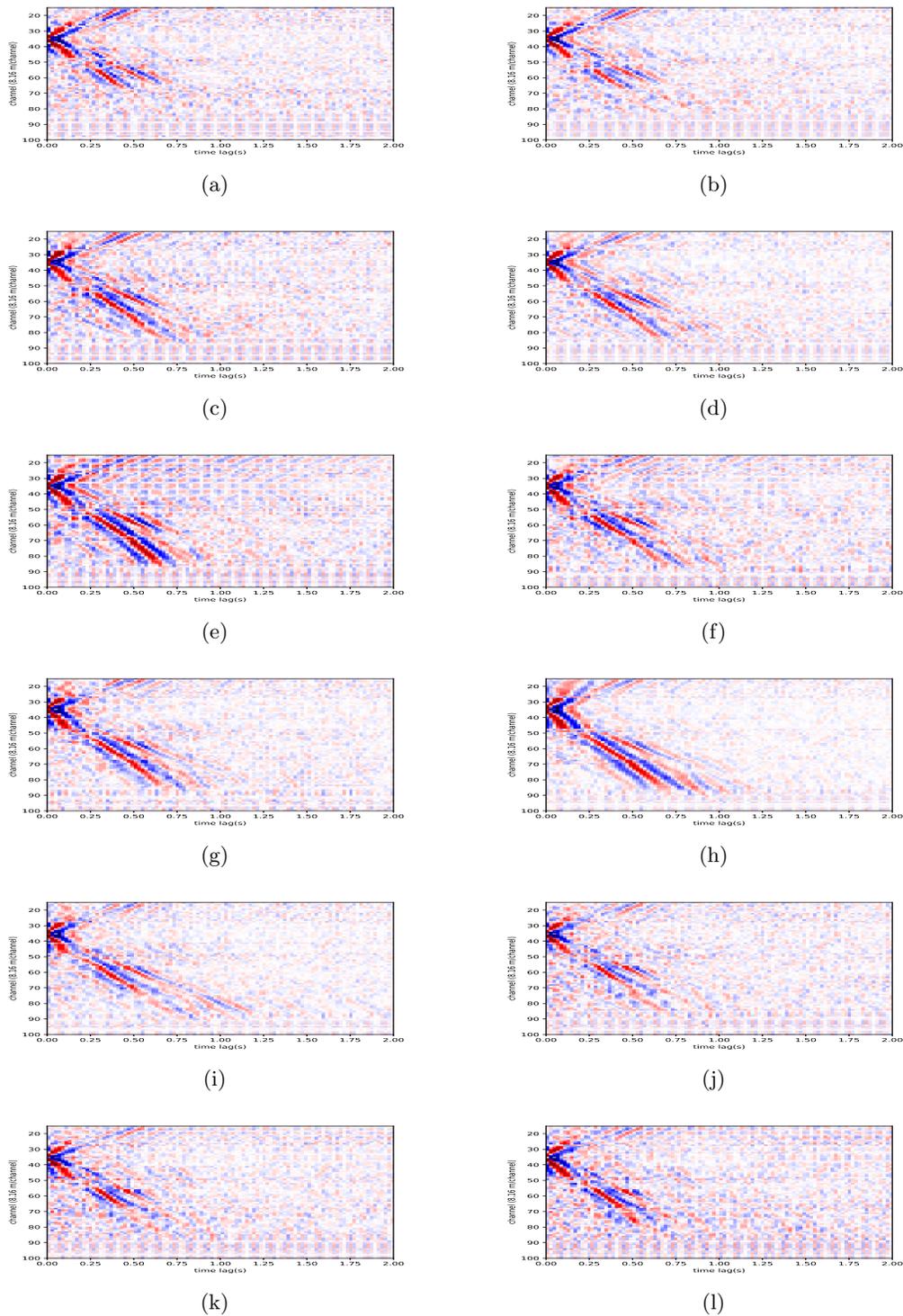


Figure 2: Cross-correlations around one corner of the array responding to a virtual source at channel 35 show an increase in signal to noise ratio at longer distances from September 2016 (top left) and October 2016 (top right) through July 2017 (bottom left) and August 2017 (bottom right). Each month's cross-correlations were calculated independently (not stacked as more months were recorded). [CR]

eileen1/. x1a,x2a,x3a,x4a,x5a,x6a,x7a,x8a,x9a,x10a,x11a,x12a

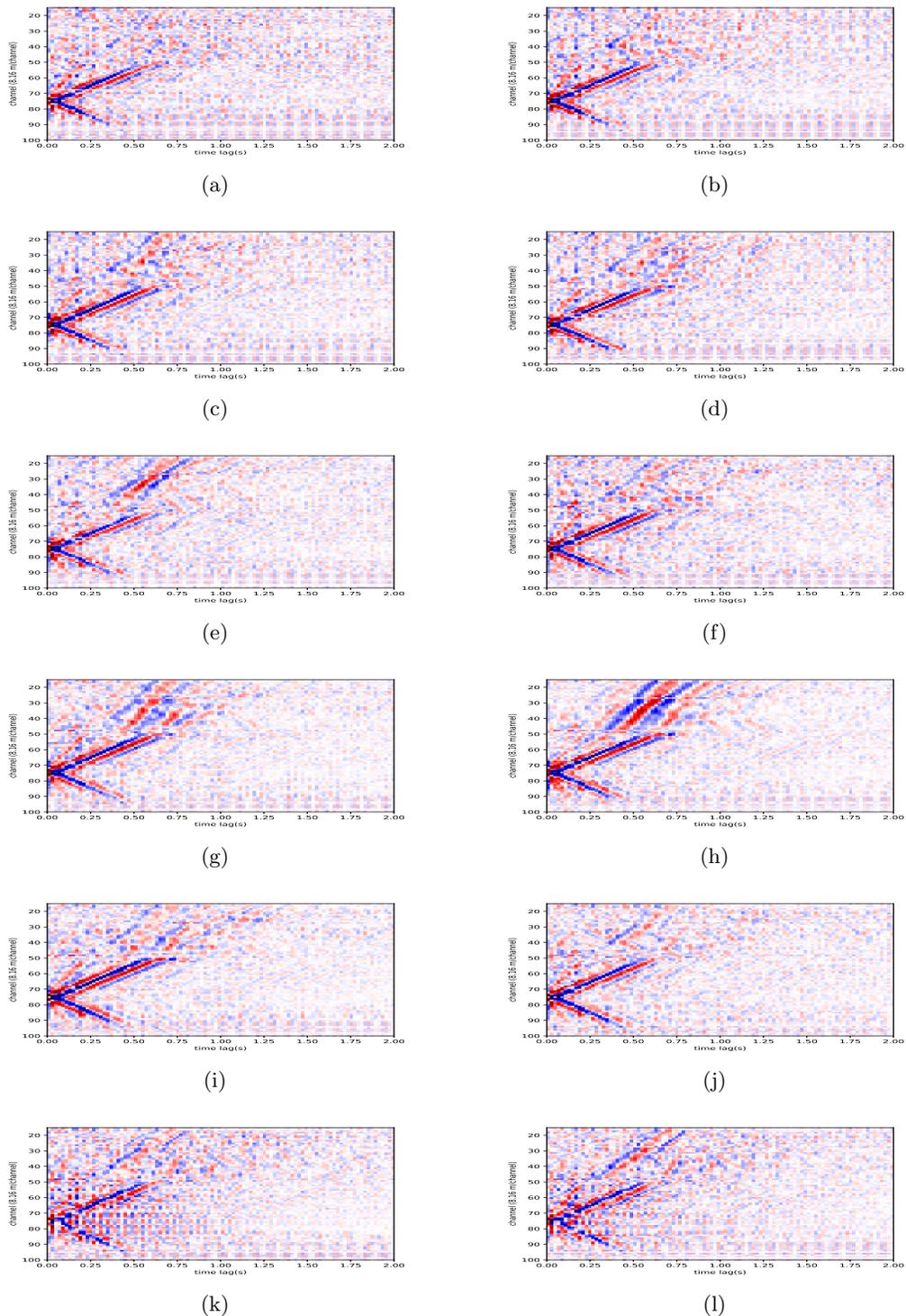


Figure 3: Cross-correlations around one corner of the array responding to a virtual source at channel 75 show an increase in signal to noise ratio at longer distances from September 2016 (top left) and October 2016 (top right) through July 2017 (bottom left) and August 2017 (bottom right). Each month's cross-correlations were calculated independently (not stacked as more months were recorded). [CR]

eileen1/. x1b,x2b,x3b,x4b,x5b,x6b,x7b,x8b,x9b,x10b,x11b,x12b

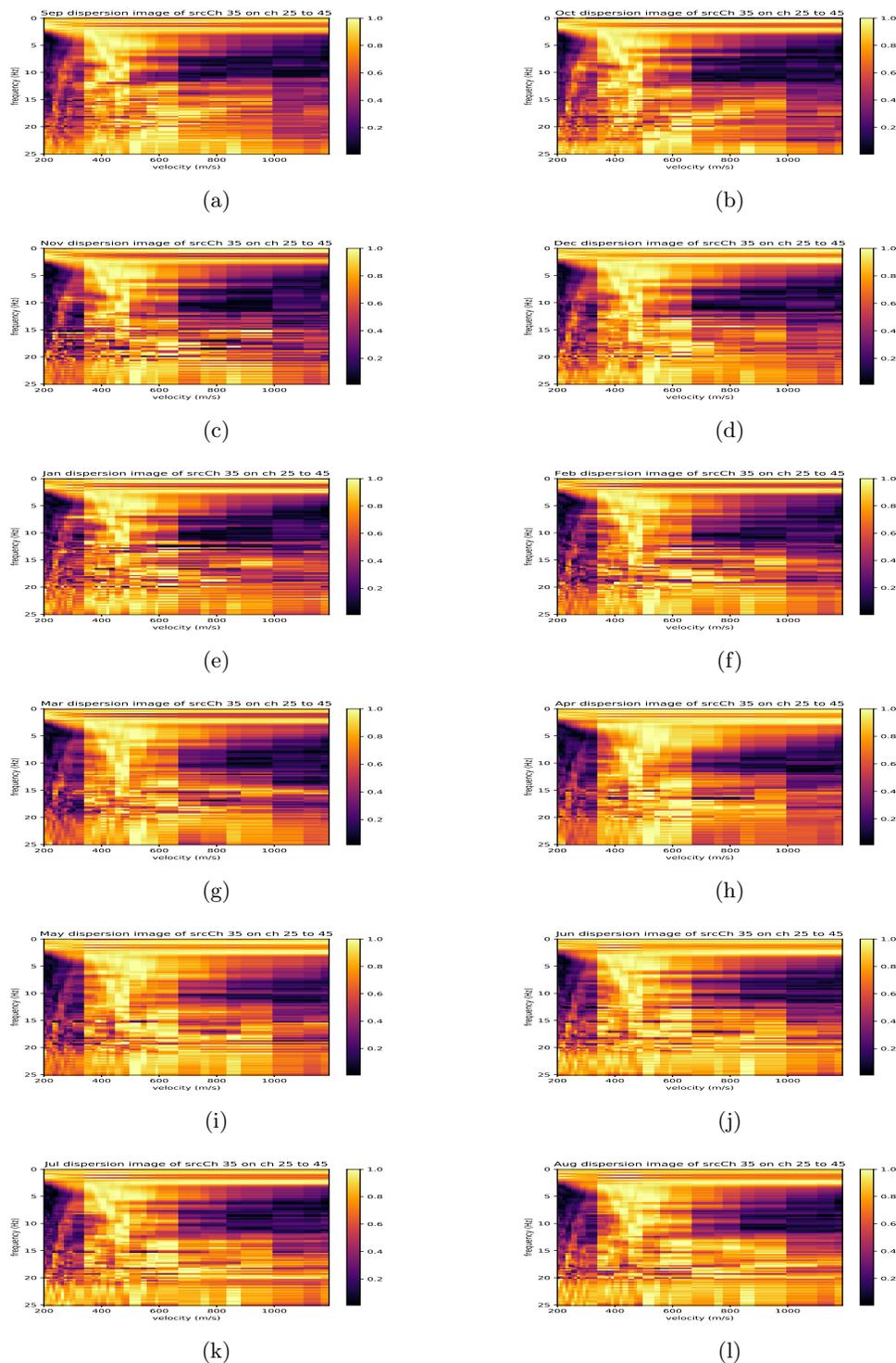


Figure 4: We calculated dispersion images from the Rayleigh-wave parts of the cross-correlations in Figures 2(a) to 2(l) between September 2016 and August 2017. The Rayleigh waves are expected to primarily be extracted from channels 15 to 50. The peak velocity for each frequency is normalized to 1 (so energy being spread in the 20-25 Hz range actually indicates that we do not extract much coherent energy in that range). [CR]

eileen1/. d1a,d2a,d3a,d4a,d5a,d6a,d7a,d8a,d9a,d10a,d11a,d12a

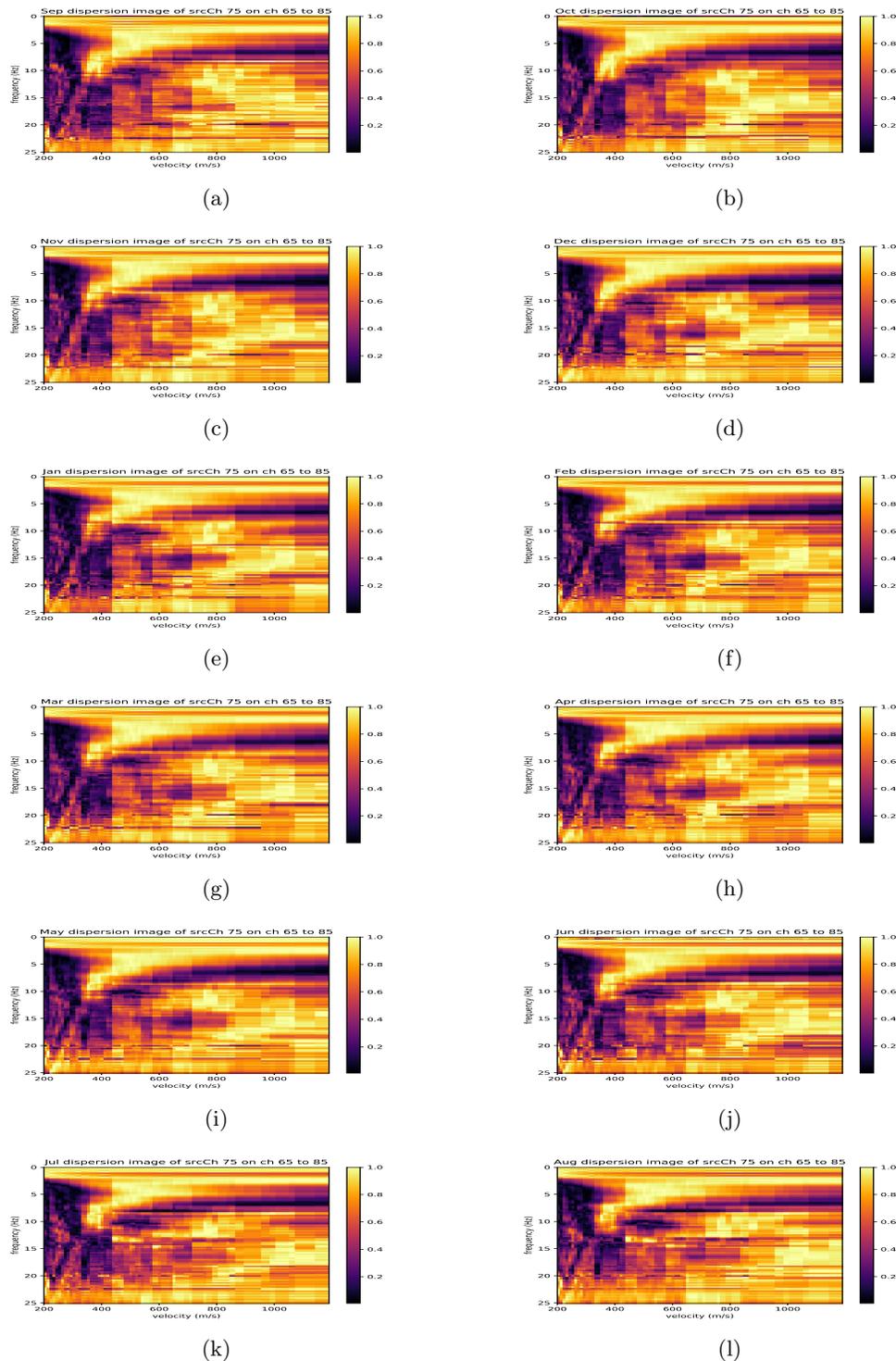


Figure 5: We calculated dispersion images from the Rayleigh-wave parts of the cross-correlations in Figures 3(a) to 3(l). The Rayleigh waves are expected to primarily be extracted from channels 50 to 100. The peak velocity for each frequency is normalized to 1 (so energy being spread in the 20-25 Hz range actually indicates that we do not extract much coherent energy in that range). [CR]

eileen1/. d1b,d2b,d3b,d4b,d5b,d6b,d7b,d8b,d9b,d10b,d11b,d12b

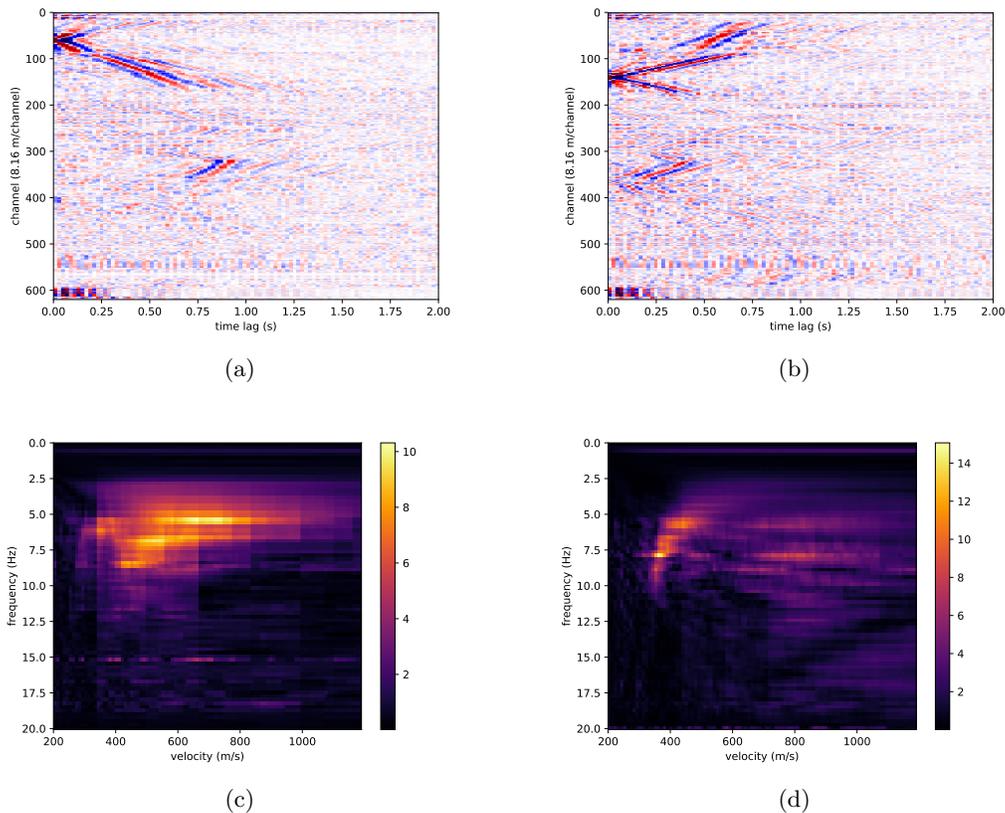


Figure 6: Throughout April we calculated virtual source response estimates throughout both loops of the entire array, and the dispersion images just for fiber in-line with the virtual source channel’s orientation. In addition to the in-line channels yielding Rayleigh waves, we can also see channels parallel to the virtual sources yielding something made primarily of Love waves. Responding to virtual source channel 25 (left, marked as 69 when both loops are there), we see a reaction between 0.6 and 1.0 seconds on parallel channels 300 to 370. Responding to a virtual source at channel 75 (right, marked as 141 when both loops are there) we see a reaction between channels 320 and 400 between 0.2 and 0.5 seconds which is likely to be primarily converted wave energy due to the orthogonal orientation between the virtual source and receivers. [CR]

eileen1/. xcorrAllApr35,xcorrAllApr75,dispImgAllApr35,dispImgAllApr75

While the methods used have their limitations, these findings establish that changes in the virtual source response estimates in an urban area throughout one year can be significant, whether by changes in the background noise field or by changes in the velocity profile due to saturation or other effects. We believe this is the first year-long study of ambient noise interferometry from a dense array in an urban area, and it was made possible primarily by our novel acquisition method: DAS in existing telecom conduits. This method of deploying many dense sensors is much easier to maintain in an urban area over long periods of time than geode or node systems. As we further probe the cause of these changes in interferometry, we will no doubt learn more about trends in anthropogenic noise on campus, and potentially even our urban hydrology system.

ACKNOWLEDGEMENTS

We thank sponsors of the Stanford Exploration Project for funding this work. E. Martin has also been supported in part by the DOE CSGF under grant number DE-FG02-97ER25308, and a Schlumberger Innovation Fellowship. We would like to thank OptaSense, particularly M. Karrenbach and S. Cole for the use of their ODH-3 interrogator unit and for many helpful suggestions. We would also like to thank Jonathan Ajo-Franklin (Lawrence Berkeley National Lab), Nate Lindsey (University of California Berkeley), George Papanicolaou, Jason Chang and Bob Clapp for their insights and assistance.

REFERENCES

- Chang, J., S. de Ridder, and B. Biondi, 2016, High-frequency rayleigh-wave tomography using traffic noise from long beach, california: *Geophysics*, **81**, 843–854.
- Daskalakis, E., C. Evangelidis, J. Garnier, N. Melis, G. Papanicolaou, and C. Tsoga, 2016: *Geophysics Journal International*, **205**, 1926–1936.
- de Ridder, S., 2014, Passive seismic surface-wave interferometry for reservoir-scale imaging: PhD thesis, Stanford University.
- Fichtner, A., 2014, Source and processing effects on noise correlations: *Geophysics Journal International*, **197**, 1527–1531.
- Huot, F., Y. Ma, R. Cieplicki, E. Martin, and B. Biondi, 2017, Automatic noise exploration in urban areas: Expanded Abstracts of the 87th Ann. Internat. Mtg.
- Martin, E. and B. Biondi, 2017, Ambient noise interferometry across two-dimensional das arrays: Expanded Abstracts of the 87th Ann. Internat. Mtg.
- Martin, E., G. Fabien-Oullet, R. Clapp, and B. Biondi, 2017, Sensitivity analysis of distributed acoustic sensing arrays: SEP Report, **170**.
- Martin, E., N. Lindsey, S. Dou, J. Ajo-Franklin, T. Daley, B. Freifeld, M. Robertson, C. Ulrich, A. Wagner, and K. Bjella, 2016, Interferometry of a roadside das array in fairbanks, ak: Expanded Abstracts of the 86th Ann. Internat. Mtg., 2725–2629.
- Seats, K., J. Lawrence, and G. Prieto, 2012, Improved ambient noise correlation functions using welch’s method: *Geophys. Journal Internat.*, **188**, 512–523.
- Thomas, P., I. Wong, J. Zachariasen, R. Darragh, and W. Silva, 2013, 2013 update to the site-specific seismic hazard analyses and development of seismic design ground motions: Stanford university, california: URS Corporation and Pacific Engineering & Analysis.
- Yuan, S., E. Martin, J. Chang, S. Cole, and B. Biondi, 2017, Catalog of northern california earthquakes recorded by das: SEP Report, **170**.

Zeng, X., C. Thurber, H. Wang, D. Fratta, E. Matzel, and P. Team, 2017, High-resolution shallow structure revealed with ambient noise tomography on a dense array: Proceedings, 42nd Workshop on Geothermal Reservoir Engineering, SGP-TR-212.

Sensitivity analysis of distributed acoustic sensing arrays

Eileen R. Martin, Biondo Biondi, Gabriel Fabien-Ouellet, and Robert G. Clapp

ABSTRACT

Distributed acoustic sensing (DAS) measures the average axial strain (strain rate) along a subset of a fiber optic cable, as opposed to the particle displacement (velocity) at a particular small point sensor. In shifting from measuring a vector field to a tensor field, DAS effectively increases the directional sensitivity of measurements of every type of seismic wave when compared to single-component geophones. This switch from vector to tensor quantities leads to a plausible explanation for sign-flips between orthogonal channels seen during some S-wave and surface-wave events in our recordings of earthquakes. We show this through theoretical analysis of planar Rayleigh, Love, P- and S-waves over both infinitesimally small and realistic gauge lengths. We extend the analysis of individual sensor detection of surface waves to inter-receiver cross-correlations of these detections showing even more directionally-dependent sensitivity trends than individual sensors.

INTRODUCTION

It is well known that DAS channels have more extreme sensitivity patterns to waves incident on the fiber at an angle than single-component geophones (Kuvshinov, 2016), but there are in fact more complex relationships at play, particularly as we pursue arrays with increasingly more unusual geometry. In many of our recordings of earthquakes at the Stanford Distributed Acoustic Sensing Array (SDASA-1), we noticed a recurring pattern, particularly after the arrival of S-waves: fibers that were parallel would extend at the same time fibers in the orthogonal direction would compress. Immediately following the P-wave arrivals, it was less clear whether this behavior was observed. Examples of this behavior can also be seen at the L-shaped DAS array at the Richmond Field Station, and the orthogonal grid lines in Fairbanks, AK. Previously, we hypothesized based on 2D tensor rotation math that this was to be expected for waves which propagate orthogonal to their direction of particle motion (Biondi et al., 2017). In this report, we aim to better understand how angle of incidence, wavenumber, and gauge length modify DAS recordings of plane waves, building on the work of (Kuvshinov, 2016).

Here, we work through the 3D math for sensitivity assuming infinitesimally small gauge length, which predicts the same trends: P-waves and Rayleigh waves recorded on orthogonal channels are equal up to a factor of $\tan^2(\theta)$, where θ is the angle between one fiber and the wave propagation direction, while Love waves are equal up to a factor of -1. S-wave recordings are made up of two terms: one that orthogonal channels record up to a factor of $\tan^2(\theta)$, and the other which is recorded up to a factor of -1. The balance between these two terms is determined by the angle between the polarization direction and the horizontal surface that the two orthogonal fibers lay on, so that if the S-wave's particle motion direction is perfectly horizontal, it would show the same trend as a Love wave.

From the plane wave analysis with infinitesimally small gauge length, we calculate the response of a channel with finite gauge length responding to each type of plane wave coming from any angle to better understand earthquake and active source response. Essentially, the angular sensitivity response begins to develop blind angles (like frequency notches, but in space!), which increase in angular density as the wavenumber increases.

Once we understand the response of a single channel to a plane surface-wave, we analyze the response of a pair of sensors to that same plane wave. Naturally, the directional sensitivity is made even more extreme by combining both receiver sensitivity responses. For Rayleigh waves detected by colinear channels, this makes little difference since sources traveling in-line with the sensor pair are preferentially sensed.

But for Love waves detected by parallel non-colinear channels, this often leads to false apparent velocity increases even in the presence of an angularly uniform noise field. Previously we had used a simple model to express interferometry of parallel channels as a linear combination of Love wave Green's function estimates that would each yield the proper velocity (Martin and Biondi, 2017), but we had neglected to consider destructive interference in the signals could lead to false apparent velocities. Understanding these responses is the first step towards correcting for them and extracting true Love wave velocities, as well as learning to use channels at an arbitrary orientation with respect to each other.

PREDICTED EFFECT OF FIBER ORIENTATION ON RECORDED PLANE WAVES

In this section, we assume an infinitesimally small gauge length, and calculate the axial strain along a fiber in the direction $(\cos(\theta), \sin(\theta), 0)$ for any $\theta \in [0, 2\pi)$ as it reacts to multiple types of body and surface waves:

- Rayleigh waves propagating in the direction $(x, 0, 0)$ with particle motion in the x and z directions
- P-waves propagating in the direction $(x, 0, z)$ with particle motion in the same direction
- Love waves propagating in the direction $(x, 0, 0)$ with particle motion in the y direction
- S-waves propagating in the direction $(x, 0, z)$ with particle motion in any direction perpendicular to the direction of propagation (special cases: particle motion parallel to $x - z$ plane or $x - y$ plane)

Note that the orientation of the fiber can change freely enough that we simply change our coordinate system to line up the $x - z$ plane with the wave propagation direction. All results that follow can be easily transferred to strain-rate by taking a time derivative. For each type of wave, we calculate the expected response for:

1. a point-wise axial strain measurement of a horizontal straight fiber
2. the point-wise axial strain measurement of an orthogonal horizontal straight fiber
3. the average axial strain throughout one channel of a horizontal straight fiber

We make the simple assumption that DAS measurement is the average axial strain rate along one gauge length centered at (x, y, z) , called $\bar{\sigma}(x, y, z, t)$, with all points weighted equally. Modifications of how the interrogator unit is implemented may change the data from theoretical ideals predicted here (Bona et al., 2017), but that information is typically not available to the user and may be considered a secondary effect.

As our starting point for understanding strain data, we have some strain tensor field observed in the (x, y, z) coordinate system:

$$\Sigma(x, y, z, t) = \begin{bmatrix} \frac{\partial u_x}{\partial x} & \frac{1}{2} \left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \right) & \frac{1}{2} \left(\frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x} \right) \\ \frac{1}{2} \left(\frac{\partial u_y}{\partial x} + \frac{\partial u_x}{\partial y} \right) & \frac{\partial u_y}{\partial y} & \frac{1}{2} \left(\frac{\partial u_y}{\partial z} + \frac{\partial u_z}{\partial y} \right) \\ \frac{1}{2} \left(\frac{\partial u_z}{\partial x} + \frac{\partial u_x}{\partial z} \right) & \frac{1}{2} \left(\frac{\partial u_z}{\partial y} + \frac{\partial u_y}{\partial z} \right) & \frac{\partial u_z}{\partial z} \end{bmatrix}. \quad (1)$$

However, we are generally most interested in the axial strain as observed by some horizontal fiber at an angle θ from the x direction, so that is represented by the $(0, 0)$ entry of the strain tensor rotated into the coordinate system of the fiber (using shorthand that $\mathcal{S}_\theta = \sin(\theta)$ and $\mathcal{C}_\theta = \cos(\theta)$):

$$\Sigma_\theta(x, y, z, t) = \begin{bmatrix} \mathcal{C}_\theta & \mathcal{S}_\theta & 0 \\ -\mathcal{S}_\theta & \mathcal{C}_\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \Sigma(x, y, z, t) \begin{bmatrix} \mathcal{C}_\theta & -\mathcal{S}_\theta & 0 \\ \mathcal{S}_\theta & \mathcal{C}_\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2)$$

So the $(0, 0)$ entry of this, representing the axial strain at the point (x, y, z) in the θ direction is

$$\sigma(x, y, z, t) = \mathcal{C}_\theta^2 \frac{\partial u_x}{\partial x} + \mathcal{C}_\theta \mathcal{S}_\theta \left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \right) + \mathcal{S}_\theta^2 \frac{\partial u_y}{\partial y} \quad (3)$$

Rayleigh Waves

As in Pujol (2003), we consider a monochromatic Rayleigh plane wave propagating in the x direction at phase velocity c , frequency $\omega = kc$, and wavenumber k in a half space:

$$\mathbf{u}(x, y, z, t) = \left((Ae^{-\gamma_\alpha kz} + iB\gamma_\beta e^{-\gamma_\beta kz})e^{ik(ct-x)}, 0, (-iA\gamma_\alpha e^{-\gamma_\alpha kz} + B\gamma_\beta e^{-\gamma_\beta kz})e^{ik(ct-x)} \right), \quad (4)$$

where $\gamma_\delta = \sqrt{1 - \frac{c^2}{\delta^2}}$ for $\delta = \alpha$ and β . Here, α and β are the irrotational and solenoidal parts of the solution to the elastic wave equation, so $\mathbf{u}_\alpha + \mathbf{u}_\beta = \mathbf{u}$, $\nabla \times \mathbf{u}_\alpha = 0$ and $\nabla \cdot \mathbf{u}_\beta = 0$. Following Equation 3, we can calculate the axial strain at a point (x, y, z) on a horizontal fiber at an angle θ counter-clockwise from the propagation direction to be:

$$\sigma(x, y, z, t) = \cos^2(\theta) \frac{\partial u_x}{\partial x} = -ik \cos^2(\theta) (Ae^{-\gamma_\alpha kz} + iB\gamma_\beta e^{-\gamma_\beta kz}) e^{ik(ct-x)} \quad (5)$$

so we can split this into a geometry-dependent amplitude factor, $-ik \cos^2(\theta) (Ae^{-\gamma_\alpha kz} + iB\gamma_\beta e^{-\gamma_\beta kz}) e^{-ikx}$, multiplied by an oscillatory factor, e^{ikct} . That geometry-dependent amplitude factor is plotted for many angles in Figure 1. In the Stanford DAS Array, and multiple other surface DAS arrays in existence, a common trait is that the lines primarily follow two orthogonal directions, so let's investigate the relationship between the observation at an angle θ and $\theta + \frac{\pi}{2}$. Along this orthogonal line at the same location, we expect to

measure:

$$\sigma^\perp(x, y, z, t) = \cos^2\left(\theta + \frac{\pi}{2}\right) \frac{\partial u_x}{\partial x} = \frac{\sin^2(\theta)}{\cos^2(\theta)} \sigma(x, y, z, t). \quad (6)$$

Thus, this rotation is only expected to lead to a $\tan^2(\theta)$ amplitude change, not a shift in the oscillatory part of the signal.

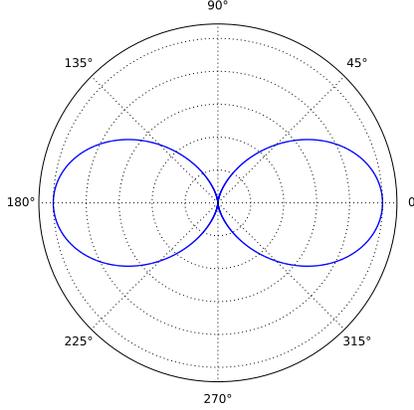


Figure 1: The radius of this polar plot represents the axial strain sensitivity along a horizontal fiber reacting to Rayleigh plane waves propagating at many angles θ . The amplitude of this sensitivity is proportional to wavenumber, k , but its shape is independent of wavenumber. [ER] eileen2/. RSens

More realistically, we expect to measure the average axial strain along a segment of fiber of length g , the gauge length. This is expected to lead to notches in the FK spectrum. Thus, if we had a channel oriented at the angle θ spanning $(x - \frac{g}{2} \cos(\theta), y - \frac{g}{2} \sin(\theta), z)$ to $(x + \frac{g}{2} \cos(\theta), y + \frac{g}{2} \sin(\theta), z)$, it would measure:

$$\begin{aligned} \bar{\sigma}(x, y, z, t) &= \int_{-g/2}^{g/2} \sigma(x + \nu \cos(\theta), y + \nu \sin(\theta), 0) d\nu \\ &= -ik \cos^2(\theta) (A + iB\gamma_\beta) e^{ikct} \int_{-g/2}^{g/2} e^{-ik(x + \nu \cos(\theta))} d\nu \\ &= -ik \cos^2(\theta) (A + iB\gamma_\beta) e^{ikct} e^{-ikx} \left[\frac{e^{-ik\nu \cos(\theta)}}{-ik \cos(\theta)} \right]_{\nu=-g/2}^{g/2} \\ &= \cos(\theta) (A + iB\gamma_\beta) e^{ikct} e^{-ikx} (e^{ik\frac{g}{2} \cos(\theta)} - e^{-ik\frac{g}{2} \cos(\theta)}) \\ &= 2i \cos(\theta) (A + iB\gamma_\beta) e^{ik(ct-x)} \sin\left(\frac{kg}{2} \cos(\theta)\right). \end{aligned} \quad (7)$$

As a sanity check, this makes sense because it says all fiber channels centered on a given x coordinate (and any x coordinates offset by $2\pi/k$) will measure the same response as the plane wave rolls through, propagating in the x direction. The factor $\cos(\theta) \sin(kg \cos(\theta)/2)$ determines the sensitivity of a given channel to plane waves incident on the fiber at an angle θ , so we see that not only orientation, but also the wavenumber-gauge length relationship play an important role in the detected signal. Polar plots of this sensitivity for various kg values are plotted in Figure 2. We see that this bears some similarity to the usual \cos^2

sensitivity, but with a more complex interaction leaving increasingly more frequency notches as k and/or g increase.

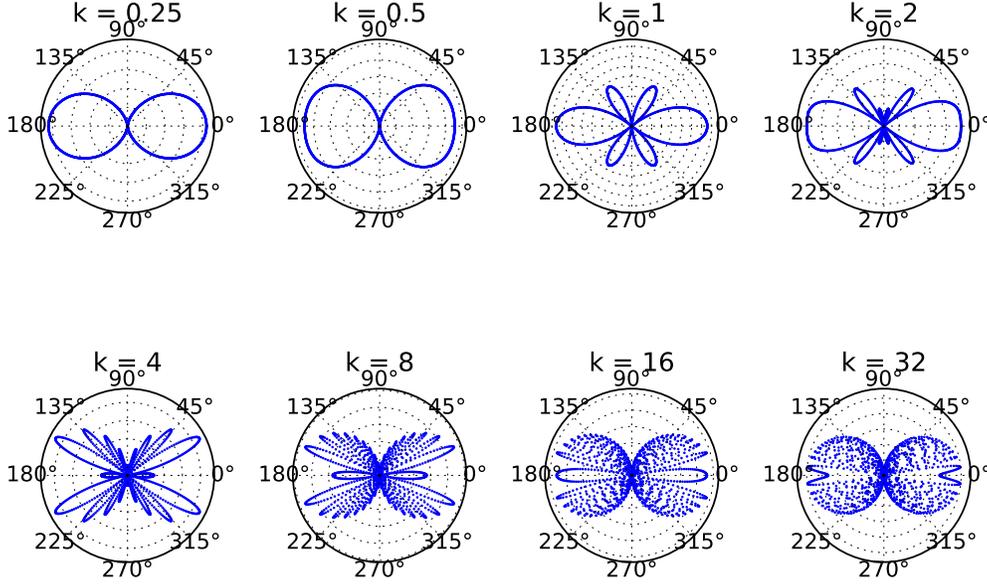


Figure 2: The radius of these polar plots represents the axial strain sensitivity along a horizontal fiber reacting to Rayleigh plane waves propagating at an angle θ relative to the fiber for multiple wavenumbers, k , assuming a gauge length $g = 8$ meters. Each plot is normalized so the peak sensitivity angle appears to be the same radius. [ER] `eileen2/. RChannSens`

As seen in Figure 2, when k is small the sensitivity pattern's shape resembles that of the point-wise measurement. There are two ways to see that in fact the limit as $k \rightarrow 0$ the sensitivity converges to the point limit. We could define the 'shape' of the sensitivity to be the ratio in sensitivity between any two angles, θ_1 and θ_2 , which for point-wise measurements is:

$$R(\theta_1, \theta_2) = \frac{\cos^2(\theta_1) \sin^2(\phi_1)}{\cos^2(\theta_2) \sin^2(\phi_2)} \quad (8)$$

Then the channel-wise measurement ratio, R_c is:

$$R_c(\theta_1, \theta_2, k) = \frac{\cos(\theta_1) \sin(\phi_1) \sin(0.5kg \cos(\theta_1) \sin(\phi_1))}{\cos(\theta_2) \sin(\phi_2) \sin(0.5kg \cos(\theta_2) \sin(\phi_2))} \quad (9)$$

and if we take the limit as $k \rightarrow 0$ using L'Hospital's rule:

$$\begin{aligned} R_c(\theta_1, \theta_2, 0) &= \frac{\cos(\theta_1) \sin(\phi_1) \cos(0.5 * 0 * g \cos(\theta_1) \sin(\phi_1)) 0.5g \cos(\theta_1) \sin(\phi_1)}{\cos(\theta_2) \sin(\phi_2) \cos(0.5 * 0 * g \cos(\theta_2) \sin(\phi_2)) 0.5g \cos(\theta_2) \sin(\phi_2)} \\ &= \frac{\cos^2(\theta_1) \sin^2(\phi_1)}{\cos^2(\theta_2) \sin^2(\phi_2)} \end{aligned} \quad (10)$$

which is the same as the point-wise angular sensitivity ratio. More simply, we can look at the Taylor series expansion about $k = 0$ for the channel sensitivity then drop the small nonlinear higher order terms:

$$\cos(\theta) \sin\left(\frac{kg}{2} \cos(\theta)\right) \approx \cos(\theta) \left[\frac{kg}{2} \cos(\theta) + \frac{(kg)^3}{8} \cos^3(\theta) + \dots \right] \propto kg \cos^2(\theta) \quad (11)$$

so we see it is proportional to the point-wise measurement solution. This observation in fact holds true to the rest of the wave types as well, as verified in the appendix.

P-Waves

Similar to Rayleigh waves, P-waves have particle motion in the same direction as their propagation (but no orthogonal rolling motion). As in Pujol (2003)

$$\mathbf{u}(x, y, z, t) = \left(A \sin(\phi) e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\alpha)}, 0, -A \cos(\phi) e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\alpha)} \right) \quad (12)$$

is the equation for a monochromatic plane P-wave traveling at velocity α at frequency ω , where ϕ is the angle between the z -axis and the direction of propagation. Again, $u_y = 0$, so the axial strain at a given point on a horizontal fiber at an angle θ from the x -axis is:

$$\sigma(x, y, z, t) = -i \cos^2(\theta) \frac{\omega}{\alpha} A \sin^2(\phi) e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\alpha)}. \quad (13)$$

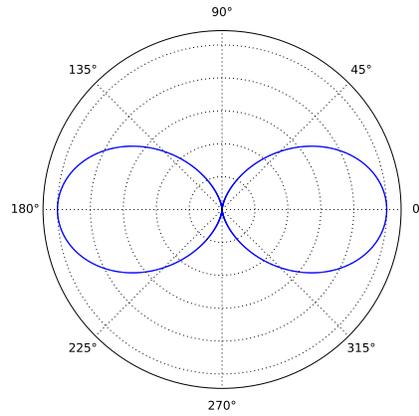
The geometry-dependent amplitude factor depends on the orientation of the fiber with respect to the wave propagation direction, but not on the wavenumber, as seen in Figures 3(a) and 3(b). If we were to also have a colocated orthogonal horizontal fiber, it would simultaneously be sensitive to

$$\sigma^\perp(x, y, z, t) = \cos^2(\theta + \frac{\pi}{2}) \frac{\partial u_x}{\partial x} = \frac{\sin^2(\theta)}{\cos^2(\theta)} \sigma(x, y, z, t), \quad (14)$$

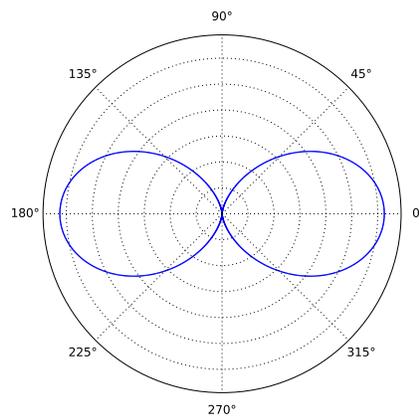
and so like the Rayleigh wave, we expect orthogonal observations of P-waves to exhibit only an amplitude difference proportional to \tan^2 of the angle between the horizontal projection of the propagation direction and one of the fibers, but no phase shift assuming point measurements.

Again, assume we have a channel oriented at the angle θ spanning $(x - \frac{g}{2} \cos(\theta), y - \frac{g}{2} \sin(\theta), z)$ to $(x + \frac{g}{2} \cos(\theta), y + \frac{g}{2} \sin(\theta), z)$, so it would measure:

$$\begin{aligned} \bar{\sigma}(x, y, z, t) &= \int_{-g/2}^{g/2} -i \cos^2(\theta) \frac{\omega}{\alpha} A \sin^2(\phi) e^{i\omega(t - ((x + \nu \cos(\theta)) \sin(\phi) - z \cos(\phi))/\alpha)} d\nu \\ &= -i \cos^2(\theta) \frac{\omega}{\alpha} A \sin^2(\phi) e^{i\omega(t + z \cos(\phi)/\alpha)} e^{-i\omega x \sin(\phi)/\alpha} \int_{-g/2}^{g/2} e^{-i\omega \nu \cos(\theta) \sin(\phi)/\alpha} d\nu \\ &= -i \cos^2(\theta) \frac{\omega}{\alpha} A \sin^2(\phi) e^{i\omega(t + z \cos(\phi)/\alpha)} e^{-i\omega x \sin(\phi)/\alpha} \left[\frac{e^{-i\omega \nu \cos(\theta) \sin(\phi)/\alpha}}{-i\omega \cos(\theta) \sin(\phi)/\alpha} \right]_{-g/2}^{g/2} \\ &= \cos(\theta) A \sin(\phi) e^{i\omega(t + z \cos(\phi)/\alpha)} e^{-i\omega x \sin(\phi)/\alpha} (e^{-i\omega g \cos(\theta) \sin(\phi)/(2\alpha)} - e^{i\omega g \cos(\theta) \sin(\phi)/(2\alpha)}) \\ &= 2i A \cos(\theta) \sin(\phi) e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\alpha)} \sin(\omega g \cos(\theta) \sin(\phi)/(2\alpha)). \end{aligned} \quad (15)$$



(a)



(b)

Figure 3: The radius of these polar plots represents the axial strain sensitivity along a horizontal fiber oriented at that angle θ from the horizontal propagation direction of a P-wave which is also propagating at an angle ϕ from vertical. The ϕ values plotted here are (top) $\pi/2$ and (bottom) $\pi/4$. The shape of these sensitivities (the ratio between the radii of any two angles) is constant with respect to both parameters. [ER]

eileen2/. PSensPiHalf,PSensPiQuart

This easily splits into a time-dependent oscillatory part $e^{i\omega t}$, a geometry-dependent spatial oscillatory part (dependent on apparent wavenumber where the true wavenumber $k = \omega/\alpha$): $e^{-ik(x \sin(\phi) - z \cos(\phi))}$, and a geometry-dependent amplitude factor proportional to $\cos(\theta) \sin(\phi) \sin(\omega g \cos(\theta) \sin(\phi)/(2\alpha))$. We can simplify the amplitude factor a little to be $\cos(\theta) \sin(\phi) \sin(kg \cos(\theta) \sin(\phi)/2)$ where $k = \omega/\alpha$, so like in the Rayleigh wave observations, our amplitude depends on the product of the wavenumber and gauge length as well as the orientation between the fiber and the propagation direction of the wave.

A quick check that this is reasonable: if $\phi = 0$ the wave is vertically propagating, and a horizontal fiber should be insensitive to it, which we verify because the $\sin(\phi)$ factor in the amplitude is 0. Assuming $\phi = \pi/2$ (the special case of horizontally propagating P-waves), plots of the amplitude factor for various kg values are shown in Figures 4(a) and 4(b). Again, we mostly see a pattern that is like the frequently quoted \cos^2 sensitivity trend, but as the wavenumber and/or gauge length increase, the fiber has increasingly more blind angles. Of course, real waves are composed of a range of frequencies, so even if part of the array misses a source at one frequency, it may still detect that source at other frequencies with the same gauge length.

Love Waves

Say that we have a half-space with a layer in the top H meters with group velocity β_1 and β_2 for the half-space underneath. In this scenario, a Love wave can propagate horizontally at phase velocity c in the x direction following the displacement equation:

$$\mathbf{u}(x, y, z, t) = \left(0, (Ae^{-i\eta_1 kz} + Be^{i\eta_1 kz})e^{ik(ct-x)}, 0\right) \quad (16)$$

in the top layer ($z < H$) and

$$\mathbf{u}(x, y, z, t) = \left(0, Ce^{-\eta_2 kz + ik(ct-x)}, 0\right) \quad (17)$$

in the bottom half. Note that $\eta_i = \sqrt{(c/\beta_i)^2 - 1}$. Here, we assume for simplicity that the horizontal DAS array is contained within the top layer. Following a similar procedure with the equation for the lower half could reveal Love wave response for a deeper horizontal buried array (perhaps a deviated well with fiber, or fiber installed in slim holes beneath the complexities of the near surface layer).

Following the same steps as above, we are interested in the strain rate of a fiber at a point oriented at an angle θ from the x -axis. In this case:

$$\begin{aligned} \sigma(x, y, z, t) &= \cos(\theta) \sin(\theta) \frac{\partial u_y}{\partial x} \\ &= -ik \cos(\theta) \sin(\theta) (Ae^{-i\eta_1 kz} + Be^{i\eta_1 kz}) e^{ik(ct-x)}. \end{aligned} \quad (18)$$

We see the angular-dependent amplitude factor plotted in Figure 5, showing a four-lobe structure. This behaves quite differently than the two-lobe sensitivity of Rayleigh and P-waves.

As before, we might wonder how Love waves appear on corners of arrays made of orthogonal lines. At the same point, an orthogonal fiber's axial strain would be:

$$\sigma^\perp(x, y, z, t) = \cos\left(\theta + \frac{\pi}{2}\right) \sin\left(\theta + \frac{\pi}{2}\right) \frac{\partial u_y}{\partial x} = -\sin(\theta) \cos(\theta) \frac{\partial u_y}{\partial x} = -\sigma(x, y, z, t), \quad (19)$$

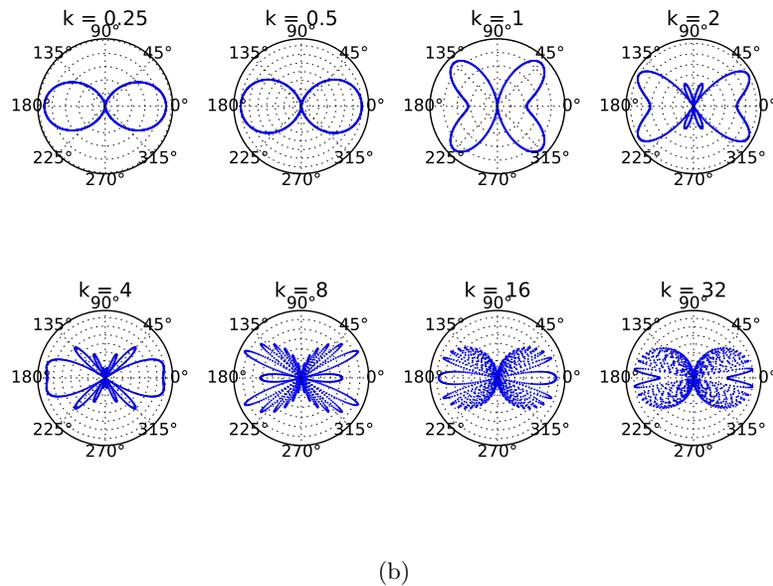
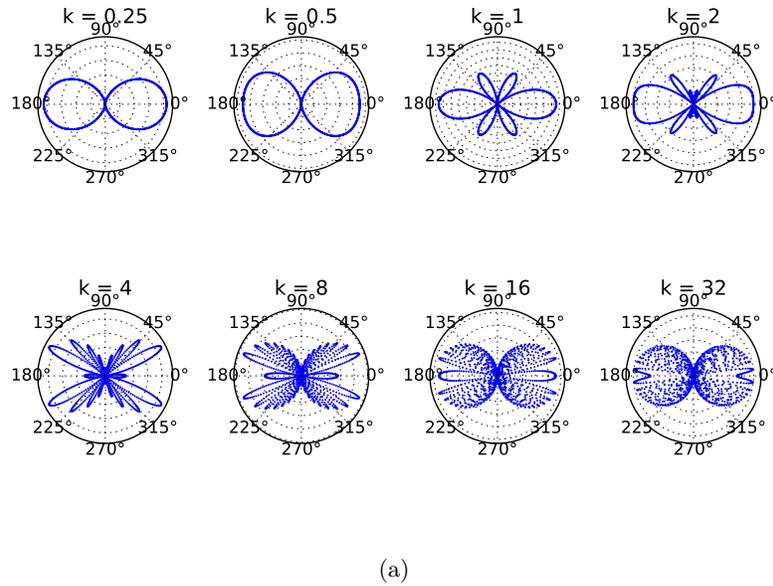


Figure 4: For a channel of gauge length $g = 8$ meters, the radius of these polar plots represents the axial strain sensitivity along a horizontal fiber oriented at that angle θ from the horizontal propagation direction of a P-wave which is also propagating at an angle ϕ from vertical. The ϕ values plotted here are (top) $\pi/2$ and (bottom) $\pi/4$. Unlike the case where only point-wise sensitivity is measured, the angular sensitivities varies significantly with angle. [ER] eileen2/. PChannSensPiHalf,PChannSensPiQuart

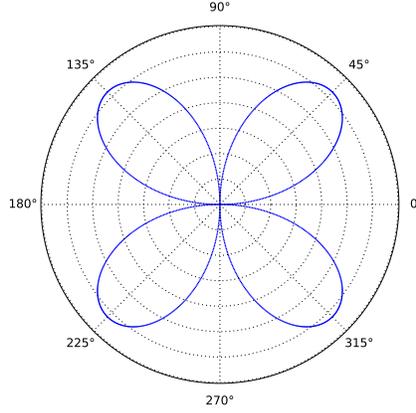


Figure 5: The radius of this polar plot represents the axial strain sensitivity along a horizontal fiber oriented at that angle θ from the horizontal propagation direction of a Love wave. Angular sensitivity is independent of wavenumber for a point measurement. [ER] eileen2/. LSens

suggesting that no matter what angle the plane wave is traveling, the orthogonal fibers will always record -1 times each other.

If we had a fiber channel of gauge length g oriented at the angle θ from the x -axis (direction of propagation) spanning $(x - \frac{g}{2} \cos(\theta), y - \frac{g}{2} \sin(\theta), z)$ to $(x + \frac{g}{2} \cos(\theta), y + \frac{g}{2} \sin(\theta), z)$, that channel would actually measure:

$$\begin{aligned}
 \bar{\sigma}(x, y, z, t) &= \int_{-g/2}^{g/2} -ik \cos(\theta) \sin(\theta) (Ae^{-i\eta_1 kz} + Be^{i\eta_1 kz}) e^{ik(ct - (x + \nu \cos(\theta)))} d\nu \\
 &= -ik \cos(\theta) \sin(\theta) (Ae^{-i\eta_1 kz} + Be^{i\eta_1 kz}) e^{ik(ct-x)} \int_{-g/2}^{g/2} e^{-ik\nu \cos(\theta)} d\nu \\
 &= -ik \cos(\theta) \sin(\theta) (Ae^{-i\eta_1 kz} + Be^{i\eta_1 kz}) e^{ik(ct-x)} \left[\frac{e^{-ik\nu \cos(\theta)}}{-ik \cos(\theta)} \right]_{-g/2}^{g/2} \\
 &= \sin(\theta) (Ae^{-i\eta_1 kz} + Be^{i\eta_1 kz}) e^{ik(ct-x)} \left[e^{-ik\frac{g}{2} \cos(\theta)} - e^{ik\frac{g}{2} \cos(\theta)} \right] \\
 &= 2i \sin(\theta) (Ae^{-i\eta_1 kz} + Be^{i\eta_1 kz}) e^{ik(ct-x)} \sin\left(-\frac{kg}{2} \cos(\theta)\right) \quad (20)
 \end{aligned}$$

which again simplifies into a geometry- and wavenumber-dependent oscillatory part $(Ae^{-i\eta_1 kz} + Be^{i\eta_1 kz})e^{-ikx}$, a time-dependent oscillatory part e^{ikct} , and a geometry- and wavenumber-dependent amplitude part proportional to $\sin(\theta) \sin(-\frac{kg}{2} \cos(\theta))$. As seen in Figure 6, we get more frequency notches as kg increases, but note that for Love waves, the sensitivity frequency notches are imprinted over a four-lobe pattern similar to the single-point $\cos(\theta) \sin(\theta)$ sensitivity. These fiber channels are not sensitive at all to Love waves traveling in-line or orthogonal to the fiber, but rather are sensitive to waves travelling at angles. This leads to data that are very counter-intuitive for people used to particle velocity measurements, as a horizontal component of a 3C geophone would be maximally sensitive to Love waves propagating orthogonally to that component.

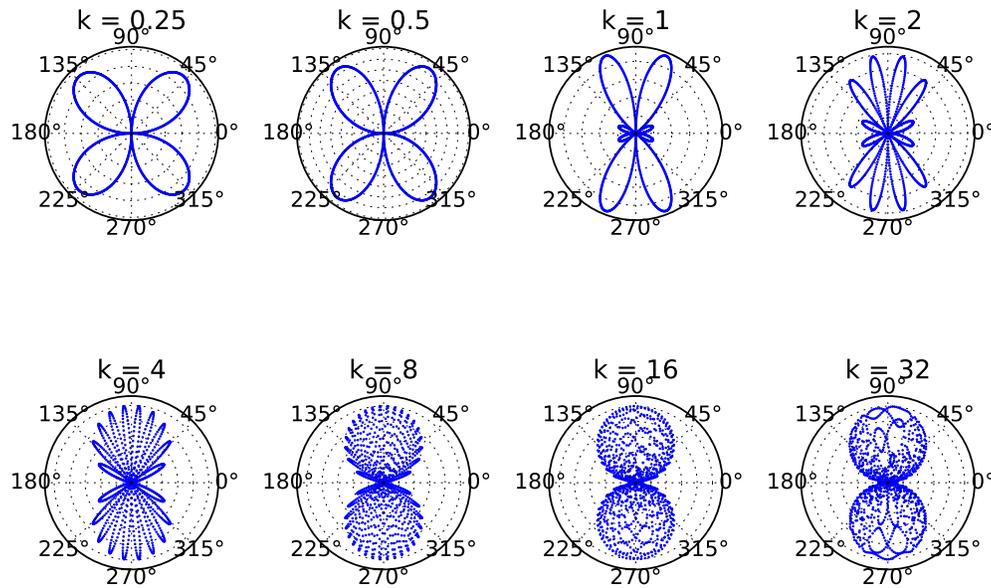


Figure 6: The radius of these polar plots represents the axial strain sensitivity of a channel along a horizontal fiber oriented at that angle θ from the horizontal propagation direction of a Love wave. We assume a gauge length of $g = 8$ meters, and show sensitivity for several wavenumbers, k , to emphasize that angular sensitivity is highly dependent on wavenumber, changing from a 4-lobe to almost being a 2-lobe structure (actually made up of many narrow lobes). [ER] eileen2/. LChannSens

S-Waves

Here, we follow Pujol (2003) in the simple model that if we have an S-wave propagating in the $x - z$ plane at an angle ϕ from the vertical axis, then the SH and SV motions can be described in simple cases by:

$$\mathbf{u}(x, y, z, t) = e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} (B \cos(\phi), C, B \sin(\phi)) \quad (21)$$

where β is the propagation velocity. In this case the axial strain at a point on a horizontal fiber oriented at an angle θ from the x -axis is:

$$\begin{aligned} \sigma(x, y, z, t) &= \cos^2(\theta) \frac{\partial u_x}{\partial x} + \cos(\theta) \sin(\theta) \frac{\partial u_y}{\partial x} \\ &= \cos^2(\theta) B \cos(\phi) (-i\omega \sin(\phi)/\beta) e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} \\ &\quad + \cos(\theta) \sin(\theta) C (-i\omega \sin(\phi)/\beta) e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} \\ &= \frac{-i\omega \sin(\phi)}{\beta} e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} \cos(\theta) (B \cos(\theta) \cos(\phi) + C \sin(\theta)) \end{aligned} \quad (22)$$

The factor in the point-wise axial strain sensitivity dependent on propagation angles can be seen in Figures 7(a) and 7(b). The general trend is that we have a four-lobe structure, but the pair (and the angle of emphasis) of lobes that is larger depends on ϕ but is independent of k .

Then if we wish to calculate the response of an orthogonal horizontal fiber at the same point, it would be:

$$\begin{aligned} \sigma^\perp(x, y, z, t) &= \frac{-i\omega \sin(\phi)}{\beta} e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} \cos(\theta + \pi/2) (B \cos(\theta + \pi/2) \cos(\phi) \\ &\quad + C \sin(\theta + \pi/2)) \\ &= \frac{-i\omega \sin(\phi)}{\beta} e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} \sin(\theta) (B \sin(\theta) \cos(\phi) - C \cos(\theta)) \end{aligned} \quad (23)$$

Looking at just the last factors of that expression, we see that one term, $B \cos^2(\theta) \cos(\phi)$ changes to $B \sin^2(\theta) \cos(\phi)$, so it follows the previous trend in P and Rayleigh waves of a \tan^2 amplitude factor between orthogonal lines. The other term, $C \cos(\theta) \sin(\theta)$ becomes $-C \cos(\theta) \sin(\theta)$, so the orthogonal lines have a sign flip between their observations of this term, much like the Love waves.

In fact, this makes sense when we consider that when $\phi = \pi/2$ (the wave is travelling horizontally), and $\cos(\theta) = 0$, so it simplifies to see the same trend as Love waves. However, a shear wave has more degrees of freedom and as such contains a mix of behaviors.

Consider a horizontal fiber channel of gauge length g centered at (x, y, z) oriented at an angle θ from the x -axis, so it spans $(x - \frac{g}{2} \cos(\theta), y - \frac{g}{2} \sin(\theta), z)$ to $(x + \frac{g}{2} \cos(\theta), y + \frac{g}{2} \sin(\theta), z)$,

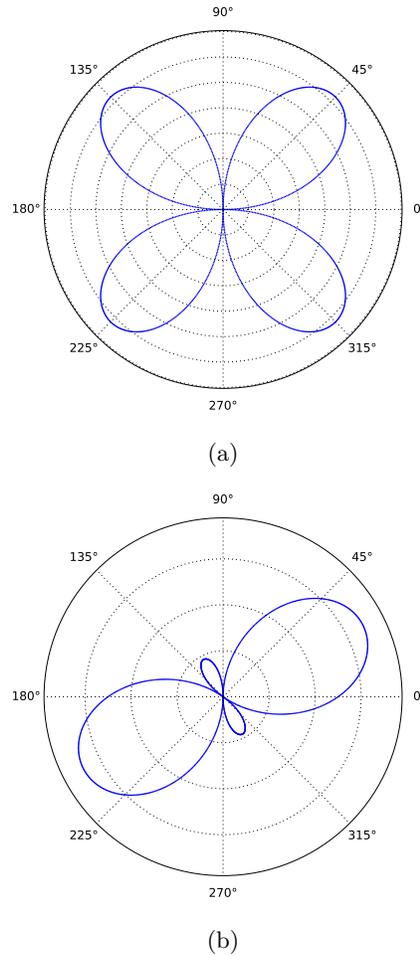


Figure 7: The radius of these polar plots represents the axial strain sensitivity along a horizontal fiber oriented at that angle θ from the horizontal propagation direction of an S-wave which is also propagating at an angle ϕ from vertical. The ϕ values plotted here are (top) $\pi/2$ and (bottom) $\pi/4$. The shape of the sensitivity is significantly affected by how much energy is propagating vertically, but is independent of wavenumber k . [ER]

eileen2/. SSensPiHalf,SSensPiQuart

that channel would actually measure:

$$\begin{aligned}
\bar{\sigma}(x, y, z, t) &= \int_{-g/2}^{g/2} \frac{-i\omega \sin(\phi)}{\beta} e^{i\omega(t - ((x + \nu \cos(\theta)) \sin(\phi) - z \cos(\phi))/\beta)} \cos(\theta) \dots \\
&\quad \times (B \cos(\theta) \cos(\phi) + C \sin(\theta)) d\nu \\
&= -\frac{i\omega \sin(\phi)}{\beta} e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} \cos(\theta) (B \cos(\theta) \cos(\phi) + C \sin(\theta)) \dots \\
&\quad \times \int_{-g/2}^{g/2} e^{-i\omega \nu \cos(\theta) \sin(\phi)/\beta} d\nu \\
&= e^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} (B \cos(\theta) \cos(\phi) + C \sin(\theta)) \left[e^{-i\omega \nu \cos(\theta) \sin(\phi)/\beta} \right]_{-g/2}^{g/2} \\
&= 2ie^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} (B \cos(\theta) \cos(\phi) + C \sin(\theta)) \sin\left(\frac{\omega g \cos(\theta) \sin(\phi)}{2\beta}\right).
\end{aligned}$$

Using the fact that $\omega/\beta = k$ our wavenumber, this simplifies to

$$\bar{\sigma}(x, y, z, t) = 2ie^{i\omega(t - (x \sin(\phi) - z \cos(\phi))/\beta)} (B \cos(\theta) \cos(\phi) + C \sin(\theta)) \sin\left(\frac{-kg}{2} \cos(\theta) \sin(\phi)\right) \quad (24)$$

This easily factors into a temporal oscillatory part $e^{i\omega t}$, a geometry-dependent oscillatory part $e^{-i\omega(x \sin(\phi) - z \cos(\phi))/\beta}$, and a geometry-dependent amplitude factor proportional to $(B \cos(\theta) \cos(\phi) + C \sin(\theta)) \sin(-kg \cos(\theta) \sin(\phi)/2)$.

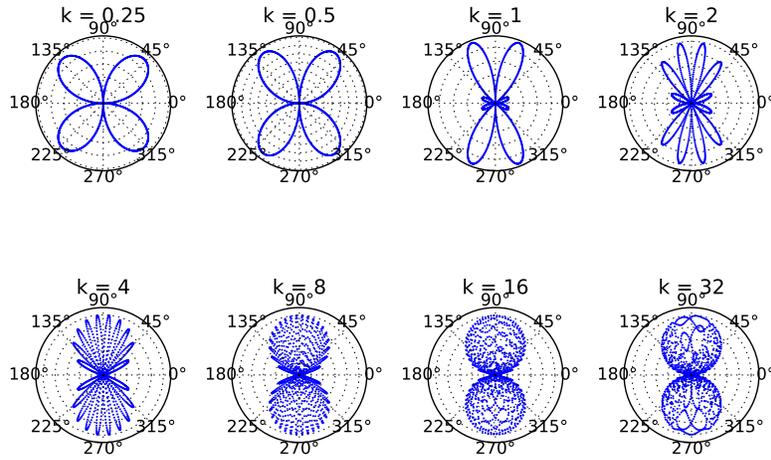
When $\phi = 0$, the S-wave is traveling vertically. We expect zero sensitivity to both the SH and SV wave motions, which is what we get from that geometry-dependent amplitude factor. Assume $\phi = \pi/2$, so it is an S-wave traveling horizontally. Then the sensitivity to plane waves coming in from the angle θ is seen in Figures 8(a) and 8(b) assuming $B = C = 1$. Again, the sensitivity gets more notches as kg grows, and in this case, the sensitivity shape starts out with the four-lobed shape similar to Love waves. One might wonder how getting further from the horizontal propagation direction affects this angular response, so the response for $\phi = \pi/4$ is plotted assuming $B = C = 1$ for a variety of kg values. Note that as k increases, the sensitivity approaches a two-lobe shape but is rotated depending on ϕ .

IMPLICATIONS FOR SURFACE WAVE CROSS-CORRELATIONS

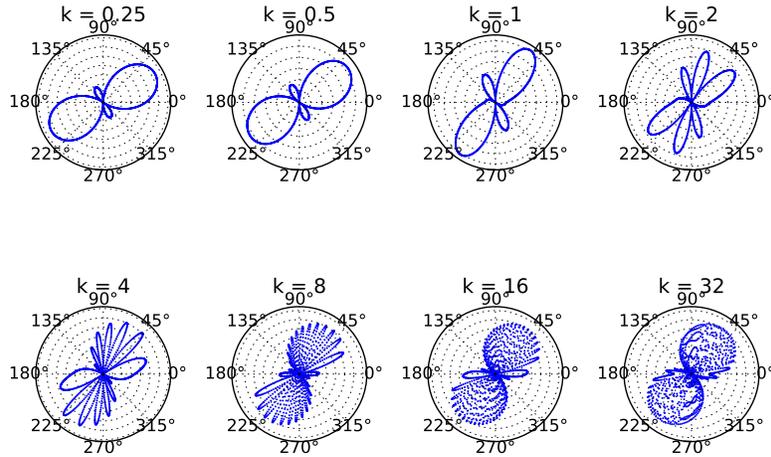
During ambient noise interferometry, we are often extracting surface waves responding to virtual sources. These surface waves we extract can be used to infer near-surface features. Thus, we focus on the sensitivity of the cross-correlation of two receivers both detecting the same plane-wave source. The same sensitivity analysis could be repeated for body waves, although it is rare that researchers are able to extract body waves from ambient noise.

Rayleigh waves

First, let's look at how sensitive the cross-correlation of two colinear channels oriented at an angle θ from the x -axis are to planar Rayleigh waves propagating along the x -direction. Let's say the first channel spans $(-\frac{d}{2} \cos(\theta) - \frac{g}{2} \cos(\theta), -\frac{d}{2} \sin(\theta) - \frac{g}{2} \sin(\theta), 0)$



(a)



(b)

Figure 8: The radius of these polar plots represents the axial strain sensitivity of a channel along a horizontal fiber oriented at that angle θ from the horizontal propagation direction of an S-wave which is also propagating at an angle ϕ from vertical. The ϕ values plotted here are (top) $\pi/2$, which represents horizontally propagating S-waves, and (bottom) $\pi/4$. Note that the relative angular sensitivity to horizontally propagating S-waves is the same as Love waves. We assume a gauge length $g = 8$ meters. We see that the shape of the sensitivity is significantly affected by how much energy is propagating vertically. For both ϕ values, the sensitivities approach a 2-lobe-like shape, but when the S-wave is coming in at $\phi = \pi/4$, the lobes are rotated. [ER] `eileen2/.SCHannSensPiHalf,SCHannSensPiQuart`

to $(-\frac{d}{2} \cos(\theta) + \frac{g}{2} \cos(\theta), -\frac{d}{2} \sin(\theta) + \frac{g}{2} \sin(\theta), 0)$, and the second channel spans $(\frac{d}{2} \cos(\theta) - \frac{g}{2} \cos(\theta), \frac{d}{2} \sin(\theta) - \frac{g}{2} \sin(\theta), 0)$ to $(\frac{d}{2} \cos(\theta) + \frac{g}{2} \cos(\theta), \frac{d}{2} \sin(\theta) + \frac{g}{2} \sin(\theta), 0)$. Following Equation 7, signals recorded by each of those channels are:

$$\begin{aligned}\bar{\sigma}\left(-\frac{d}{2} \cos(\theta), -\frac{d}{2} \sin(\theta), 0, t\right) &= 2i \cos(\theta)(A + iB\gamma_\beta)e^{ik(ct + \frac{d}{2} \cos(\theta))} \sin\left(\frac{kg}{2} \cos(\theta)\right) \\ \bar{\sigma}\left(\frac{d}{2} \cos(\theta), \frac{d}{2} \sin(\theta), 0, t\right) &= 2i \cos(\theta)(A + iB\gamma_\beta)e^{ik(ct - \frac{d}{2} \cos(\theta))} \sin\left(\frac{kg}{2} \cos(\theta)\right)\end{aligned}\quad (25)$$

Thus their cross-correlation would be:

$$\begin{aligned}C(\tau) &= \int_{-\infty}^{\infty} \bar{\sigma}\left(-\frac{d}{2} \cos(\theta), -\frac{d}{2} \sin(\theta), 0, t\right) \bar{\sigma}^*\left(\frac{d}{2} \cos(\theta), \frac{d}{2} \sin(\theta), 0, t + \tau\right) dt \\ &= \int_{-\infty}^{\infty} 4 \cos^2(\theta)(A^2 - B^2\gamma_\beta^2)e^{ik(-c\tau + d \cos(\theta))} \sin^2\left(\frac{kg}{2} \cos(\theta)\right) dt \\ &\approx 4 \cos^2(\theta)(A^2 - B^2\gamma_\beta^2)e^{ik(-c\tau + d \cos(\theta))} \sin^2\left(\frac{kg}{2} \cos(\theta)\right)\end{aligned}\quad (26)$$

That last approximation is because that integral does not actually converge, but the terms have no dependence on t . Note that an integrable wavelet could be used to overcome this obstacle, for instance, convolution with a Gaussian around a particular start time, and the same angularly-dependent term would factor out. The term above does describe the geometric dependence of the cross-correlation on propagation direction and the distance between fiber channels. It has a temporal oscillatory factor, $e^{-ikc\tau}$, a spatial oscillatory term proportional to the apparent wavenumber times the distance between channels $e^{kd \cos(\theta)}$, and a spatial factor proportional to $\cos^2(\theta) \sin^2(kg \cos(\theta)/2)$, plotted in Figure 9

Love waves

Now let's look at the angular sensitivity of the cross-correlation of two parallel channels distance d apart with start and end points directly across from each other (as in Figure 10). One channel runs from $(\frac{d}{2} \sin(\theta) - \frac{g}{2} \cos(\theta), -\frac{d}{2} \cos(\theta) - \frac{g}{2} \sin(\theta), 0)$ to $(\frac{d}{2} \sin(\theta) + \frac{g}{2} \cos(\theta), -\frac{d}{2} \cos(\theta) + \frac{g}{2} \sin(\theta), 0)$, and the other runs from $(-\frac{d}{2} \sin(\theta) - \frac{g}{2} \cos(\theta), \frac{d}{2} \cos(\theta) - \frac{g}{2} \sin(\theta), 0)$ to $(-\frac{d}{2} \sin(\theta) + \frac{g}{2} \cos(\theta), \frac{d}{2} \cos(\theta) + \frac{g}{2} \sin(\theta), 0)$.

Following Equation 20, signals recorded by each of those channels are:

$$\begin{aligned}\bar{\sigma}\left(\frac{d}{2} \sin(\theta), -\frac{d}{2} \cos(\theta), 0, t\right) &= 2i \sin(\theta)(A + B)e^{ik(ct - \frac{d}{2} \sin(\theta))} \sin\left(\frac{-kg}{2} \cos(\theta)\right) \\ \bar{\sigma}\left(-\frac{d}{2} \sin(\theta), \frac{d}{2} \cos(\theta), 0, t\right) &= 2i \sin(\theta)(A + B)e^{ik(ct + \frac{d}{2} \sin(\theta))} \sin\left(\frac{-kg}{2} \cos(\theta)\right)\end{aligned}\quad (27)$$

It follows that their cross-correlation would be:

$$\begin{aligned}C_\tau &= \int_{-\infty}^{\infty} \bar{\sigma}\left(\frac{d}{2} \sin(\theta), -\frac{d}{2} \cos(\theta), 0, t\right) \bar{\sigma}^*\left(-\frac{d}{2} \sin(\theta), \frac{d}{2} \cos(\theta), 0, t\right) dt \\ &= \int_{-\infty}^{\infty} 4 \sin^2(\theta)(A + B)^2 e^{-ik(c\tau + d \sin(\theta))} \sin^2\left(\frac{-kg}{2} \cos(\theta)\right) dt \\ &\approx 4 \sin^2(\theta)(A + B)^2 e^{-ik(c\tau + d \sin(\theta))} \sin^2\left(\frac{-kg}{2} \cos(\theta)\right)\end{aligned}\quad (28)$$

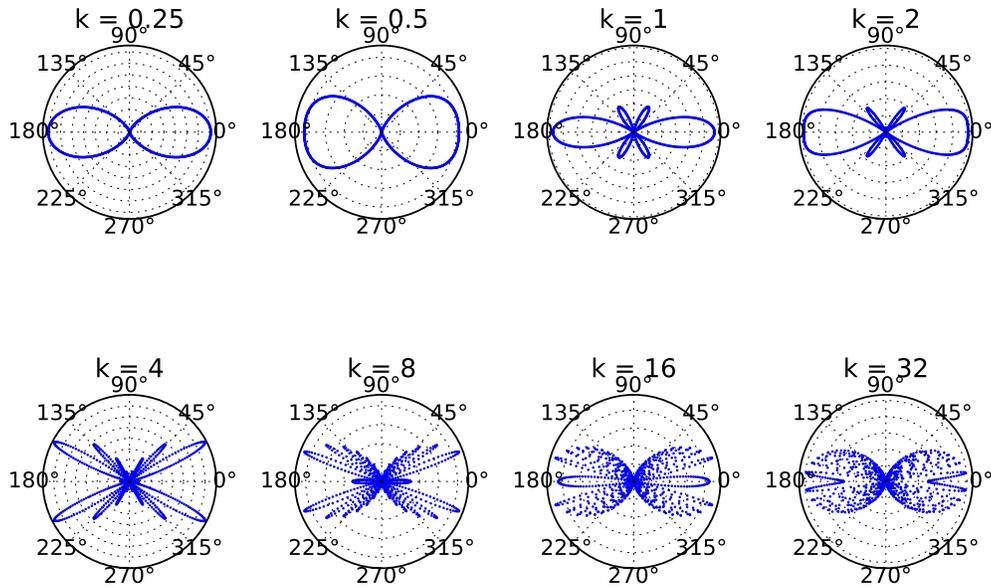


Figure 9: Assuming a set gauge length $g = 8$ meters, we plot the geometry-dependent amplitude factor as the radius at each angle θ , where that marks the angle between two parallel fiber channels and the propagation direction of a Rayleigh wave. Notice that as the wavenumber k increases, the sensitivity pattern moves away from a two-lobe pattern, then fills back in the two-lobe pattern. Thus at low and high wavenumbers, the cross-correlation emphasizes waves traveling straight from one channel to the other. [ER]

eileen2/. RXCorrSens

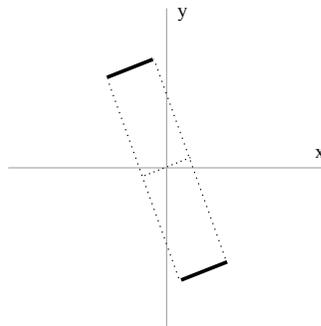


Figure 10: The dark lines represent two channels directly across from each other to be cross-correlated as they react to Love plane waves. [NR] eileen2/. chOrientationLove

where again, we use the approximate symbol to indicate that the monochromatic wavelet is not integrable. This factors into an oscillatory time part $e^{ikc\tau}$, an oscillatory spatial part dependent on the apparent wavenumber and distance between channels $e^{-idk \sin(\theta)}$, and a geometry-dependent amplitude factor proportional to $\sin^2(\theta) \sin^2(\frac{-kg}{2} \cos(\theta))$.

However, the geometry-dependent amplitude factor, plotted in Figure 11 for many k values, starts at small wavenumbers (low frequencies) looking like the point measurements of Figure 5. At higher, more realistic wavenumbers, it approaches a two-lobe shape which is primarily sensitive to Love waves mostly traveling in-line with the radial direction between the two channels. However, if one is dealing with very low frequency/wavenumber signals, one absolutely must be aware that the four-lobe sensitivity pattern could lead to false apparent velocities in the interferometry of signals coming from all directions (as is assumed in ambient noise interferometry with a homogeneous angular distribution of far-field sources) which must be corrected for. Additionally, even at higher wavenumbers, the zero-phase signal will never truly be picked up, so careful analysis of whether a Green's function can actually be extracted should be performed in the future.

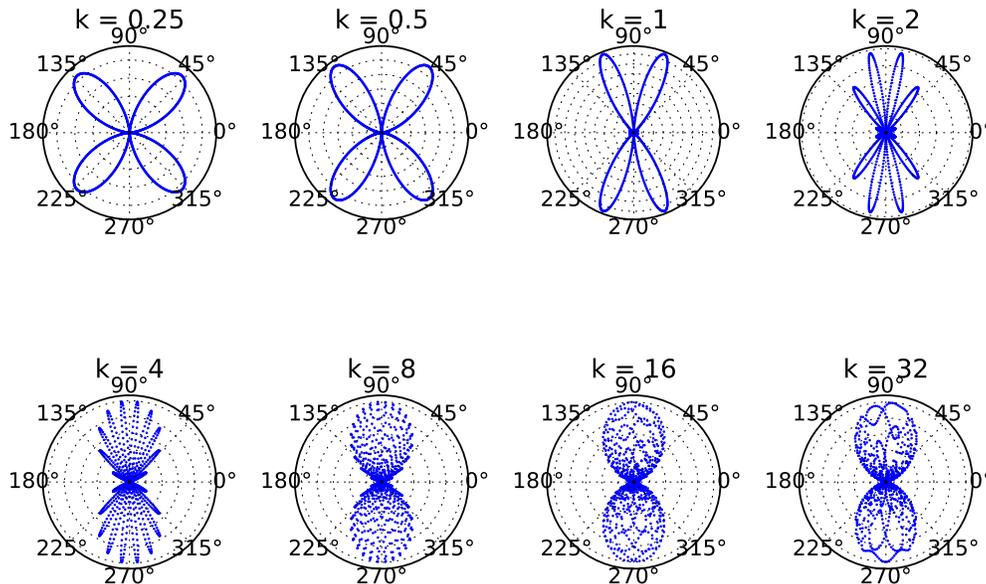


Figure 11: Assuming a set gauge length $g = 8$ meters, we plot the geometry-dependent amplitude factor as the radius at each angle, θ , where θ is the angle between two parallel fiber channels and the propagation direction of a Love wave. Notice that as the wavenumber k increases, the sensitivity pattern moves from a four-lobe pattern to a pattern with more lobes, then towards a two-lobe pattern which emphasizes waves traveling straight from one channel to the other. [ER] eileen2/. LXCrrSens

PRACTICAL IMPLICATIONS OF SENSITIVITY

How should you choose your gauge length? All figures in this report have been generated assuming an 8 meter gauge length, and typical gauge length settings in seismic imaging/-monitoring are 8-20 meters. However, you can use the same equations and generate sensitivity plots for a set k with many g values, with an increasing g having the same effect as an increasing k . Figures 6, 8(a), 8(b) and 11 suggest that we would not want to use a very small g if we care about detecting S- and Love waves. This is particularly the case for ambient noise Love wave interferometry because the sensitivity at small kg is primarily to sources off at angles leading to false apparently high velocities. For a given k we can correct for these biases, but since the angular sensitivity peaks rotate as k increases for a set g , as seen in Figure 11, we would need to correct for these biases independently for different wavelengths. This further suggests that for small kg the DAS interferometry results may include false dispersion trends. Thus, by using a g that is large enough for all expected k , we do not need to worry as much about these issues for Love wave interferometry. Rayleigh wave interferometry has much less of this issue at any kg .

In fact, for typical seismic wavelengths and gauge length settings, we tend to be on the low end of k values shown in this report. For instance, at the high end of seismic wavenumbers, a wavelength of 10 meters means a wavenumber $k \approx 0.63$, too low to see many of the unusual lobe shapes for a gauge length of 8 meters. If the gauge length increases, then lower wavenumbers can demonstrate these unusual behaviors for a fixed k , but as you continue to increase the gauge length, the sensitivity pattern fills in again (just as it would for very high k). However, seismic waves at these very high wavenumbers tend to die out quickly, so it is unlikely we could get much signal in these more filled in sensitivity trends seen for large k . But, we can force this fill-in by mimicking larger gauge lengths: averaging neighboring non-overlapping channels. This must be balanced with the loss in resolution. Further work is needed to develop a method to combine results from different apparent gauge lengths.

Consider that what may be an acceptable sensitivity pattern for type of wave may not be acceptable for another type. If we wish to force a sensitivity pattern more like a small kg , we should be able to decrease the apparent g in post-processing if (a) we record overlapping channels or (b) we record on a box that spits out two sets of data each corresponding to a different gauge length and those gauge lengths aren't multiples of each other. More details on the theory of why this should work can be found in the Appendix. Future work will include testing this theory.

CONCLUSIONS AND FUTURE WORK

In this report we derive the sensitivity patterns for horizontal DAS arrays reacting to various plane waves: P, S, Rayleigh and Love. Further, we compare the axial strain on orthogonal horizontal fibers to better understand trends, particularly polarity flips, observed at corners of existing DAS arrays. While point-wise axial strain starts to give us some insight into those trends, these results can mislead us with regards to the true data in two ways:

- The existence of frequency notches at certain frequencies and angles is not apparent in point strains,

- As wavenumber or gauge length increase, lobes in sensitivity may merge, expand or contract, but only when integrating over gauge lengths.

Thus, we also show sensitivity results for P, S, Rayleigh and Love waves detected through the average axial strain along a segment of fiber of length g . We extend these results to cross-correlation sensitivities of colinear channels detecting Rayleigh waves and parallel channels detecting Love waves. Evidence suggests that for larger wavenumbers, the cross-correlation is primarily sensitive to waves traveling directly from one fiber to another. These results suggest there is some hope for ambient noise interferometry to yield reasonable results from DAS data.

However, there are some small details that we are continuing to investigate both theoretically and through the use of computational modeling:

- Parallel channels' lack of sensitivity to Love waves traveling directly through the radial direction between them could lead to small biases in velocities picked from cross-correlations. In the limit as kg grows, is ambient noise interferometry with an ideal noise field guaranteed to converge to Green's function extraction?
- For low wavenumbers in both the Love and Rayleigh wave interferometry cases, it is apparent that velocities picked from Love wave interferometry must be corrected for. What is a practical method to correct for this bias given limited knowledge of the subsurface velocity model?

ACKNOWLEDGEMENTS

E. Martin has also been supported in part by the DOE CSGF under grant number DE-FG02-97ER25308, and a Schlumberger Innovation Fellowship. We would like to thank OptaSense, particularly M. Karrenbach and S. Cole for the use of their ODH-3 interrogator unit and for many helpful suggestions as we worked with data from SDASA-1 to identify gaps in our understanding of sensitivity. We would also like to thank Jason Chang, Nori Nakata (University of Oklahoma), Nate Lindsey (University of California Berkeley), and Jonathan Ajo-Franklin (Lawrence Berkeley National Lab) for their insights and assistance.

REFERENCES

- Biondi, B., E. Martin, S. Cole, and M. Karrenbach, 2017, Earthquakes analysis using data recorded by the stanford das array: SEP Report, **168**.
- Bona, A., T. Dean, J. Correa, R. Pevzner, K. Tertyshnikov, and L. V. Zaanen, 2017, Amplitude and phase response of das receivers: 79th EAGE Conference & Exhibition Proceedings.
- Kuvshinov, B., 2016, Interaction of helically wound fibre-optic cables with plane seismic waves: Geophysical Prospecting, **64**, 671–688.
- Martin, E. and B. Biondi, 2017, Ambient noise interferometry across two-dimensional das arrays: Expanded Abstracts of the 87th Ann. Internat. Mtg.
- Pujol, J., 2003, Elastic wave propagation and generation in seismology: Cambridge University Press.

APPENDIX: POINTWISE SENSITIVITY SHAPES ARE THE SAME AS ZERO WAVENUMBER SHAPE

For all wave types, the shape of the angular sensitivity could be considered to be the ratio between the geometric amplitude sensitivity term for any two angles, θ_1 and θ_2 . The Rayleigh wave results were verified in this paper earlier. Here we verify this result for P-waves, Love waves, S-waves, which are nearly identical.

P-waves

Ratio between angles pointwise:

$$R(\theta_1, \theta_2) = \frac{\cos^2(\theta_1) \sin^2(\phi_1)}{\cos^2(\theta_2) \sin^2(\phi_2)} \quad (29)$$

The ratio between angles for a channel measurement:

$$R_c(\theta_1, \theta_2, k) = \frac{\cos(\theta_1) \sin(\phi_1) \sin(0.5kg \cos(\theta_1) \sin(\phi_1))}{\cos(\theta_2) \sin(\phi_2) \sin(0.5kg \cos(\theta_2) \sin(\phi_2))} \quad (30)$$

Take limit as $k \rightarrow 0$ with L'Hospital's rule:

$$\begin{aligned} R_c(\theta_1, \theta_2, 0) &= \frac{\cos(\theta_1) \sin(\phi_1) \cos(0.5 * 0 * g \cos(\theta_1) \sin(\phi_1)) 0.5g \cos(\theta_1) \sin(\phi_1)}{\cos(\theta_2) \sin(\phi_2) \cos(0.5 * 0 * g \cos(\theta_2) \sin(\phi_2)) 0.5g \cos(\theta_2) \sin(\phi_2)} \\ &= \frac{\cos^2(\theta_1) \sin^2(\phi_1)}{\cos^2(\theta_2) \sin^2(\phi_2)} \end{aligned} \quad (31)$$

Love waves

Ratio between angles pointwise:

$$R(\theta_1, \theta_2) = \frac{\cos(\theta_1) \sin(\theta_1)}{\cos(\theta_2) \sin(\theta_2)} \quad (32)$$

Ratio between angles for a full channel measurement:

$$R_c(\theta_1, \theta_2, k) = \frac{\sin(\theta_1) \sin(-0.5kg \cos(\theta_1))}{\sin(\theta_2) \sin(-0.5kg \cos(\theta_2))} \quad (33)$$

Take limit as $k \rightarrow 0$ and use L'Hospital's rule

$$\begin{aligned} R_c(\theta_1, \theta_2, 0) &= \frac{-0.5g \cos(\theta_1) \sin(\theta_1) \cos(-0.5 * 0 * g \cos(\theta_1))}{-0.5g \cos(\theta_2) \sin(\theta_2) \cos(-0.5 * 0 * g \cos(\theta_2))} \\ &= \frac{\cos(\theta_1) \sin(\theta_1)}{\cos(\theta_2) \sin(\theta_2)} \end{aligned} \quad (34)$$

S-waves

Ratio between angles pointwise:

$$R(\theta_1, \theta_2) = \frac{\sin(\phi_1) \cos(\theta_1) (B \cos(\theta_1) \cos(\phi_1) + C \sin(\theta_1))}{\sin(\phi_2) \cos(\theta_2) (B \cos(\theta_2) \cos(\phi_2) + C \sin(\theta_2))} \quad (35)$$

Ratio between angles for a full channel measurement:

$$R_c(\theta_1, \theta_2, k) = \frac{(B \cos(\theta_1) \cos(\phi_1) + C \sin(\theta_1)) \sin(-0.5kg \cos(\theta_1) \sin(\phi_1))}{(B \cos(\theta_2) \cos(\phi_2) + C \sin(\theta_2)) \sin(-0.5kg \cos(\theta_2) \sin(\phi_2))} \quad (36)$$

Take limit as $k \rightarrow 0$ and use L'Hospital's rule:

$$\begin{aligned} R_c(\theta_1, \theta_2, k) &= \frac{-0.5g \cos(\theta_1) \sin(\phi_1)(B \cos(\theta_1) \cos(\phi_1) + C \sin(\theta_1)) \cos(-0.5 * 0 * g \cos(\theta_1) \sin(\phi_1))}{-0.5g \cos(\theta_2) \sin(\phi_2)(B \cos(\theta_2) \cos(\phi_2) + C \sin(\theta_2)) \cos(-0.5 * 0 * g \cos(\theta_2) \sin(\phi_2))} \\ &= \frac{\sin(\phi_1) \cos(\theta_1)(B \cos(\theta_1) \cos(\phi_1) + C \sin(\theta_1))}{\sin(\phi_2) \cos(\theta_2)(B \cos(\theta_2) \cos(\phi_2) + C \sin(\theta_2))} \end{aligned} \quad (37)$$

APPENDIX: RECOVERING SUB-GAUGE-LENGTH RESOLUTION FROM DAS

Say we have a DAS interrogator unit with channel spacing set at c meters, and gauge length g meters, and it's attached to a fiber of length nc meters. Then when the interrogator sends a laser pulse down the fiber, we observe $n - (g/c) + 1$ measurements, $m_0, m_1, \dots, m_{n-(g/c)}$, where

$$m_i = K \int_{ic}^{ic+g} s(x) dx \quad (38)$$

where s is the axial strain along the fiber parameterized by x , and K is some constant so m_i is a phase shift quasi-linearly proportional to s .

Frequently when geophysicists start using DAS, they have some confusion between gauge length and channel spacing, assuming that we always have $g = c$. This is often not the case, and in fact leads people to assume that their data have much smaller resolution than the raw data truly have. For instance, in SDASA-1, our active mode recording is $g = 7.14$, $c = 1.02$, so a new sensor of length 7.14 m begins every 1.02 m. Although we start a new measurement channel every 1.02 m, each measurement is the average axial strain over a 7.14 m segment.

Assuming that $g > c$ and that $g/c \in \mathbb{Z}_+$, we would like to reconstruct $\{u_i\}_{i=0}^{n-1}$ where

$$u_i = K \int_{ic}^{(i+1)c} s(x) dx \quad (39)$$

so one really can get the average strain for every meter along the fiber. We can express each measurement as a combination of the smaller scale measurements that could have been taken with a shorter gauge length as follows:

$$\begin{aligned} m_i &= K \int_{ic}^{ic+g} s(x) dx \\ &= K \left[\int_{ic}^{(i+1)c} s(x) dx + \int_{(i+1)c}^{(i+2)c} s(x) dx + \dots + \int_{(i+(g/c)-1)c}^{(i+(g/c))c} s(x) dx \right] \\ &= u_i + u_{i+1} + \dots + u_{i+(g/c)-1} \end{aligned} \quad (40)$$

This can be written as an underdetermined linear system where \mathbf{u} is the unknown we wish to solve for:

$$\begin{bmatrix} m_0 \\ m_1 \\ \vdots \\ m_{n-(g/c)} \end{bmatrix} = \begin{bmatrix} 1 & \dots & 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 1 & 0 & 0 \\ \vdots & & \ddots & & \ddots & \vdots \\ \vdots & & & 0 & 1 & \dots & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-1} \end{bmatrix} \quad (41)$$

Let's call this matrix W , the weight matrix. For simplicity, we assume the integrals for averaging strain are uniformly weighted, so the bands of values in W are all 1's, but you could potentially imagine using other weightings. This is a $n - (g/c) + 1$ by n matrix, so $\mathbf{m} = W\mathbf{u}$ is an underdetermined system.

Possible solutions

We propose two potential fine-scale measurement recovery methods:

- Keep the underdetermined system, perform a regularized inversion, and play around with starting points or random noise to see how sensitive your system is to its null-space.
- Add enough independent rows to W so it will be invertible, or even add enough rows that it becomes an overdetermined system. This can be done by making simultaneous measurements with a gauge length g' that's not a factor or multiple of g .

Regularized inversion

How it works: Find the u that minimizes the misfit, while not blowing up:

$$\min_{u \in \mathbb{R}^n} \|Wu - m\|_2^2 + \lambda \|u\|_2^2 \quad (42)$$

although $\|u\|_2^2$ could be replaced with some other regularization.

Pros: Does not require any modifications to acquisition. Can be tested using the 1 meter spaced nodes from the active survey in March 2017 on Stanford campus.

Cons: Not guaranteed to return v_i for any i . Likely to smooth out measurements (depending on regularization type). It is not totally obvious how to choose λ .

Multiple simultaneous gauge lengths

How it works: At (roughly) the same time that the first gauge length's measurements are taken, also measure $\{m'_i\}_{i=0}^{n-(g'/c)}$, defined in the same way but for gauge length $g' > c$, where g' is neither a multiple or a divisor of g . Then new rows are added to the weight matrix representing the equations:

$$m'_i = u_i + u_{i+1} + \dots + u_{i+(g'/c)-1} \quad (43)$$

so the new matrix, W' has the same structure as W except g'/c bands instead of g/c bands, and it contains $n - (g'/c) + 1$ rows. Thus the system

$$\begin{bmatrix} \mathbf{m} \\ \mathbf{m}' \end{bmatrix} = \begin{bmatrix} W \\ W' \end{bmatrix} \mathbf{u} \quad (44)$$

which is an overdetermined system that can be solved via least-squares. Since this system must be solved at each time step for u , it makes sense to directly invert the sparse matrix

$W^T W + W'^T W'$, then the solution can be done via a matrix multiplication. This can be done for more than two gauge lengths also, but two should be sufficient.

Note: if g' is a multiple or divisor of g , the system is still rank deficient. If g and g' are close (say 9 and 10 m) the condition number is lower (around a few hundred, relatively independent of n) and the system tends to behave better.

Pros: Guaranteed to return $\{u_i\}_{i=0}^{n-1}$ up to noise levels. Different gauge lengths (that aren't multiples of each other) also means reduced effect from frequency notches. Even with just two gauge lengths, the system is already overdetermined which helps with robustness in the face of noise.

Cons: Requires a different interrogator unit configuration than is available.

Catalog of Northern California earthquakes recorded by DAS

Siyuan Yuan, Eileen R. Martin, Jason P. Chang, Steve Cole, and Biondo Biondi

ABSTRACT

We catalogued more than 800 seismic events recorded at Stanford Distributed Acoustic Sensing Array (SDASA-1) from September 2016 to August 2017. The catalog is being continuously updated as new events occur. We have developed open-source interfaces so that users can query the database and extract earthquake recordings efficiently. Pulling the data via the interfaces, we performed signal repeatability analyses for blasts at nearby Permanente Quarry and nearby weak earthquakes from Ladera and Felt Lake. We found that geographically close events could have repeatable signals in terms of S-wave arrivals and surface-wave phase changes. With rich event recordings, the catalog enables us to extract and characterize distant and weak events, which we will use to quantify our array's sensitivity and study event detection and noise attenuation algorithms in future work.

INTRODUCTION

Distributed Acoustic Sensing (DAS) holds great promise for application in cost-effective monitoring of microseismic signals and detecting earthquakes. Unlike the mainstream work previously done using DAS (e.g., using fibers in wells or burying the array in trenches), Stanford DAS Array-1 (SDASA-1) uses a fiber-optic cable laying in an already existing polyvinyl chloride (PVC) conduit buried in the ground that makes the installation more convenient and economic (Martin et al., 2017; Biondi et al., 2017). Understanding how well this type of cost-effective array records seismic events is one of our objectives. Some preliminary event analyses have been done. Through analysis of six seismic events recorded by SDASA-1, Biondi et al. (2017) show the recorded data can provide us with valuable information, thereby demonstrating the suitability of our DAS array for event detection. We would like to get a better understanding of our array by further analyzing many more events. The location of SDASA-1 in a tectonically active location allows us to record hundreds of events per year.

Due to the large number of the events, manual data management and extraction become infeasible. Therefore we built an event catalog system to manage the data automatically. The catalog was built based on the United States Geographical Survey (USGS) online database, and we also provide a program that can pull the broadband data recorded at the Jasper Ridge Seismic Station (JRSC) station for comparative studies. Additionally, we offer open-source interfaces for the users to select only events of interest and to extract the event data recorded by DAS efficiently. Using the interfaces, we performed the signal repeatability analysis in a more efficient manner. Here we analyze data from blasts at Permanente Quarry and several natural seismic events. This analysis shows some repeatable patterns.

In the next section, we give an overview of the passive data from which we extract earthquake recordings. Then we show how we manage the passive data, how to build the

catalog, and how the catalog interfaces work. We then provide examples using the database to perform repeatability analyses for 67 quarry blasts, two nearby natural earthquakes from Ladera and two natural earthquakes from Felt Lake; and we present the repeatable wave arrival timings and waveform patterns we found.

OVERVIEW OF PASSIVE DATA

Martin et al. (2017) and Biondi et al. (2017) show the SDASA-1 array design, geometry, and experiment setup. We used passive data recorded by SDASA-1 to extract earthquake recordings. To interpret our results in the following sections, we first give an overview of the data.

The data have been recorded by 626 channels along the SDASA-1 since September 2016. SDASA-1 is a double-loop array with the layout outlined by the red line in Figure 1. At the end of the first loop, two fibers in the same jacket are spliced end-to-end. We numbered the channels in the interlacing order with a 4.08-meter effective spacing. The data are recorded by each channel on both fibers simultaneously.

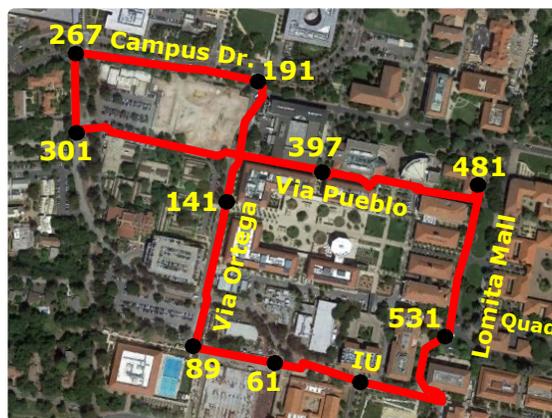


Figure 1: The layout of the fiber following telecommunications conduits overlaid on the map. The longest linear section is roughly 600 meters wide. Some deviations from straight lines had to occur due to existing conduit geometry constraints. [NR] [siyuan1/. Fig1-fiberMap](#)

There are two recording modes: active and passive. For the active mode, the sampling rate is 2500 samples per second; while 50 samples per second is the rate for the passive mode. The active data have 4x denser channel spacing: 1.02 meters. Active data and passive data require different processing procedures. Because the active mode accounts for a negligible proportion of recordings compared to the passive mode, we focus on the passive mode data with the 25 Hz Nyquist rate for now.

MANAGEMENT OF PASSIVE DATA

Our first step in data management was to organize files into smaller, easily navigated subdirectories. The passive data files initially show up in a directory called `/data/biondo/DAS` in SEG-Y format and are named based on the UTC time stamp of its first sample. For a limited number of files this could be the end of the story, but even simple file system operations like an `ls` become painfully slow when thousands of files are in one directory. Therefore, a python script called `checkOrganizeFiles.py` is run regularly to move files into subdirectories named as `/data/biondo/DAS/year/month/day`. For example, a file starting on December 10, 2016 at 13:01:54 and 351 milliseconds would show up as `/data/biondo/DAS/cbt_processed_20161210_130154.351+0000.sgy` and the next time `checkOrganizeFiles.py` is run, it will move to `/data/biondo/DAS/2016/12/10/cbt_processed_20161210_130154.351+0000.sgy`.

Further organization is based on the idea of *fileSet* objects, defined in `fileSet.py`. A *fileSet* is a collection of files named using a particular grammar that allows times to be associated with the names of files. Given a file name with that grammar, the *fileSet* can interpret the start time of the file (as a python *datetime* object), and given a start time, the *fileSet* can generate the name of a file that starts at that time. To define a *fileSet*, one first creates a list of strings, some of which are keywords tied to parts of the timestamp. For instance, when files initially show up in `/data/biondo/DAS`, the *fileSet* grammar that describes them is `['/data/biondo/DAS/cbt_processed_', 'year4', 'month', 'day', '-', 'hour24start0', 'minute', 'second', '.', 'millisecond', '+0000.sgy']`. The *fileSet* that describes files that have already been organized into subdirectories is `['/data/biondo/DAS/', 'year4', '/', 'month', '/', 'day', '/', 'cbt_processed_', 'year4', 'month', 'day', '-', 'hour24start0', 'minute', 'second', '.', 'millisecond', '+0000.sgy']`.

However, we do not always know the start time of every file. We just know that we want files that contain time A to time B. One clunky way to do this is every time you want data between A and B, you list all the files in the `/data/biondo/DAS/year/month/day` subdirectory for each `year/month/day` between A and B, use the *fileSet* described above to check their start times, also check the SEG-Y header with the file length, then generate a list of files. Instead, we streamline the process by organizing consecutive files into *regularFileSet* objects, which are a type of *fileSet* that describes files that are the same length, same sample rate, the end of one file is right before the start of the next, and all are named using the same grammar. The *regularFileSet* object is defined by the start time of the first file, the number of consecutive files, the grammar for naming, and the length in seconds of the files. A *regularFileSet* also tracks the end of its last file, can return the name of the file in that *regularFileSet* containing a particular time, and can build a list of file names that include the data between time A and time B.

Generally, the interrogator chugs along without interruption, so we often have *regularFileSets* made up of hundreds or thousands of files. But we needed to automate the tracking of all these *regularFileSets* because there are occasional hiccups: a software license gets renewed, a power outage, a switch between active and passive recording, among other reasons. To do this, we have written a script called `fillSQLiteDB.py` to build and add on to a SQL database `regFileSetLog.sqlite` where each row is a distinct *regularFileSet*. Since there are typically hundreds or thousands of files in a *regularFileSet*, this means that the SQL database contains relatively little metadata, making it easy to navigate and figure out which file names should be used to study a particular event or time window.

The *fillsQLiteDB.py* script requires the *sqlite3* package and accepts three command-line arguments. The first is a text file containing the paths to directories containing the SEG-Y files to be added to the SQL database. While users can specify the directories directly containing the SEG-Y files, they can also specify directories containing subdirectories of SEG-Y files. Additionally, users can include multiple directories to search through by specifying them on different lines. As an example, if users would like to add SEG-Y files from all of August and from only the first day of September, the text file would look like:

```
/data/biondo/DAS/2017/08/  
/data/biondo/DAS/2017/09/01/
```

Note that if there are already some files from August in the database, the script will be able to append *regularFileSets* to the existing database entries. The second argument is the path of the directory containing the database. By default, the database will be located in an associated subdirectory *log* and named *regFileSetLog.sqlite*. The third argument is a flag for whether to print out status updates to screen (1 for yes, 0 for no). The first few entries of the SQL database are shown in Figure 2.

	startTime	endTime	secondsBetweenFiles	nFiles
0	2016-09-02 17:38:53.923000	2016-09-02 17:43:53.923000	60.0000	5
1	2016-09-02 17:43:53.923000	2016-09-02 17:44:54.923000	61.0000	1
2	2016-09-02 17:44:54.932000	2016-09-07 00:08:54.932000	60.0000	6144
3	2016-09-07 01:06:54.932000	2016-09-12 13:05:54.932000	60.0000	7919
4	2016-09-12 13:05:54.932000	2016-09-12 13:06:17.752000	22.8200	1
5	2016-09-12 13:27:43.376000	2016-09-17 16:38:43.376000	60.0000	7391
6	2016-09-17 16:38:43.374000	2016-09-17 16:39:43.374000	60.0000	1
7	2016-09-17 16:39:43.373000	2016-09-17 16:41:43.373000	60.0000	2
8	2016-09-17 16:41:43.372000	2016-09-17 16:42:43.372000	60.0000	1
9	2016-09-17 16:42:43.374000	2016-09-17 16:43:43.374000	60.0000	1

Figure 2: First nine entries of the SQL database for passive data management. [NR] siyuan1/. database

EVENT CATALOG AND INTERFACES

The essential idea of building the event catalog was that for each event around the Bay area since September 2016, we locate the passive data using the *regFileSetLog.sqlite* SQL database and python tools described in the previous section, preprocess and save the data; and then record the data file paths and file names along with the metadata in our event SQL database *event_log.sqlite*. We also created interfaces for the users to efficiently select the events of interest and extract the event data.

To start with a reliable set of events, metadata are pulled from the USGS online database. These metadata include the event start time, magnitude, and location of the epicenter. We calculate the three-dimensional (3D) distance from the epicenter to our array for each event. With the start time and estimated time range (proportional to the distance to our array) for each event, and taking advantage of data managing tools in the previous section, we locate the raw data on our storage system. The original data are proportional to strain. We take their temporal derivative to convert them to strain rate, which has a broader spectrum, useful for visualizing events (Martin et al., 2017). We then high-pass the data by setting the low-cut frequency to be $8/(\text{event time range})$, considering that the closer events are to us the greater proportion high-frequency components arise in the spectrum. Last, but not least, at each time slice we calculate the median data rate and subtract the median value from the data rate recorded by each channel to compensate for possible laser drift occurring on all channels at once. We save the preprocessed data into both a SEG-Y file and a SEPLib file format organized into a `/data/biondo/DAS/EventCatalog/-Data/year/month/day` folder. The file names and data paths were recorded in our event catalog SQL database. We have cataloged more than 800 events ranging from September 2016 to August 2017.

The interface of the event catalog takes time range, magnitude range, epicenter depth range, distance to SDASA-1 range, and azimuth angle to SDASA-1 range as inputs to filter the events and return the corresponding metadata, data paths, and file names for both of the SEG-Y and SEPLib data files to users. Users can choose to either set every input option or choose any subset of the options of particular interest. They may save a limited amount of data to their local directories for further analysis. In the next section, we show how we used our catalog for signal repeatability analyses.

SIGNAL REPEATABILITY ANALYSIS

Because weak events tend to be contaminated by local noise, especially during daytime, we are always interested in whether or not the data recorded are from the actual events. A nearby quarry blast (around 13km from SDASA-1) and some frequently occurring nearby earthquakes provided us with an opportunity to assess signal repeatability of the arrivals. To validate the events observed in DAS data, we used the data recorded at the Jasper Ridge Seismic Station (JRSC station) by a broadband seismometer managed by the Berkeley Digital Seismic Network. Data from JRSC are available online. The Jasper Ridge station is located approximately 6.4 km from our DAS array. Because near-surface conditions are different below our array and JRSC and ray paths are different, the waveforms were not directly comparable. However, JRSC data provided a rough indication of the arrival time and relative strength of the signal corresponding to different arrivals (i.e., P-waves, S-waves, and surface waves).

Permanente Quarry Blasts

Biondi et al. (2017) show the repeatable patterns from two quarry blasts. With the earthquake catalog tools presented in the last section, we quickly pulled quarry blast data by setting the geographical region (latitude range and longitude range around the quarry blast) through the catalog interface. We identified a total of 67 quarry blasts recorded by SDASA-

1. The maximum distance, by which these events were separated is less than 800 meters, which means they are good candidates for waveform repeatability assessment. Indeed we did see satisfactory repeatability between a number of events. However, some events, especially weak ones, gave us no recognizable patterns. Here we first present our analyses of six representative events showing clear repeatability. The following list provides the main information for the six events in order of increasing magnitude. The magnitude (M) and depth (z) of each event are based on the online USGS database. Distance (Δ) to DAS array of each event is calculated based on the 3D distance between SDASA-1 and the epicenter.

- *Blast#1*: September 23, 2016 – M 1.47 – z=-0.31 km – Δ =13.43 km
- *Blast#2*: September 29, 2016 – M1.53 – z=-0.31 km – Δ =13.17 km
- *Blast#3*: November 28, 2016 – M1.64 – z=-0.22 km – Δ =13.38 km
- *Blast#4*: October 12, 2016 – M1.68 – z=-0.28 km – Δ =13.21 km
- *Blast#5*: May 23, 2017 – M1.73 – z=-0.31 km – Δ =13.46 km
- *Blast#6*: February 09, 2017 – M1.87 – z=-0.26 km – Δ =13.51km

To give the reader a sense of these SDASA-1 event records, the top panels of Figure 3 and Figure 4 show the data for *Blast#1* and *Blast#2* respectively, after bandpassing from 0.23 to 2.0 Hz. The origin of the time axis (0 seconds) is the event start time provided by the USGS online database. We use the same convention for the following data displays. The trace at the bottom shows the data recorded by the vertical component of broadband data recorded by JRSC (JRSC-BH data). This trace was bandpassed with the same filter parameters as the DAS data. The two events have similar surface-wave kinematics. We observe that these two events have a repeatable surface-wave arrival at approximately 9 seconds. Furthermore, compared with JRSC-BH data, for these quarry blast events, SDASA-1 records more reliable data.

To make the interpretation more straightforward, for each of the events, we stacked the envelopes of the traces after normalization by maximum absolute value. Figure 5 shows the stacking results of these blasts on the same plot, with each color line corresponding to each blast. Clearly, their waveforms have much in common. All the waveforms have amplitude jumps at around 8 seconds, which we interpret as S-wave arrivals. They all have amplitude peaks at around 12 seconds, which can be explained as arrivals of surface waves with strong energy.

The following events (*Blast#7* to *Blast#10*), however, represent events with no evident regular patterns, as stacking results (traces stacked in the same way as *Blast#1* to *Blast#6*) shown in Figure 6. The reason may be that these events were over-whelmed by locally-generated noise.

- *Blast#7*: April 12, 2017 – M 1.52 – z=-0.31 km – Δ =13.49 km
- *Blast#8*: September 13, 2016 – M1.58 – z=-0.23 km – Δ =13.47 km
- *Blast#9*: May 04, 2017 – M1.65 – z=-0.31 km – Δ =13.49 km

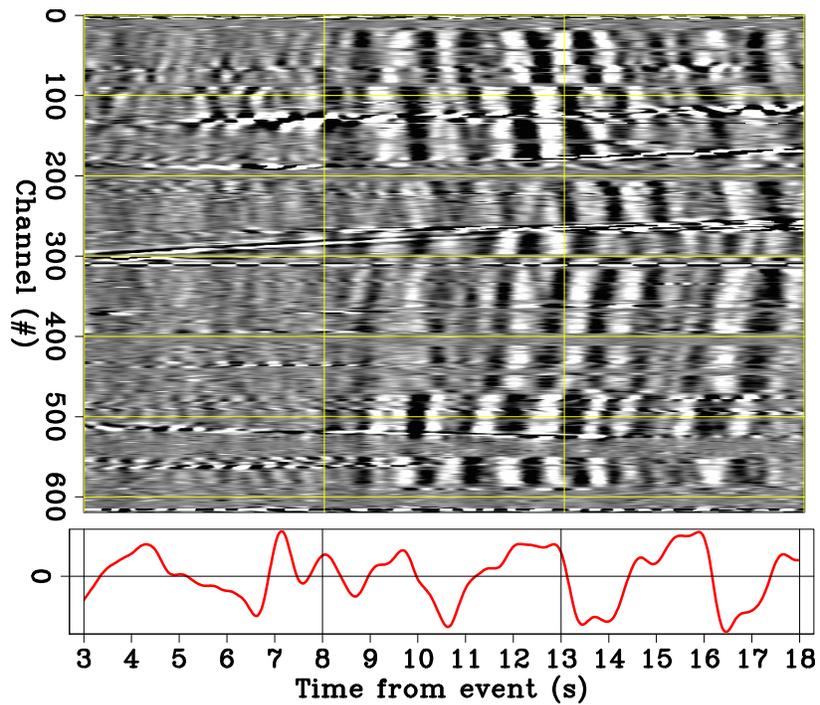


Figure 3: *Blast#1* Data bandpassing (0.23-2.0 Hz). Top: DAS array data; Bottom: JRSC-BH vertical component. The time origin (0 seconds) is the event time according to the USGS and channel numbers correspond to the markings on Figure 1. [CR]

siyuan1/. blast1-DAS-JR

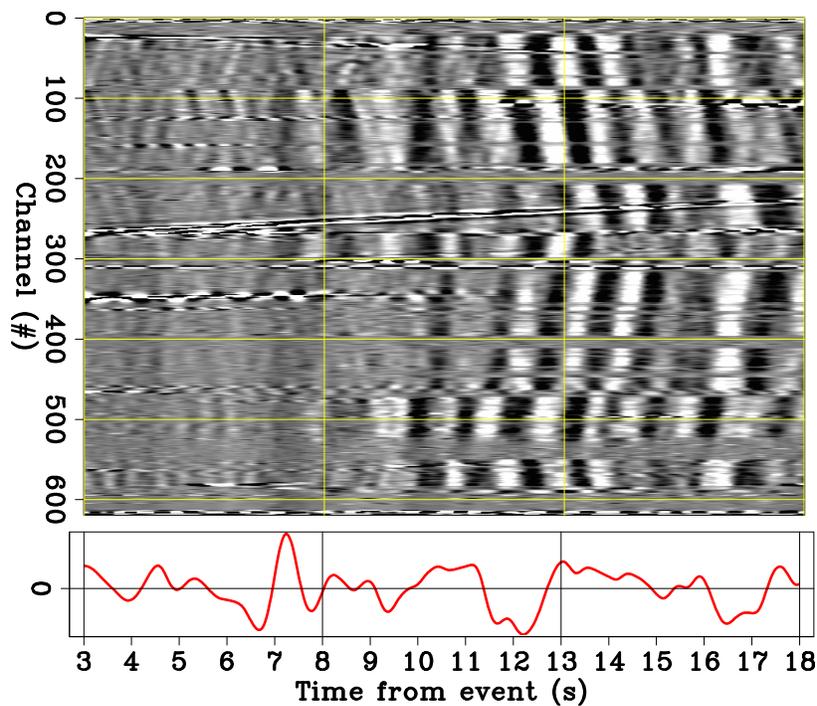


Figure 4: *Blast#2* Data after bandpassing (0.23-2.0 Hz). Top: SDASA-1 array data; Bottom: JRSC-BH vertical component. The time origin and channel numbers follow the same convention as Figure 3 [CR] `siyuan1/. blast2-DAS-JR`

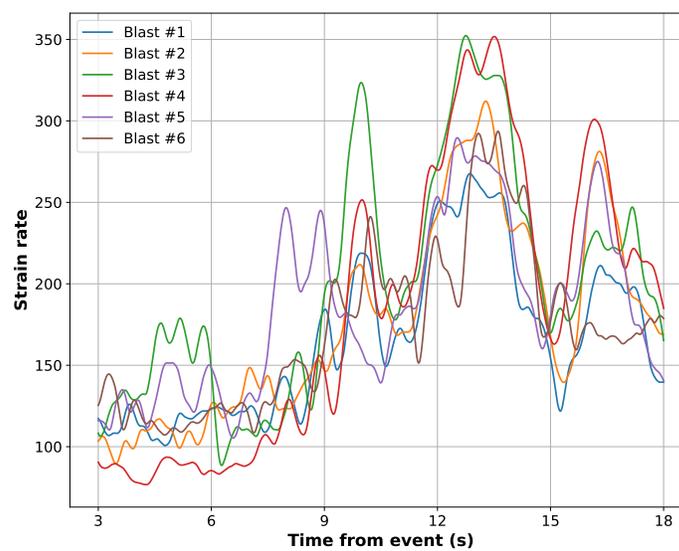


Figure 5: Stacks of the envelopes of the normalized traces for *Blast#1* to *Blast#6* after bandpassing (0.23-2 Hz). The time-axis origin (0 seconds) is the time of the event according to the USGS online database. [CR] `siyuan1/. QB-Stack-env-good`

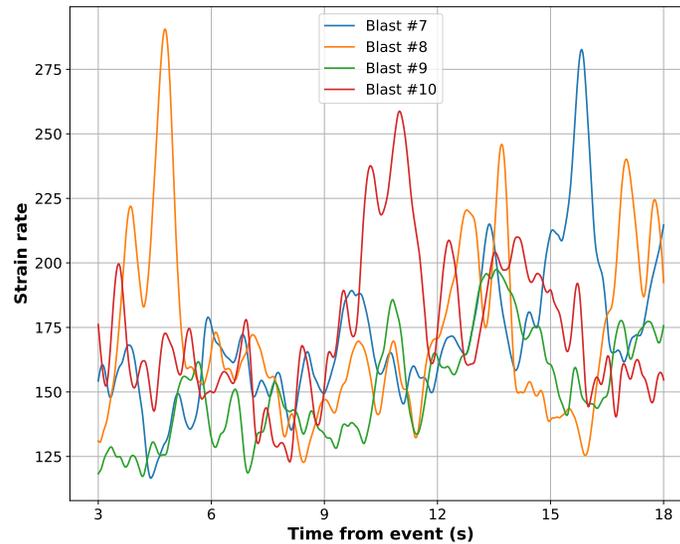


Figure 6: Stacks of the envelopes of the normalized traces for less clear events, *Blast#7* to *Blast#10* blasts after bandpassing (0.23-2.0 Hz). The time-axis origin (0 seconds) is the time of the event according to the USGS online database. [CR] siyuan1/. QB-Stack-env-bad

- *Blast#10*: December 13, 2016 – M1.66 – $z=-0.28$ km – $\Delta=13.56$ km

Besides artificial events, we pulled natural earthquake recordings using our catalog interfaces. We observed satisfactorily repeatable signals for nearby events in terms of the arrival times of P-waves, S-waves, and surface-waves. We show the analysis for two earthquakes near Ladera, CA and two earthquakes at Felt Lake near the Stanford campus in the next two subsections.

Ladera earthquakes

The main information for the two Ladera events is listed in the following:

- *Lad#1*: August 10, 2017 – M 1.63 – $z=+3.63$ km – $\Delta=4.62$ km
- *Lad#2*: August 10, 2017 – M 1.79 – $z=+3.25$ km – $\Delta=4.62$ km

These two events are very close to each other both in time (26 minutes apart) and in location (3D distance 456-meters apart). In addition, they were very close to Stanford, and both happened at night (local time), which means there should be less anthropogenic noise near the DAS array. The top panel of Figure 7 and Figure 8 show the data bandpassed between 0.25 Hz and 20 Hz for the two events, respectively. The bottom panels show the vertical component of JRSC-BH for comparison. We did see great repeatabilities for arrival timings. Although there is a little time shift, both events have the P-wave arriving at

approximately 1.5 seconds, and mix of S-wave and surface-wave arrivals at approximately 3.0 seconds.

We scaled both events in terms of the maximum value for each trace. Then we trace-wise cross-correlated the scaled data of these two events from 1.4 seconds to 5 seconds after their USGS-picked start times, and stacked the results, shown in Figure 9. The peak value is at 0.078 seconds. We advanced the scaled data of *Lad#1* by 0.078 seconds, as shown in Figure 10. Figure 11 shows the scaled data of *Lad#2*. The difference between the two events is shown in Figure 12. We can see that significant cancellations between these events' waveforms happen after P wave arrives (around 1.5 seconds) showing great repeatability. Figure 13 shows the Fast Fourier Transform (FFT) spectra of these two events' scaled data and of their difference. The Nyquist rate of SDASA-1 recording is 25 Hz. Thus, we cannot see higher frequency content without risk of aliasing. The spectra of the two events are consistent with each other, especially for frequencies below 15 Hz. Their difference's spectrum shifts towards high frequency content, which shows that lower frequencies are more repeatable.

Additionally, we estimated the energy for each event by summing up the squared amplitude of all traces ranging from 1.48 seconds to 4.5 seconds. We obtained an estimated energy ratio (*Lad#2* to *Lad#1*) of 1.15. For comparison, we calculated the energy ratio by $10^{1.79}/10^{1.63}$, where 1.79 is the magnitude of *Lad#2* and 1.63 is that of *Lad#1*. The result is 1.45. The difference between the two numbers (1.15 and 1.45) could be explained as a combination of higher levels of noise in our recordings and the fact that we do not record complete three component data.

Felt Lake earthquakes

Repeatability is not just observable at night. Here we show the recording of an earthquake doublet occurring on Stanford's campus near Felt Lake, 4.2 km from the DAS. The distance between their epicenters was estimated by USGS to be close to 100 m, and they occurred one minute apart just prior to 13:00, local time. Figure 14 shows the data recorded by our DAS array and the Jasper Ridge Seismic Station broadband seismometer corresponding to the first, and stronger (magnitude 1.34) of the Felt Lake events. The DAS data were preprocessed via laser-noise attenuation, trace balancing, and bandpassing from 0.25 to 12.0 Hz. Figure 15 shows the recording corresponding to the second, and weakest (magnitude 0.95) of the Felt Lake events. The same preprocessing was applied as for the data showed in the previous figures.

- *Felt#1*: July 12, 2017 – M 1.34 – $z=+3.24$ km – $\Delta=5.45$ km
- *Felt#2*: July 12, 2017 – M 0.95 – $z=+3.05$ km – $\Delta=5.34$ km

The DAS data shown in Figure 14 and Figure 15 show strong and repeatable waveforms across the whole array starting at about 3.7 seconds. These are likely to be a mix of S-wave and surface-wave arrivals. The waveforms are complex because of the complexity of the near surface both close to the epicenter and in the vicinity of the DAS array. However, they stand out from the strong background noise. As expected, the signal-to-noise ratio is higher for the first stronger event than the second weaker event. There is no clear P-wave

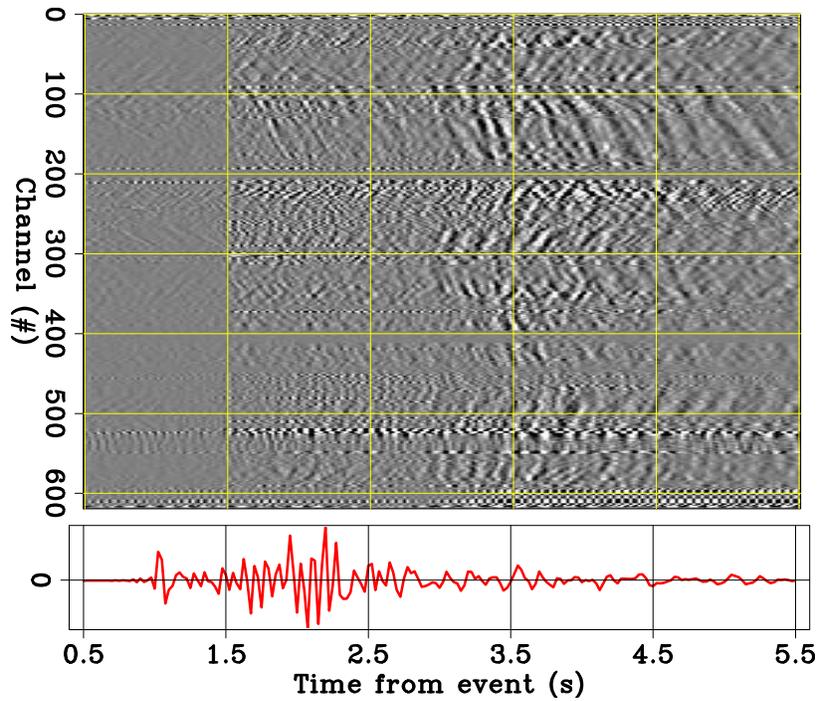


Figure 7: *Lad#1* Data bandpassing (0.25-20 Hz). Top: DAS array data; Bottom: JRSC-BH vertical component. The time origin (0 seconds) is the event time according to USGS. [CR] siyuan1/. ladera1-DAS-JR

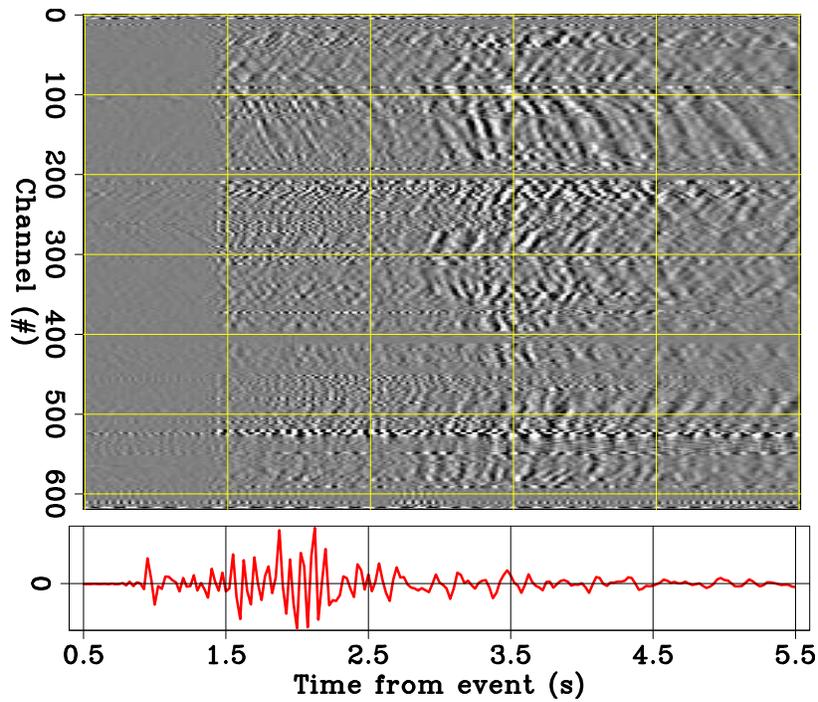


Figure 8: *Lad#2* Data bandpassing (0.25-20 Hz). Top: DAS array data; Bottom: JRSC-BH vertical component. The time origin (0 seconds) is the event time according to USGS. [CR] siyuan1/. ladera2-DAS-JR

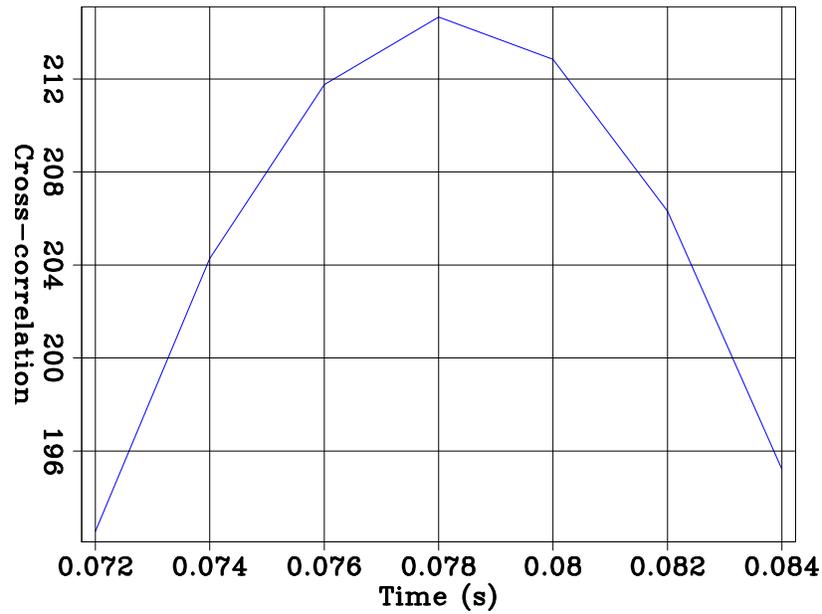


Figure 9: Stack of the tracewise cross-correlations between the bandpassing (0.25-20 Hz) data of *Lad#1* and *Lad#2*. [CR] `siyuan1/. ladera-cc`

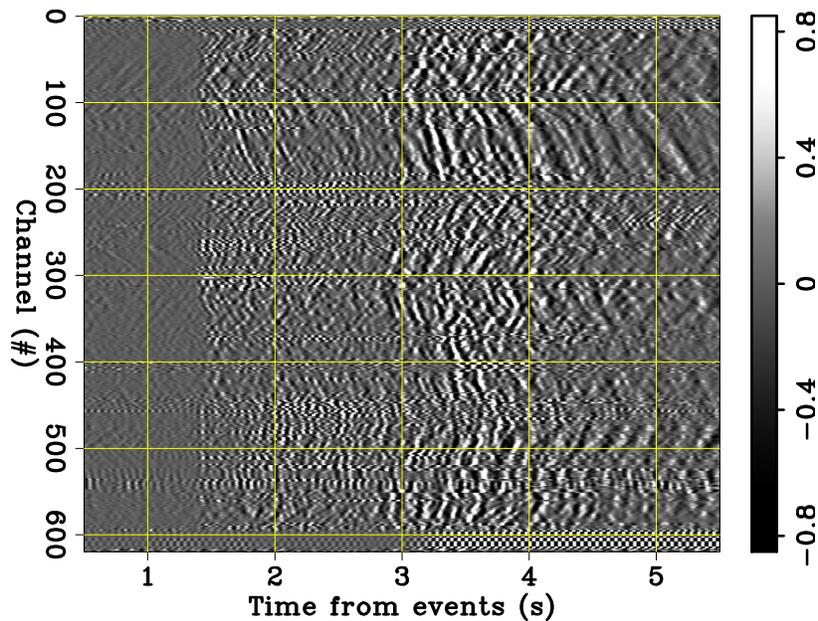


Figure 10: *Lad#1* data after bandpassing (0.25-20 Hz), scaling each trace by its maximum value, and shifting forward by 0.078 seconds. The time origin (0 seconds) is 0.078 seconds from *Lad#1* start time based on USGS. [CR] `siyuan1/. SDASA-ladera1-scaled`

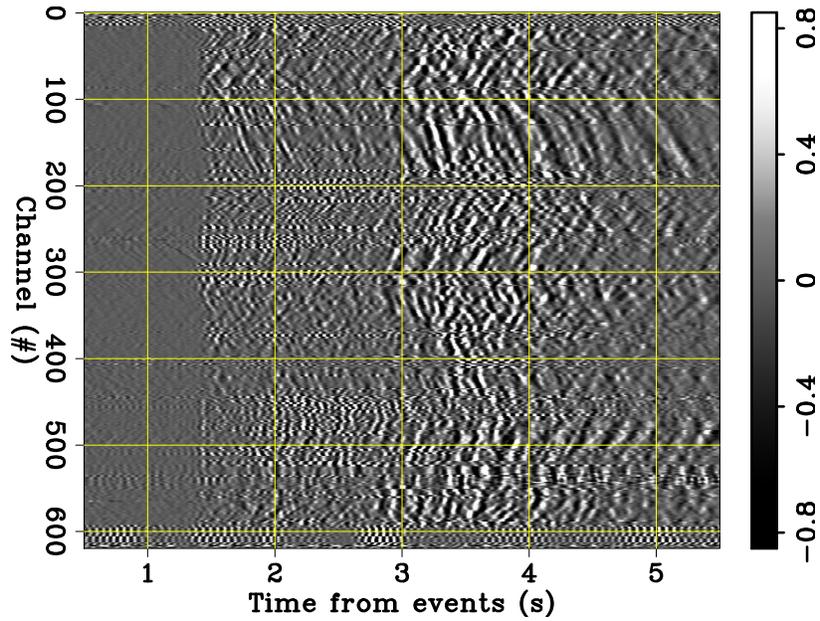


Figure 11: *Lad#2* data after bandpassing (0.25-20 Hz), and scaling each trace by its maximum value. The time origin and channel numbers follow the same convention as Figure 3. [CR] `siyuan1/. SDASA-ladera2-scaled`

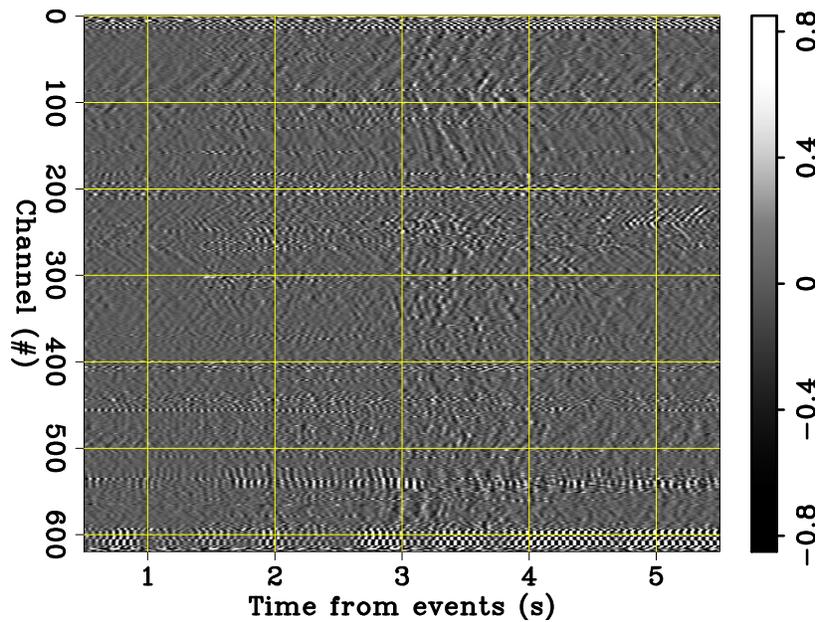


Figure 12: Difference between the trace-wise scaled data of *Lad#1* (shifted forward by 0.078 seconds) and *Lad#2*. The time-axis origin is the time of the event according to USGS's online database. [CR] `siyuan1/. ladera-diff`

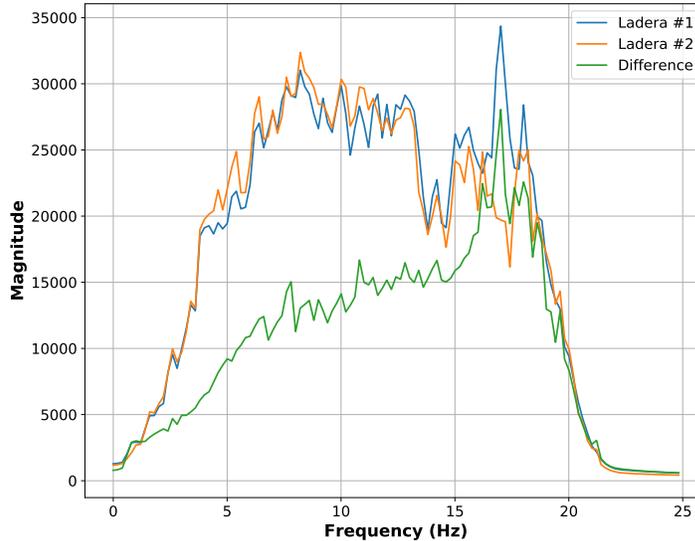


Figure 13: FFT spectra of *Lad#1*, *Lad#2*, and their difference. [CR] `siyuan1/. ladera-spec`

arrival visible in either of the two DAS recordings. The P-wave arrived at Jasper Ridge after approximately 1.4 seconds, as is clearly observable from the vertical-component trace shown in Figure 14. The corresponding vertical-component trace in Figure 15 shows a weaker, but still identifiable, P-wave arrival.

The Felt Lake examples demonstrate the repeatability of signals recorded by the DAS array, but they do not show clearly identifiable P and S phases, due in large part to strong anthropogenic noise generated on campus.

CONCLUSION AND FUTURE WORK

We have created a continuously updated event catalog for the passive data recorded by our DAS array. We have created interfaces that can be used to filter, access, and process the event data efficiently. Taking advantage of the catalog tools, we examined signal repeatability on both man-made events (quarry blasts) and natural events (Ladera and Feltlake). We found strong signal repeatability in terms of surface-wave arrival timings for Permanente Quarry blasts. And we noticed clearly repeatable P-wave, S-wave, and surface-wave arrivals for naturally occurring nearby events from Ladera and Felt Lake. In the future, we will work on estimating what distances and magnitudes are clearly detectable by the DAS array. We will use the catalog to extract data for distant and weak events. On these events, we will test common array methods for earthquake detection and location, including beamforming and Short-Term Averaging/Long-Term Averaging (STA/LTA) analysis in time and frequency. Then, further analysis should be done on detectability with methods tailored to small events (e.g., template matching).

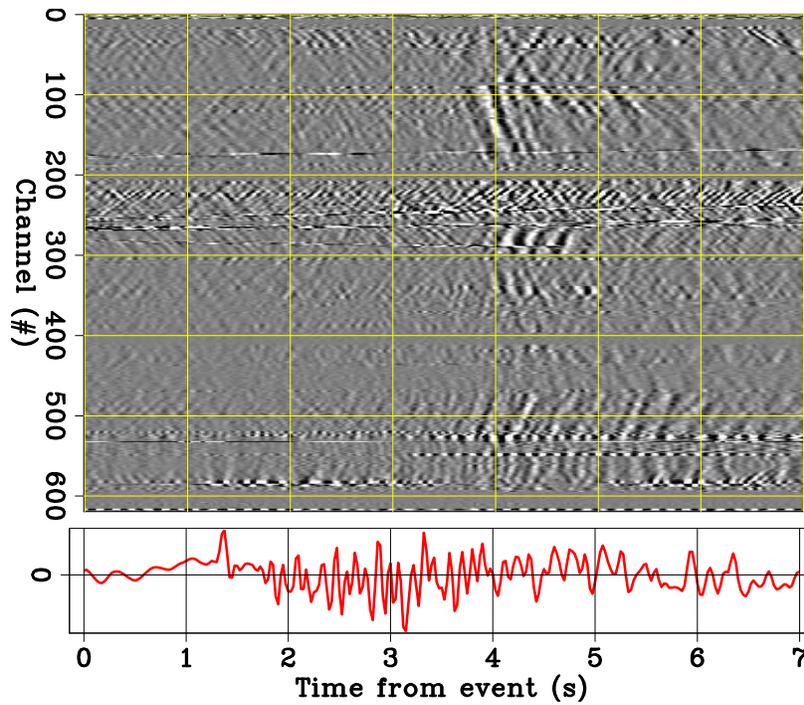


Figure 14: *Felt#1* Data bandpassing (0.23-12.0 Hz). Top: DAS array data; Bottom: JRSC-BH vertical component. The time origin (0 seconds) is the event time according to the USGS. [CR] `siyuan1/. feltLake1-DAS-JR`

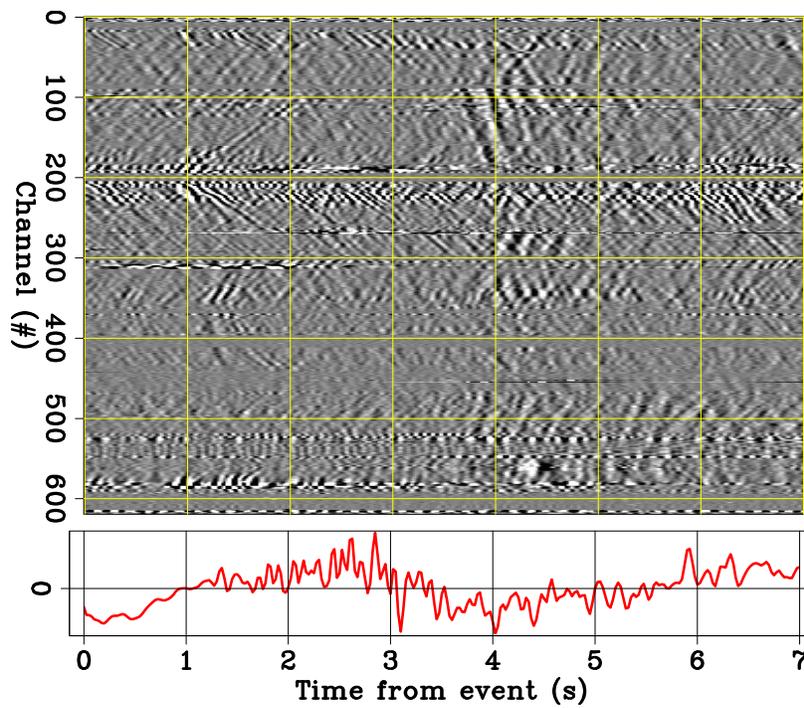


Figure 15: *Felt#2* Data bandpassing (0.23-12.0 Hz). Top: DAS array data; Bottom: JRSC-BH vertical component. The time origin (0 seconds) is the event time according to the USGS. [CR] `siyuan1/. feltLake2-DAS-JR`

ACKNOWLEDGEMENT

The interrogator unit was loaned to us by OptaSense, Inc. We thank S. Levin and S. Ronen for advice, F. Huot for sharing scripts pulling data from USGS. We thank Stanford IT and SEES IT for assistance running the array, and Stanford Center for Computational Earth and Environmental Sciences for computing resources. E. Martin has also been supported in part by the DOE CSGF under grant number DE-FG02-97ER25308, and a Schlumberger Innovation Fellowship.

REFERENCES

- Biondi, B., E. Martin, S. Cole, and M. Karrenbach, 2017, Earthquakes analysis using data recorded by the Stanford DAS Array: SEP-Report, **168**, 11–26.
- Martin, E., B. Biondi, S. Cole, and M. Karrenbach, 2017, Overview of the Stanford DAS Array-1 (SDASA-1): SEP-Report, **168**, 1–10.

Modified Tomographic Full Waveform Inversion using the variable projection method

Guillaume Barnier, Ettore Biondi, and Biondo Biondi

ABSTRACT

We propose a modified tomographic full waveform inversion (TFWI) optimization scheme that allows us to avoid the original nested-loop approach and reduce the number of inversion parameters. We use the variable projection method to solve for the linear component of the inverse problem. We show the convergence to the correct velocity model on a synthetic dataset lacking frequencies below 10 Hz.

INTRODUCTION

Almomin (2016a) showed the potential of TFWI in simultaneously recovering all model scale components while being immune to cycle-skipping. However, this technique is hampered by three major challenges. Global convergence has not yet been mathematically proved even though it has been heuristically observed. Moreover, it heavily relies on the user to tune the many inversion parameters present in the original nested-scheme approach. Finally, it is highly computational intensive. Therefore, this current formulation is hardly applicable on production field datasets.

To overcome the first and second issues, we propose a new formulation in which the extended modeling term simply ensures phase matching between observed and predicted data. During our optimization process, we reduce the contribution of the additional extended term over iterations while ensuring the convergence of the total objective function. Since our cost function is quadratic with respect to the extended component, we use the variable projection method to minimize it (Rickett, 2013; Huang and Symes, 2015). We also demonstrate the potential of our new formulation on a 2D synthetic example in which the low frequency content is removed.

We first describe the mathematical difference between our proposed algorithm and the original nested-loop formulation in Biondi and Almomin (2014). We then apply our technique on a 2D synthetic model similar to the one shown in Mora (1989) and compare the results to the ones obtained using FWI.

TFWI THEORY

We review the original TFWI formulation proposed by Biondi and Almomin (2014), and analyze the potential reason for its slow convergence rate. Then we describe our proposed method to potentially overcome this issue.

Original formulation

In the TFWI optimization algorithm devised by Biondi and Almomin (2014), the minimized objective function is effectively the standard FWI data misfit,

$$\Phi_{FWI}(\mathbf{m}) = \frac{1}{2} \|\mathbf{f}(\mathbf{m}) - \mathbf{d}^{\text{obs}}\|_2^2, \quad (1)$$

where \mathbf{f} is the wave-equation operator, \mathbf{m} is the velocity model, and \mathbf{d}^{obs} represents the observed data. The optimization is performed in a nested scheme, using an extended Born modeling operator, an auxiliary model composed of a background model \mathbf{b} , and an extended perturbation (i.e., reflectivity) $\tilde{\mathbf{p}}$. This nested scheme is designed to protect the algorithm from cycle-skipping encountered in standard FWI workflows, especially when low frequencies are missing in the data.

To understand why the original scheme may suffer from slow convergence rate, we propose to follow step by step one full iteration of the “outer” loop, say from i to $i + 1$ (equation 1). That is, to compute the model update from $\mathbf{m}^i \mapsto \mathbf{m}^{i+1} = \mathbf{m}^i + \Delta\mathbf{m}^i$. In our notation, the upper script indices correspond to the outer loop iteration number, while the subscript indices correspond to the inner loop iteration number. In order to find the model update $\Delta\mathbf{m}^i$ needed at iteration i , the following objective function is minimized,

$$\begin{aligned} \Phi_{TFWI}^i(\mathbf{b}, \tilde{\mathbf{p}}) &= \frac{1}{2} \|\mathbf{f}(\mathbf{m}^i) + \tilde{\mathbf{B}}(\mathbf{b})\tilde{\mathbf{p}} - \mathbf{d}^{\text{obs}}\|_2^2 + \frac{\epsilon^2}{2} \|\mathbf{g}(\tilde{\mathbf{p}})\|_2^2 \\ &= \frac{1}{2} \|\tilde{\mathbf{B}}(\mathbf{b})\tilde{\mathbf{p}} - \Delta\mathbf{d}_i\|_2^2 + \frac{\epsilon^2}{2} \|\mathbf{g}(\tilde{\mathbf{p}})\|_2^2, \end{aligned} \quad (2)$$

where $\tilde{\mathbf{B}}$ is the extended Born modeling operator, and \mathbf{g} is an operator enhancing the non-physical energy of the the extended reflectivity. This last operator can be linear or non-linear with respect to the extended reflectivity. $\Delta\mathbf{d}_i = \mathbf{d}^{\text{obs}} - \mathbf{f}(\mathbf{m}^i)$ is kept constant during the minimization of Φ^i . This assumption might be the cause of the slow convergence rate of the original scheme. Note that Φ^i is non-quadratic with respect to the background \mathbf{b} , but quadratic with respect to the extended perturbation $\tilde{\mathbf{p}}$. Minimizing equation 2 is thus a “nonlinear” optimization problem. The inner inversion starts by setting $\mathbf{b}_0 = \mathbf{m}^i$ and $\tilde{\mathbf{p}}_0 = \mathbf{0}$. Minimizing equation 2 is performed using a gradient-based descent method. At each inner step, a scale mixing algorithm (wavenumber filtering) is applied to both gradients $\nabla_{\mathbf{b}}\Phi^i$ and $\nabla_{\tilde{\mathbf{p}}}\Phi^i$ (the non-extended component of $\nabla_{\tilde{\mathbf{p}}}\Phi^i$) to ensure that \mathbf{b} is only updated with low-wavenumber components, and that $\tilde{\mathbf{p}}$ is only updated with higher-wavenumber components. The wavenumber cut-off is based on the dominant frequency in the data as well as the average velocity of the initial model (Almomin, 2016). Let \mathbf{b}_{opt}^i and $\tilde{\mathbf{p}}_{opt}^i$ be the solutions found at the last inner iteration of outer loop i (i.e., after minimizing equation 2). A final low-wavenumber bandpass filter is applied to both \mathbf{b}_{opt}^i and \mathbf{p}_{opt}^i , where \mathbf{p}_{opt}^i is the non-extended component of $\tilde{\mathbf{p}}_{opt}^i$. The new model \mathbf{m}^{i+1} is given by

$$\begin{aligned}
\mathbf{m}^{i+1} &= \mathbf{m}^i + \Delta \mathbf{m}^i \\
&= \mathbf{m}^i + \mathbf{F} (\mathbf{b}_{opt}^i + \mathbf{p}_{opt}^i - \mathbf{m}^i) \\
&= \mathbf{F} (\mathbf{b}_{opt}^i + \mathbf{p}_{opt}^i) + (\mathbf{I} - \mathbf{F}) \mathbf{m}^i,
\end{aligned} \tag{3}$$

where \mathbf{F} is a low-pass filter in the wavenumber domain, and \mathbf{I} is the identity operator. So far, at the end of outer loop i that we just described above, we found a pair of variables \mathbf{b}_{opt}^i and $\tilde{\mathbf{p}}_{opt}^i$ such that

$$\mathbf{f}(\mathbf{m}^i) + \tilde{\mathbf{B}}(\mathbf{b}_{opt}^i) \tilde{\mathbf{p}}_{opt}^i \approx \mathbf{d}^{obs}, \tag{4}$$

which corresponds to minimizing the data fitting term of equation 2 with the constraint that most of the non-physical energy of $\tilde{\mathbf{p}}_{opt}^i$ has been reduced to zero, which in turn means that $\tilde{\mathbf{p}}_{opt}^i$ is not extended (i.e., $\tilde{\mathbf{p}}_{opt}^i \approx \mathbf{p}_{opt}^i$). If we also assume that $\mathbf{d}^{obs} = \mathbf{f}(\mathbf{m}_{true})$, equation 4 becomes

$$\mathbf{f}(\mathbf{m}^i) + \mathbf{B}(\mathbf{b}_{opt}^i) \mathbf{p}_{opt}^i \approx \mathbf{f}(\mathbf{m}_{true}). \tag{5}$$

Even though this updating scheme has been shown to work heuristically, there is no mathematical justification to update \mathbf{m}^i using equation 3. Equation 5 can be seen as a first-order Taylor expansion neither about \mathbf{m}^i nor \mathbf{b}_{opt}^i since they are different. However, if we had allowed the first term of equation 5 to vary as \mathbf{b}^i , the Taylor expansion could be justified (and therefore using equation 3 to update \mathbf{m}^i would seem reasonable) as long as \mathbf{p}_{opt}^i is “small” enough. This stepping method does not ensure that the new model decreases the FWI objective function (equation 1). This phenomenon has been observed in Almomin (2016b).

Modified formulation

In our formulation we propose to minimize the following objective function

$$\Phi(\mathbf{m}, \tilde{\mathbf{p}}) = \frac{1}{2} \|\mathbf{f}(\mathbf{m}) + \tilde{\mathbf{B}}(\mathbf{m}) \tilde{\mathbf{p}} - \mathbf{d}^{obs}\|_2^2 + \frac{\epsilon^2}{2} \|\mathbf{D} \tilde{\mathbf{p}}\|_2^2, \tag{6}$$

where \mathbf{D} is a linear operator with respect to $\tilde{\mathbf{p}}$ that enhances the non-physical extended energy of $\tilde{\mathbf{p}}$. This equation differs from equation 2 because $\mathbf{f}(\mathbf{m})$ is not kept constant and $\mathbf{g} = \mathbf{D}$ is linear with respect to $\tilde{\mathbf{p}}$. Since Φ is quadratic with respect to $\tilde{\mathbf{p}}$, we use the variable projection method to solve equation 6 (Golub and Pereyra, 1973; Rickett, 2013; Huang and Symes, 2015), which corresponds to minimizing the following objective function

$$\Phi(\mathbf{m}) = \frac{1}{2} \|\mathbf{f}(\mathbf{m}) + \tilde{\mathbf{B}}(\mathbf{m}) \tilde{\mathbf{p}}_{opt}(\mathbf{m}) - \mathbf{d}^{obs}\|_2^2 + \frac{\epsilon^2}{2} \|\mathbf{D} \tilde{\mathbf{p}}_{opt}(\mathbf{m})\|_2^2, \tag{7}$$

where $\tilde{\mathbf{p}}_{opt}$ is an extended perturbation model, defined as the minimizer of the following objective function $\Phi_{\mathbf{m}}$,

$$\Phi_{\mathbf{m}}(\tilde{\mathbf{p}}) = \frac{1}{2} \|\tilde{\mathbf{B}}(\mathbf{m})\tilde{\mathbf{p}} - (\mathbf{d}^{obs} - \mathbf{f}(\mathbf{m}))\|_2^2 + \frac{\epsilon^2}{2} \|\mathbf{D}\tilde{\mathbf{p}}\|_2^2. \quad (8)$$

For a fixed \mathbf{m} , $\Phi_{\mathbf{m}}$ reaches its minimum for

$$\tilde{\mathbf{p}}_{opt}(\mathbf{m}) = [\tilde{\mathbf{B}}^*(\mathbf{m})\tilde{\mathbf{B}}(\mathbf{m}) + \epsilon^2\mathbf{D}^*\mathbf{D}]^{-1}\tilde{\mathbf{B}}^*(\mathbf{m}) (\mathbf{d}^{obs} - \mathbf{f}(\mathbf{m})), \quad (9)$$

where $*$ denotes adjoint operators. Note that $\tilde{\mathbf{p}}_{opt}$ also depends nonlinearly on \mathbf{m} . The data residual component on the right side of equation 7 is a modified FWI objective function where an additional term is used to ensure the phase alignment between modeled and observed data. During the optimization process we slowly reduce the contribution of this additional term by adding a regularization term on the right side of equation 7. Therefore, finding the minimum of this equation is equivalent to minimizing Φ_{FWI} (equation 1).

Equation 8 can be solved by preconditioning the variable $\tilde{\mathbf{p}}$,

$$\tilde{\mathbf{q}} = \mathbf{D}\tilde{\mathbf{p}}, \quad (10)$$

which becomes

$$\Phi_{\mathbf{m}}^{prec}(\tilde{\mathbf{q}}) = \frac{1}{2} \|\tilde{\mathbf{B}}(\mathbf{m})\mathbf{E}\tilde{\mathbf{q}} - (\mathbf{d}^{obs} - \mathbf{f}(\mathbf{m}))\|_2^2 + \frac{\epsilon^2}{2} \|\tilde{\mathbf{q}}\|_2^2, \quad (11)$$

where $\mathbf{E} = \mathbf{D}^{-1}$. Solution of equation 11 is given by

$$\tilde{\mathbf{q}}_{opt}(\mathbf{m}) = [\mathbf{E}^*\tilde{\mathbf{B}}^*(\mathbf{m})\tilde{\mathbf{B}}(\mathbf{m})\mathbf{E} + \epsilon^2\mathbf{I}]^{-1}\mathbf{E}^*\tilde{\mathbf{B}}^*(\mathbf{m})(\mathbf{d}^{obs} - \mathbf{f}(\mathbf{m})), \quad (12)$$

where \mathbf{I} is the identity operator in the extended model space. We choose to precondition the problem since the operator \mathbf{D} is badly conditioned as shown by Clapp (2005)

RESULTS

We apply our proposed technique on a 2D synthetic model similar to the one described in Mora (1989). Figure 1 displays the true velocity model used in this test. We generate synthetic data with a Ricker wavelet containing energy from 10 Hz up to 25 Hz (Figure 2). We use 20 shots with 100 m spacing and we place receivers every 10 m. We apply our technique and compare the results to the ones obtained with FWI. For both inversions we inject the full bandwidth without using a multiscale approach (Bunks et al., 1995).

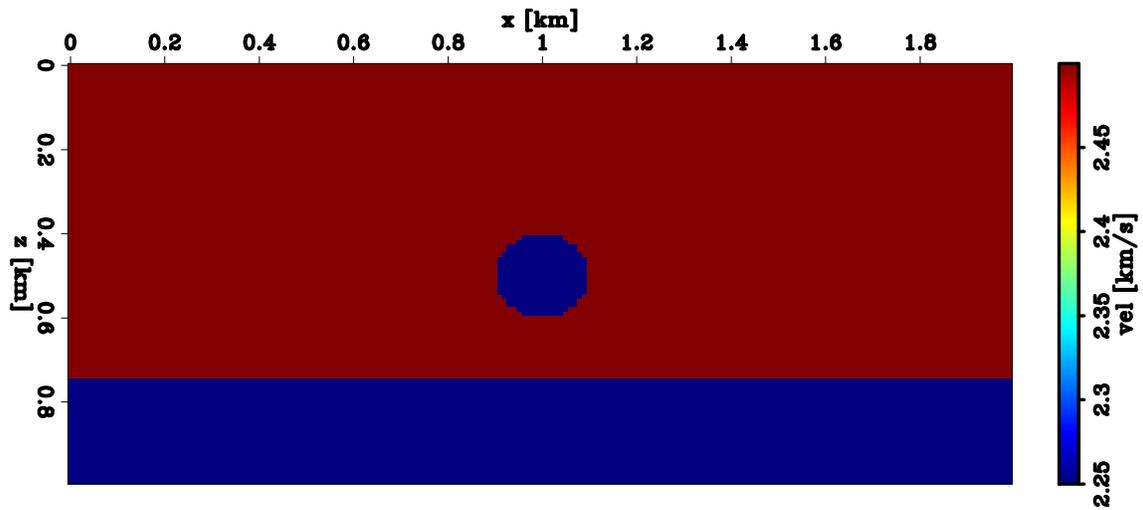


Figure 1: True velocity model similar to the one shown in Mora (1989) where an anomaly is embedded into a two layer subsurface. The velocity of the anomaly and of the second layer is 10% lower than the upper one. [ER] `guillaume1/. vel-true`

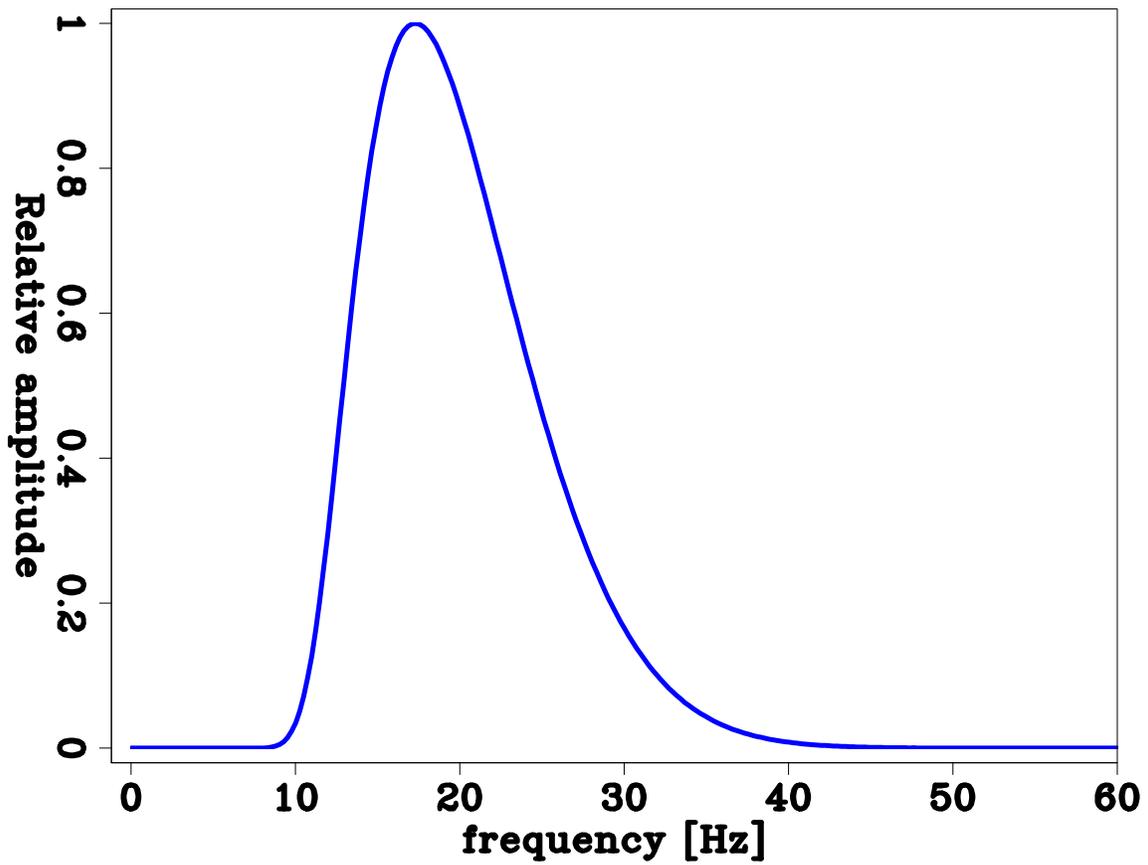


Figure 2: Spectrum of the used wavelet. [ER] `guillaume1/. spectrum`

The initial model is a constant background with a velocity of the top layer. For both inversions we use a Broyden-Fletcher-Goldfarb-Shanno (BFGS) optimization scheme (Liu and Nocedal, 1989). The linear problem solved during the TFWI optimization has been preconditioned as described in the previous section.

Figure 3 displays the FWI inverted model after 25, 100, 200 iterations. We notice that for the first 25 iterations mostly the reflectivity component of the model is retrieved by the inversion algorithm (Figure 3(a)). As we increase iteration number the algorithm is able to obtain the short-wavenumber component of the anomaly. Although most of the data are matched after 100 iterations (Figure 4), it is necessary additional 100 iterations to further improve the tomographic part of the model.

Figure 5 shows the TFWI inverted model after 5, 20, and 40 iterations. Because most of the reflection component of the data is matched by the optimal extended reflectivity, the modified TFWI algorithm inverts directly for the tomographic component of the model. Additionally, the reflection part of the model is correctly inverted. In fact, the reflector below the anomaly is present in the TFWI inverted model. As the algorithm progresses, the contribution of the extended reflectivity decreases (Figure 6). We observe the same inversion behavior as for the FWI result. In fact, despite that most of the data are matched after 20 iterations, the model is still changing significantly with additional 20 iterations.

To compare the FWI and TFWI inverted model we plot a vertical and a horizontal velocity profiles passing through the center of the anomaly along with the true model (Figure 7). Both inversion algorithms achieve similar results that in good agreement with the true velocity model. The ringing effects are caused by the limited bandwidth nature of the recorded data. In addition, the velocity contrast of the reflector below the anomaly is incorrectly placed due to the depth uncertainty present in the data.

To understand the advantage of preconditioning the linear inversion, we compare the convergence curves at the first non-linear iteration of the TFWI problem for preconditioned and un-preconditioned linear inversion (Figure 8). These curves show that the rate of convergence of the linear problem is greatly improved thanks to the preconditioning approach used.

CONCLUSIONS AND FUTURE WORK

We highlighted issues associated with the original implementation of the TFWI algorithm and proposed a new optimization scheme to potentially overcome these problems. In this new method we let the non-linear modeling operator vary during optimization. We also make use of the variable projection method to optimize the linear component independently of the non-linear one. Moreover, we showed the advantage of preconditioning the linear inversion to improve the convergence rate of the linear problem. On a model composed of two layers and a velocity anomaly we demonstrated the consistency of the proposed TFWI algorithm with FWI. Despite the lack of energy below 10 Hz in the generated data, FWI did not cycle skip in this case and both methods retrieved similar velocity models. Future work will involve the verification of global convergence on a more complicated model in which FWI fails to attain the global minimum. In addition, a complete comparison of different TFWI algorithms will be performed.

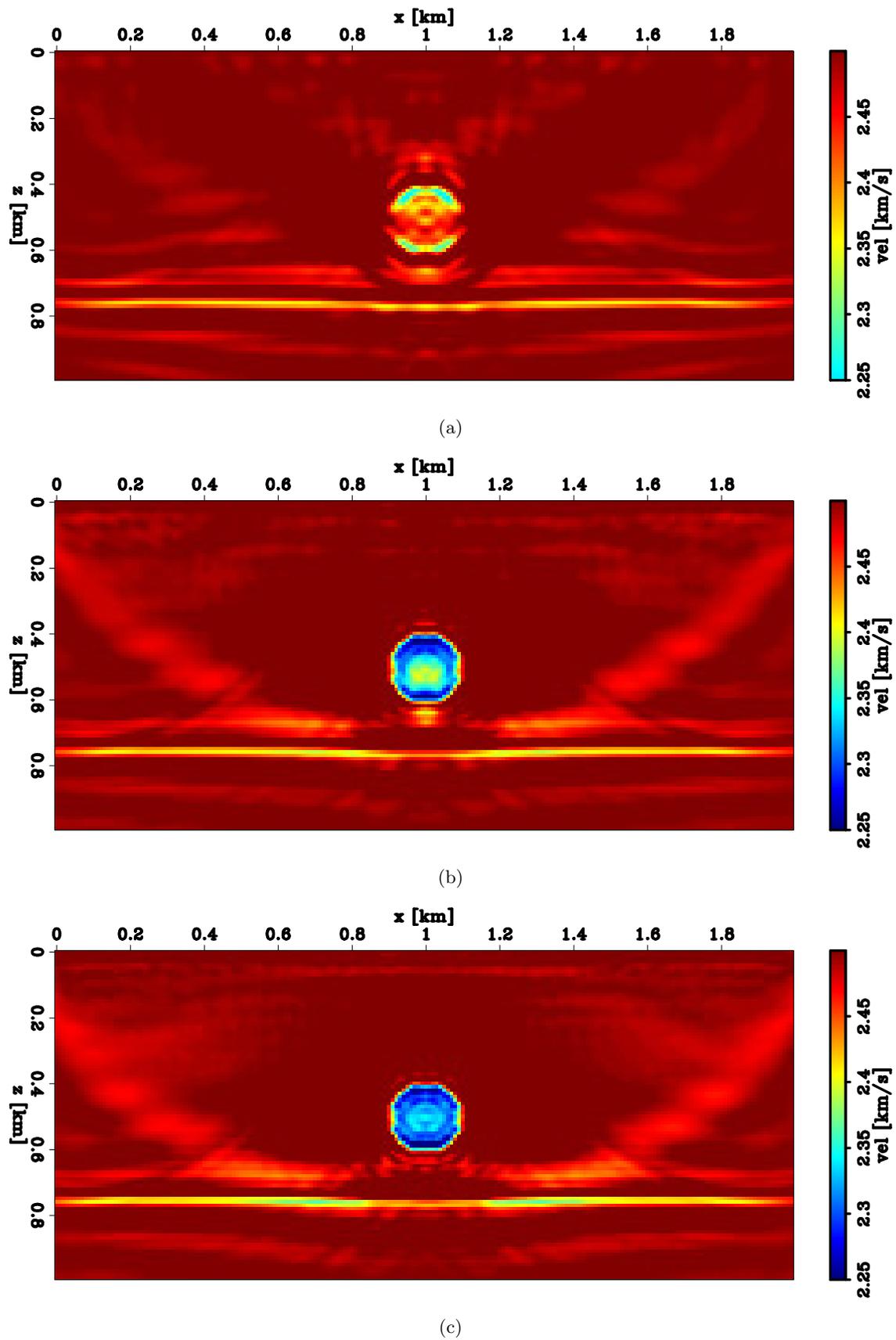


Figure 3: FWI results after (a) 25, (b) 100, and (c) 200 iterations, respectively. [CR] guillaume1/. fwi-it25-mod2d,fwi-it100-mod2d,fwi-it200-mod2d

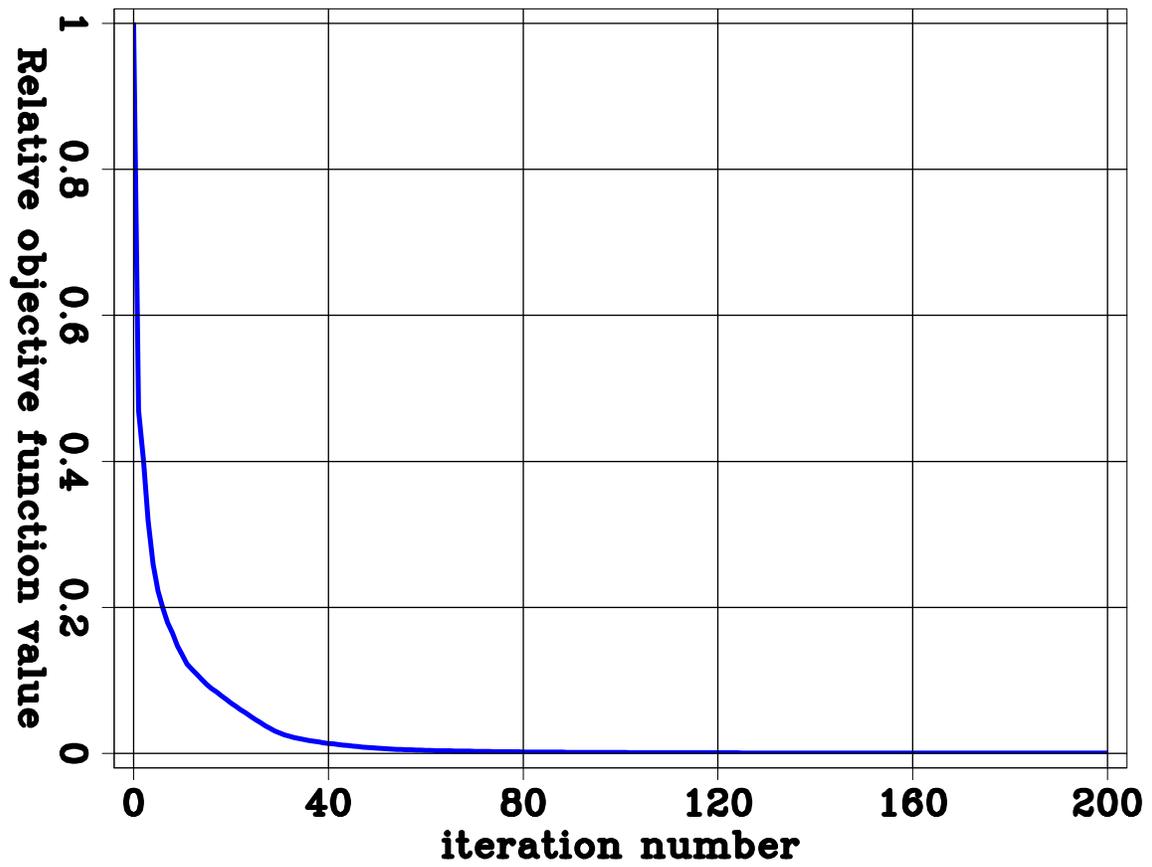


Figure 4: Relative FWI objective function (equation 7). [CR] `guillaume1/. obj-fwi`

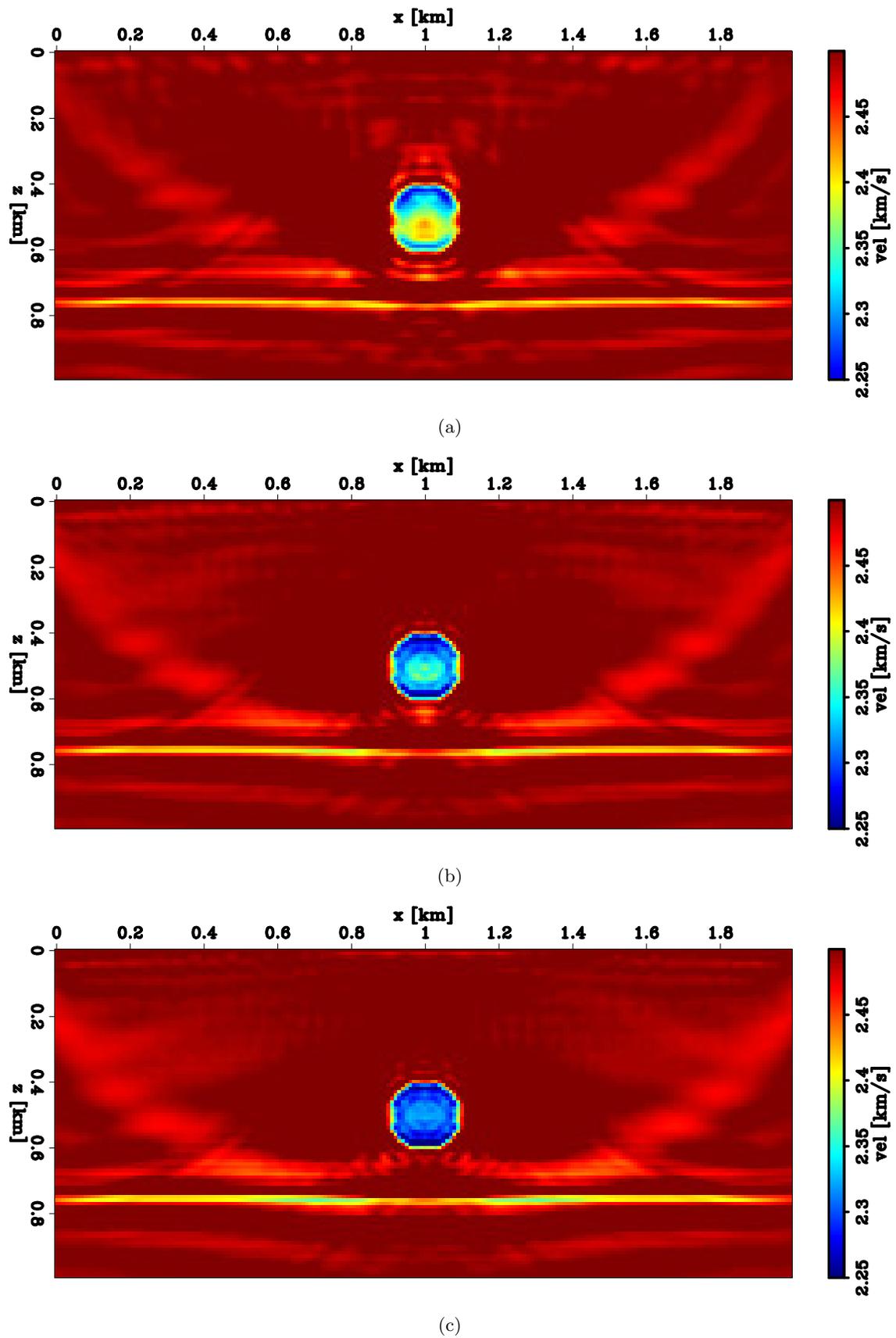


Figure 5: Modified TFWI results after (a) 5, (b) 20, and (c) 40 iterations, respectively.

[CR] [guillaume1/. vptfwi-it5-mod2d,vptfwi-it20-mod2d,vptfwi-it40-mod2d](#)

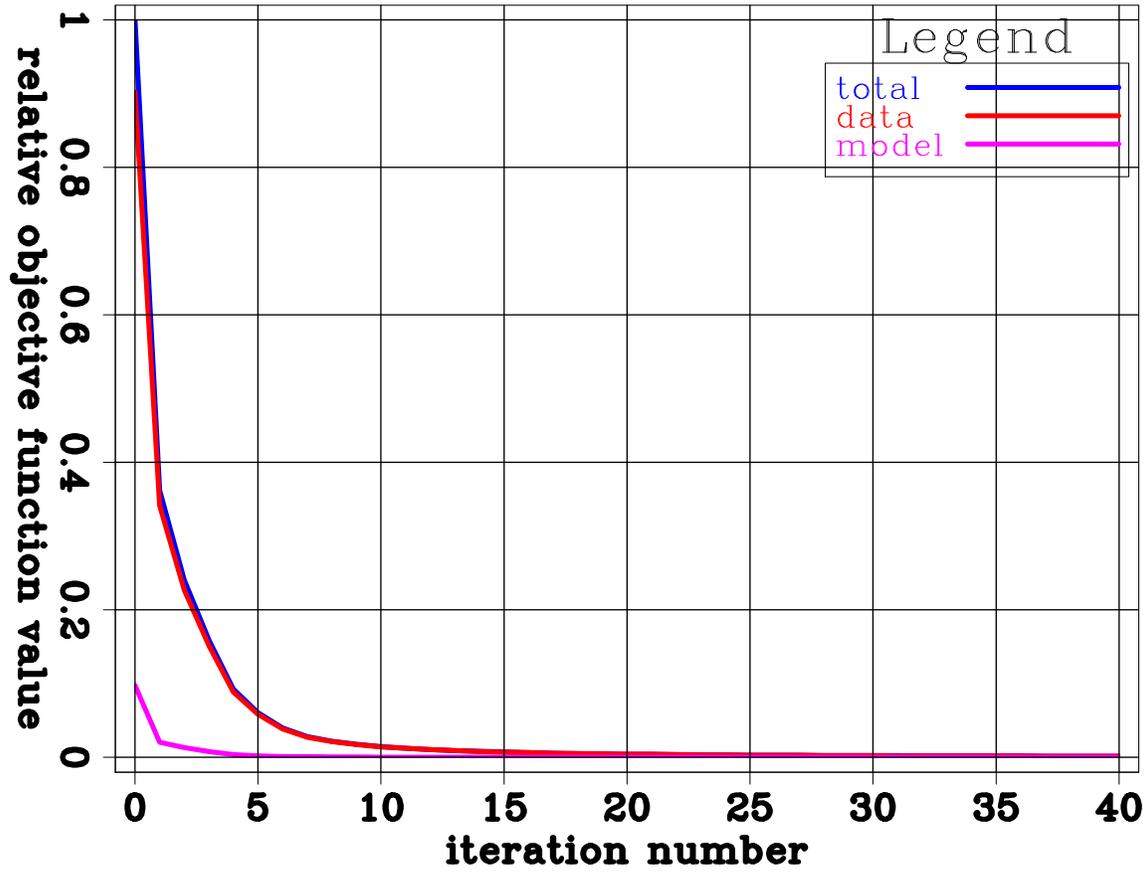


Figure 6: Relative TFWI objective function (equation 7). The total, data, and model objective functions are plotted. [CR] `guillaume1/. vptfwi-obj`

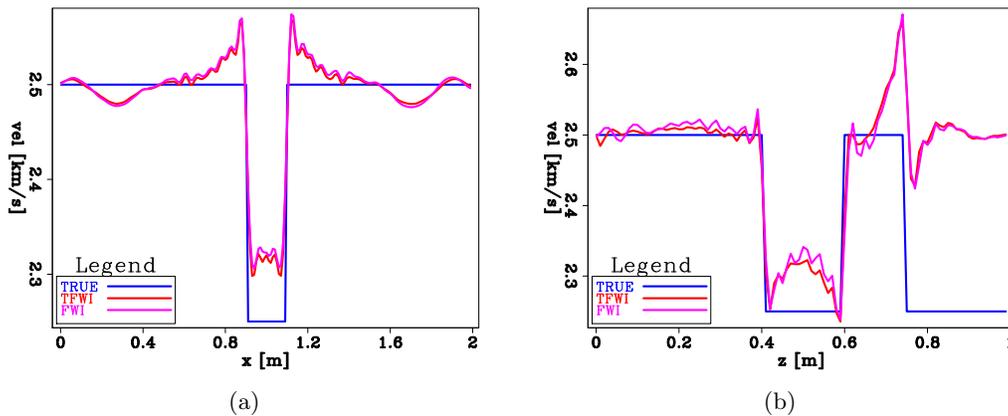


Figure 7: Horizontal and vertical velocity profiles of FWI and TFWI results at the latest iteration, and true model. Both profiles are passing through the center of the velocity anomaly. [CR] `guillaume1/. fwi-tfwi-compare1d-x,fwi-tfwi-compare1d-z`

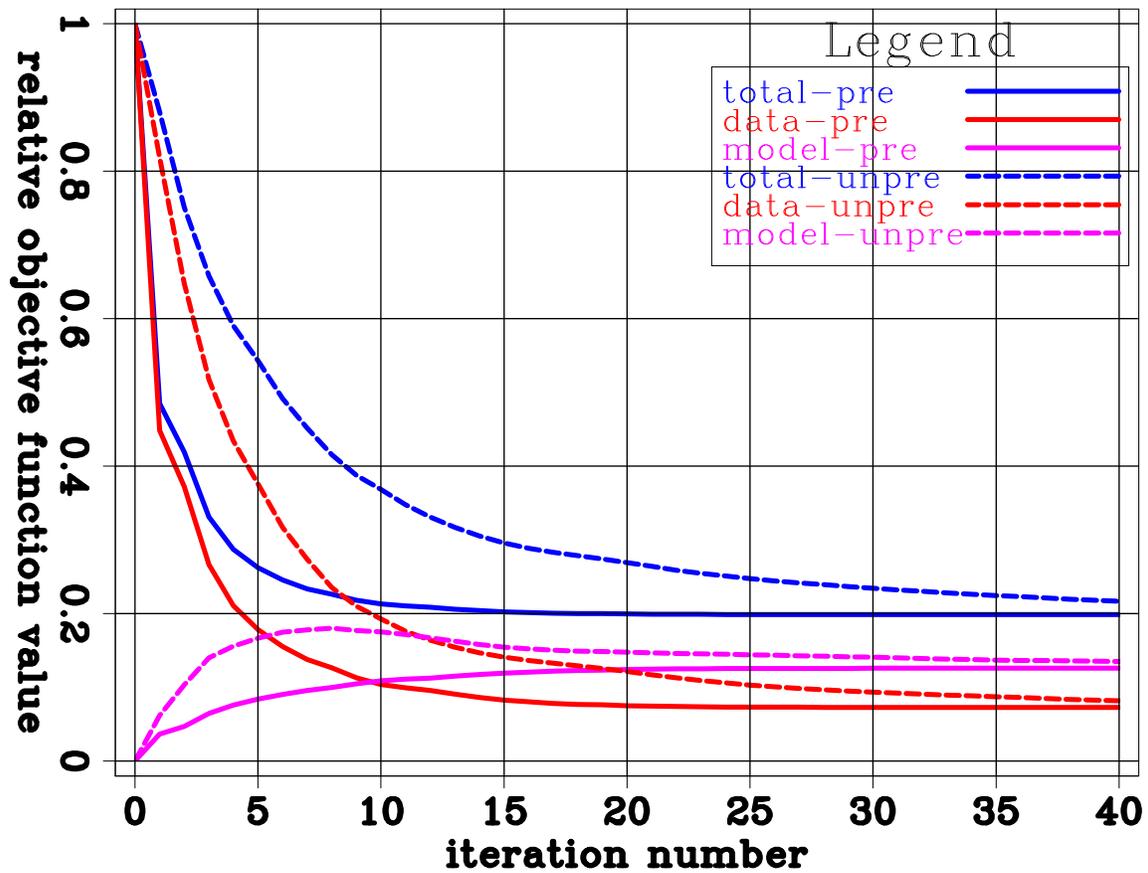


Figure 8: Convergence curves for preconditioned (solid curves) and un-preconditioned (dashed curves) linear inverse problem (equations 8 and 11) at the first non-linear iteration of the modified TFWI. [CR] [guillaume1/. compare-obj](#)

REFERENCES

- Almomin, A., 2016a, Tomographic full waveform inversion: Stanford University.
 ———, 2016b, Tomographic full waveform inversion: Presented at the 2016 SESAAI Annual Meeting.
- Biondi, B. and A. Almomin, 2014, Simultaneous inversion of full data bandwidth by tomographic full-waveform inversion: *Geophysics*, **79**, WA129–WA140.
- Bunks, C., F. M. Saleck, S. Zaleski, and G. Chavent, 1995, Multiscale seismic waveform inversion: *Geophysics*, **60**, 1457–1473.
- Clapp, M., 2005, Imaging under salt: illumination compensation by regularized inversion: Stanford University.
- Golub, G. H. and V. Pereyra, 1973, The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate: *SIAM Journal on numerical analysis*, **10**, 413–432.
- Huang, Y. and W. W. Symes, 2015, Born waveform inversion via variable projection and shot record model extension: SEG Technical Program Expanded Abstracts 2015, 1326–1331, Society of Exploration Geophysicists.
- Liu, D. C. and J. Nocedal, 1989, On the limited memory BFGS method for large scale optimization: *Mathematical programming*, **45**, 503–528.
- Mora, P., 1989, Inversion= migration+ tomography: *Geophysics*, **54**, 1575–1586.
- Rickett, J., 2013, The variable projection method for waveform inversion with an unknown source function: *Geophysical Prospecting*, **61**, 874–881.

APPENDIX

In this appendix we derive the gradient of equation 7. To compute the gradient of equation 7, we first define

$$\mathbf{r}_d(\mathbf{m}) = \mathbf{f}(\mathbf{m}) + \tilde{\mathbf{B}}(\mathbf{m})\tilde{\mathbf{p}}_{opt}(\mathbf{m}) - \mathbf{d}^{obs} \quad (13)$$

$$\mathbf{r}_m(\mathbf{m}) = \mathbf{D}\tilde{\mathbf{p}}_{opt}(\mathbf{m}). \quad (14)$$

The gradient of Φ is given by

$$\nabla\Phi(\mathbf{m}) = \left(\frac{\partial\mathbf{r}_d(\mathbf{m})}{\partial\mathbf{m}}\right)^* \mathbf{r}_d(\mathbf{m}) + \epsilon^2 \left(\frac{\partial\mathbf{r}_m(\mathbf{m})}{\partial\mathbf{m}}\right)^* \mathbf{r}_m(\mathbf{m}), \quad (15)$$

and we have

$$\begin{aligned} \left(\frac{\partial\mathbf{r}_d(\mathbf{m})}{\partial\mathbf{m}}\right)^* &= \left(\frac{\partial\mathbf{f}(\mathbf{m})}{\partial\mathbf{m}}\right)^* + \left(\frac{\partial(\tilde{\mathbf{B}}(\mathbf{m})\tilde{\mathbf{p}}_{opt}(\mathbf{m}))}{\partial\mathbf{m}}\right)^* \\ &= \mathbf{B}^*(\mathbf{m}) + \left(\frac{\partial\tilde{\mathbf{B}}(\mathbf{m})}{\partial\mathbf{m}}\tilde{\mathbf{p}}_{opt}(\mathbf{m}) + \tilde{\mathbf{B}}(\mathbf{m})\frac{\partial\tilde{\mathbf{p}}_{opt}(\mathbf{m})}{\partial\mathbf{m}}\right)^* \\ &= \mathbf{B}^*(\mathbf{m}) + \mathbf{T}^*(\mathbf{m}) + \left(\frac{\partial\tilde{\mathbf{p}}_{opt}(\mathbf{m})}{\partial\mathbf{m}}\right)^* \tilde{\mathbf{B}}(\mathbf{m})^*, \end{aligned} \quad (16)$$

where $\mathbf{T}^*(\mathbf{m}) = \left(\frac{\partial \tilde{\mathbf{B}}(\mathbf{m})}{\partial \mathbf{m}} \tilde{\mathbf{p}}_{opt}(\mathbf{m}) \right)^*$. Similarly,

$$\left(\frac{\partial \mathbf{r}_m(\mathbf{m})}{\partial \mathbf{m}} \right)^* = \left(\frac{\partial \tilde{\mathbf{p}}_{opt}(\mathbf{m})}{\partial \mathbf{m}} \right)^* \mathbf{D}^*. \quad (17)$$

Equation 15 becomes

$$\begin{aligned} \nabla \Phi(\mathbf{m}) &= \left[\mathbf{B}^*(\mathbf{m}) + \mathbf{T}^*(\mathbf{m}) + \left(\frac{\partial \tilde{\mathbf{p}}_{opt}(\mathbf{m})}{\partial \mathbf{m}} \right)^* \tilde{\mathbf{B}}(\mathbf{m})^* \right] \mathbf{r}_d(\mathbf{m}) + \epsilon^2 \left(\frac{\partial \tilde{\mathbf{p}}_{opt}(\mathbf{m})}{\partial \mathbf{m}} \right)^* \mathbf{D}^* \mathbf{r}_m(\mathbf{m}) \\ &= \left[\mathbf{B}^*(\mathbf{m}) + \mathbf{T}^*(\mathbf{m}) \right] \mathbf{r}_d(\mathbf{m}) + \left(\frac{\partial \tilde{\mathbf{p}}_{opt}(\mathbf{m})}{\partial \mathbf{m}} \right)^* \left[\tilde{\mathbf{B}}^*(\mathbf{m}) \mathbf{r}_d(\mathbf{m}) + \epsilon^2 \mathbf{D}^* \mathbf{r}_m(\mathbf{m}) \right]. \end{aligned} \quad (18)$$

Since $\tilde{\mathbf{p}}_{opt}$ satisfies equation 9, we have

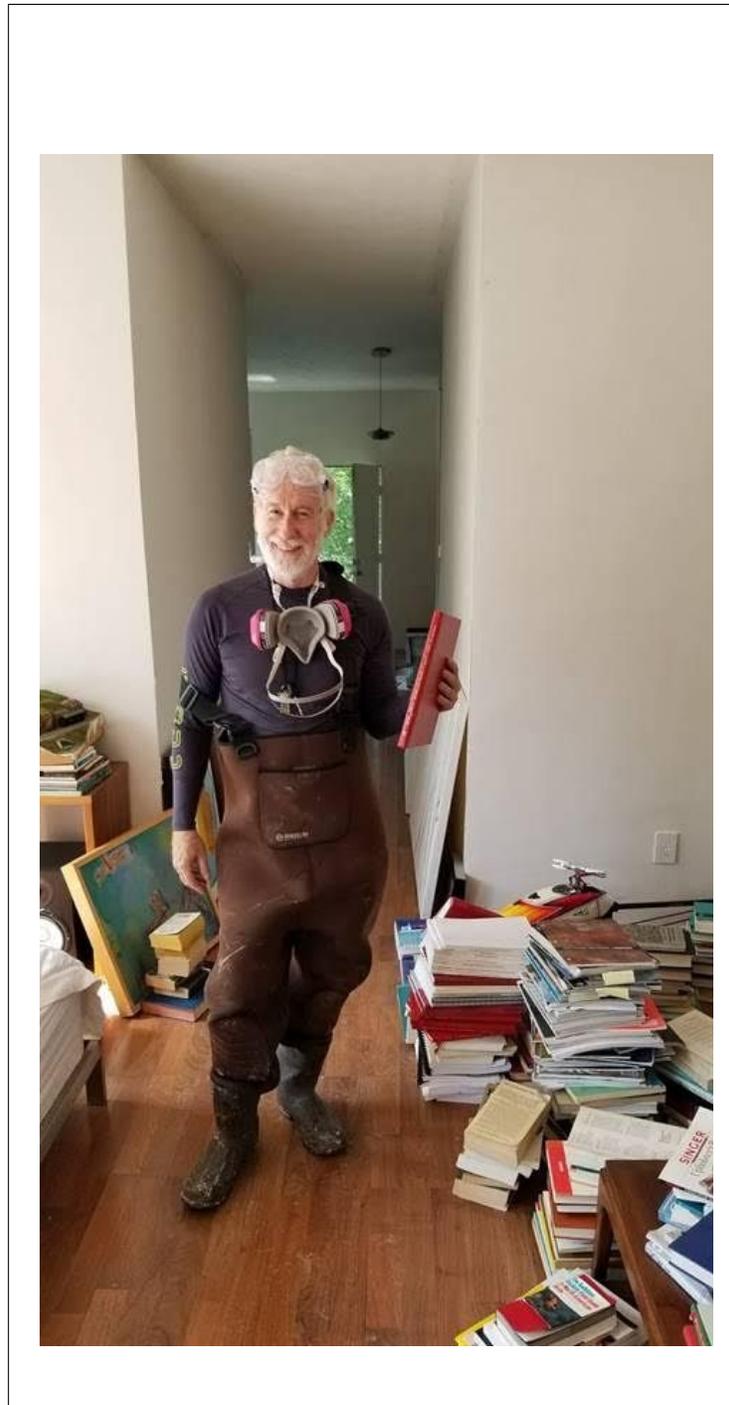
$$\left[\tilde{\mathbf{B}}^*(\mathbf{m}) \tilde{\mathbf{B}}(\mathbf{m}) + \epsilon^2 \mathbf{D}^* \mathbf{D} \right] \tilde{\mathbf{p}}_{opt}(\mathbf{m}) = \tilde{\mathbf{B}}^*(\mathbf{m}) (\mathbf{d}^{obs} - \mathbf{f}(\mathbf{m})). \quad (19)$$

Therefore,

$$\begin{aligned} \tilde{\mathbf{B}}^*(\mathbf{m}) \mathbf{r}_d(\mathbf{m}) + \epsilon^2 \mathbf{D}^* \mathbf{r}_m(\mathbf{m}) &= \tilde{\mathbf{B}}^*(\mathbf{m}) (\mathbf{f}(\mathbf{m}) + \tilde{\mathbf{B}}(\mathbf{m}) \tilde{\mathbf{p}}_{opt}(\mathbf{m}) - \mathbf{d}^{obs}) + \epsilon^2 \mathbf{D}^* \mathbf{D} \tilde{\mathbf{p}}_{opt}(\mathbf{m}) \\ &= \left[\tilde{\mathbf{B}}^*(\mathbf{m}) \tilde{\mathbf{B}}(\mathbf{m}) + \epsilon^2 \mathbf{D}^* \mathbf{D} \right] \tilde{\mathbf{p}}_{opt}(\mathbf{m}) - \tilde{\mathbf{B}}^*(\mathbf{m}) (\mathbf{d}^{obs} - \mathbf{f}(\mathbf{m})) \\ &= \mathbf{0}. \end{aligned} \quad (20)$$

Finally, equation 15 reduces to

$$\nabla \Phi(\mathbf{m}) = \left[\mathbf{B}^*(\mathbf{m}) + \mathbf{T}^*(\mathbf{m}) \right] \left(\mathbf{f}(\mathbf{m}) + \tilde{\mathbf{B}}(\mathbf{m}) \tilde{\mathbf{p}}_{opt}(\mathbf{m}) - \mathbf{d}^{obs} \right). \quad (21)$$



Can full waveform inversion image all scales of the velocity model?

Biondo Biondi, Ettore Biondi, and Guillaume Barnier

ABSTRACT

Full waveform inversion (FWI) reconstructs the velocity model based on the information contained in reflections and diving waves, in addition to a tomographic component that contributes when reflected events are not well focused. The wavenumber-domain analysis of the contributions of each of these elements shows that they are mostly complementary. The illumination patterns in the wavenumber domain are scaled by the data frequency, further expanding the wavenumber-domain region that is illuminated by reflections data. When cycle-skipping is not occurring, FWI applied to wideband data should be able to reconstruct all the scales of the velocity model. Our tests on synthetic datasets support this analysis; however, they also show that FWI applied to wideband data (2.5-30 Hz) with long offsets (up to 9km) is not able to reconstruct a small region of the wavenumber plane close to the origin. For our specific examples, the poorly reconstructed scales have vertical wavelengths longer than 400 m and horizontal wavelengths between 150 m and 600 m.

INTRODUCTION

The ultimate goal of seismic imaging is to image all scales of the subsurface velocity model. Full waveform inversion (FWI) aims to extract the information that is contained in the data for all the scales of the model. FWI estimates are more reliable for some scales (short wavelengths) than for others (long wavelengths) because of “cycle skipping” caused by the non-linearity of the wave equation with respect to the model parameters. To better understand the challenge, and possibly devise more robust inversion algorithms, we analyze the contributions to the process of imaging different scales of the model by different data components. In particular we analyze the contributions by: 1) reflections, 2) diving (overtuned) waves, and 3) a tomographic component that is related to reflections focusing (reflection tomography). These last two categories are linked to transmission (forward-scattered) effects, whereas the first one is linked to reflection (back-scattered) effects.

Our analysis is based on the “classic” graphic understanding of the model wavenumber illumination provided by single-scattered data as a function of the recording geometry that has been presented by Wu and Toksoz (1987). Their analysis is based on the assumption of single scattering in a constant medium background, but it provides a good starting point to analyze diving waves that are recorded only when the background model increases with depth. More challenging is the analysis of the reflection-tomography contributions that come into play only when the background model contains scatterers. The single-scattering framework can be extended to provide a qualitative understanding of the illumination provided by reflection tomography. However, the wavenumber components that are illuminated

by reflection tomography are not only dependent on the scattering geometry, but also on the wavenumber content of the the imaged reflector in the background model. Therefore, we think that a more complete theory that formally explains double-scattered events should be developed to quantitatively analyze the illumination provide by reflection tomography.

To illustrate and validate our analysis we modeled three synthetic datasets and imaged them by FWI. These datasets were modeled assuming random velocity perturbations superimposed onto a linearly increasing background model. The random perturbations for the first dataset were generated assuming a uniform distribution with zero mean and variance of 1.5 m/s. To generate the other two datasets we scaled these random perturbations by a factor of 10 and 20. Because of the low-frequency content of the source function, FWI converges for all three datasets.

MODEL-SCALES ILLUMINATED BY REFLECTION SEISMIC DATA

Wu and Toksoz (1987) introduced a simple graphical method to analyze the illumination of the seismic experiment as a function of acquisition geometry. The method is based on a single-scattering assumption and on plane-waves decomposition of the wavefields. Figures 1 and 2 summarize the basic idea. Figure 1 shows an incident plane wave (red) propagating downward at an angle α with respect to the vertical direction. Subsurface heterogeneities scatter back the incident plane wave. The scattered wavefield can be decomposed into its plane-waves components. The Figure represents one of them (green) propagating back towards the surface at an angle β with respect to the vertical direction. The angle γ is the scattering angle for these two plane waves. Figure 2 shows the corresponding wavenumber representation of the scattering phenomenon represented in Figure 1. If we assume the plane waves to be monochromatic, we can represent them as the vectors \mathbf{k}_s and \mathbf{k}_g in the wavenumber plane as follows:

$$\begin{aligned}\mathbf{k}_s &= \frac{\omega_0}{V_0} (\sin \alpha \mathbf{k}_x - \cos \alpha \mathbf{k}_z), \\ \mathbf{k}_g &= \frac{\omega_0}{V_0} (\sin \beta \mathbf{k}_x - \cos \beta \mathbf{k}_z),\end{aligned}\tag{1}$$

where \mathbf{k}_x and \mathbf{k}_y are the unit vectors along the wavenumber axes, ω_0 is the angular frequency of the plane waves, and V_0 is the constant velocity of the medium. The model wavenumber vector \mathbf{k}_m (black in the Figure) illuminated by the data component corresponding to the scattered plane wave is equal to

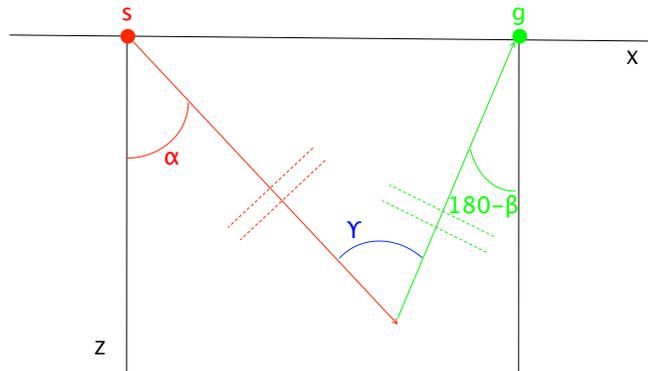
$$\mathbf{k}_m = \mathbf{k}_s - \mathbf{k}_g.\tag{2}$$

Figure 2 shows also the scattering angle γ as a function of the wavenumber-domain vectors.

Figure 3 shows the region of the model space that are illuminated by a single frequency, for all possible angles α and β . The circumference of the large gray circle corresponds to the normal-incidence reflections, which are the highest-resolution events that can be recorded at ω_0 . The orange circle is the region of the wavenumber plane that is illuminated by forward-scattered plane waves; that is, scattered plane waves for which $90^\circ \leq \gamma \leq 270^\circ$. We will refer to these events as transmission events.

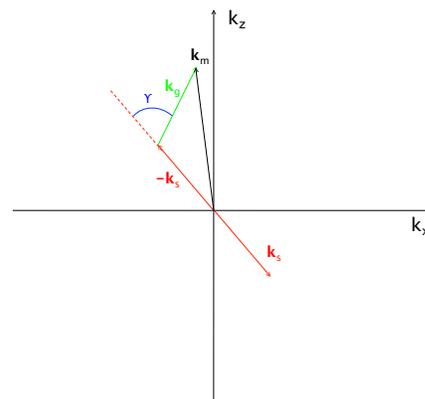
The light-blue circles on the side, and partially covered by the orange circle, are the regions of the wavenumber plane that require the vertical wavenumber of the incident and

Figure 1: The incident plane wave (red) generated by the source(s) propagates downward at an angle α with respect to the vertical. The scattered wavefield can be decomposed into several plane waves; one of them (green) propagates back towards the surface at an angle β with respect to the vertical. The angle γ is the scattering angle for these two plane waves. [NR].



biondo1/. PlaneWaves

Figure 2: Wavenumber representation of the plane-waves scattering phenomenon represented in Figure 1, accordingly to equations 1 and 2. [NR].



biondo1/. Vectors-Kspace

scattered plane waves to have the same sign. This condition is equivalent to the condition that the incident plane wave and the scattered plane wave are traveling along the same vertical direction. In a constant velocity background and under the assumption of single scattering, these plane waves cannot be recorded by seismic data acquired at the surface. Therefore, conventional reflections (that is back-scattered energy recorded at the surface) can only illuminate the regions of the gray circle that are not covered by the orange and light-blue circles.

However, if in addition to reflections we record also diving waves caused by a background model with increasing velocity with depth, we can illuminate an additional region of the wavenumber plane. The part of the orange circle visible in Figure 4 shows the region of the wavenumber plane that can be illuminated by diving waves. The recorded data are forward scattered and the incident and scattered energy propagate along the same horizontal direction, but opposite vertical directions (at least with a constant velocity background; in the presence of a vertical gradient this condition does not need to be exactly fulfilled).

Second-order scattering illuminates an additional region of the wavenumber plane. These contributions are present in the FWI gradients if the background model contains sufficient

short-wavelength heterogeneity to cause substantial back scattering, as typically happens for FWI iterations after the first one. We can consider this term the “reflection tomography” component of seismic imaging. Reflection tomography illuminates the regions of the orange circle that are visible in Figure 5. These wavenumber components are illuminated by forward scattered energy for which the incident and scattered plane waves propagate along the same vertical direction, either downward for the source wavefield, or upward for the receiver wavefield. When the source plane wave is scattered, the incident wavenumber vector points down (upward for $-\mathbf{k}_s$ in Figure 5) and the scattered wavenumber vector ($\mathbf{k}_{s'}$ in the Figure) points downward. Conversely, when the reflected plane wave is scattered, the incident wavenumber vector \mathbf{k}_g points up, and the scattered wavenumber vector $\mathbf{k}_{g'}$ points upward. It should be noted that the reflection tomography contribution not only depends on the presence of reflectors in the background model, but also that it depends on their orientation and frequency content.

The transmission components (diving waves and reflection tomography) are the most prone to cycle skipping when the background model is far from the correct one. Common workflows based on ray-tracing tomography are usually capable to estimate starting models that avoid cycle skipping of diving waves arrival. In contrast, ray-tracing tomography is less likely to yield sufficiently accurate models to avoid cycle skipping in the reflection-tomography FWI component.

Figure 6 shows the same circles shown in Figure 3 with their copies, scaled down by a factor of two, superimposed onto them. These smaller circles correspond to plane waves with half the frequency of the previous one; that is, $\omega_h = \omega_0/2$. When we record wideband data (i.e. $\omega_h \leq \omega \leq \omega_0$) the different data components (reflections, diving waves, and reflection tomography) illuminate the regions of the wavenumber plane that are swept as the circles continuously expand from the small ones to the large ones, as indicated by the blue arrows in Figure 6.

SYNTHETIC EXAMPLE

To illustrate the concepts presented in the previous section we modeled three synthetic datasets assuming random perturbations superimposed onto a background model with velocity linearly increasing with depth; the background velocity was defined as $v(z) = (1 + 0.44z)$ km/s. Because velocity increases with depth, this background model creates overturned events. The random perturbations for the first dataset (#1) were generated assuming a uniform distribution with zero mean and variance of 1.5 m/s. The other datasets were generated using the same random perturbations scaled up by a factor of 10 (#2) and 20 (#3). The random perturbations ensure that the perturbed model is different from zero at all the wavenumbers. We modeled 48 sources equally spaced 150 m apart. The receiver array was 10 km wide and stationary for all sources. Figure 7 shows that the spectrum of the sources is different from zero between 2.5 Hz and 30 Hz. The spatial grid was 10 m in both directions; therefore the Nyquist wavenumber was $.05 \text{ m}^{-1}$.

Figure 3: The regions of the model space that are illuminated by a single frequency, for all possible angles α and β . The gray areas are illuminated by surface data with reflections. The orange circle is illuminated by forward-scattered plane waves. The light-blue areas cannot be illuminated by surface data. [NR].

biondo1/. Reflections-OneFreq

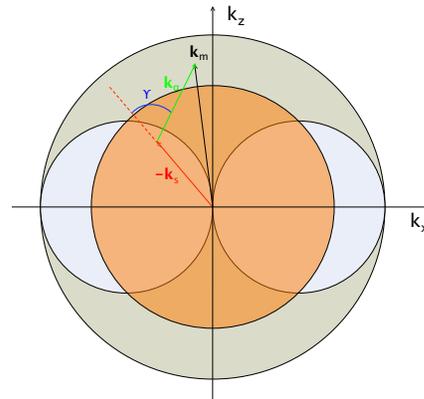


Figure 4: The portion of the orange circle not covered by the light-blue circles is illuminated by diving waves recorded by long-offset surface data when the velocity increases with depth. [NR].

biondo1/. Diving-OneFreq

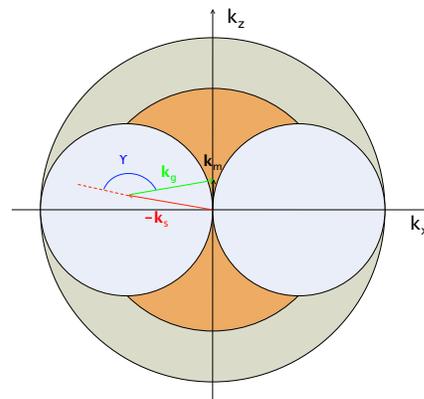


Figure 5: The portion of the orange circle not covered by the light-blue circles is illuminated by the “reflection tomography” component of the data. It is present in the FWI gradients when the background model contains scatterers as typically happens for FWI iterations after the first one. [NR].

biondo1/. ReflTomo-OneFreq

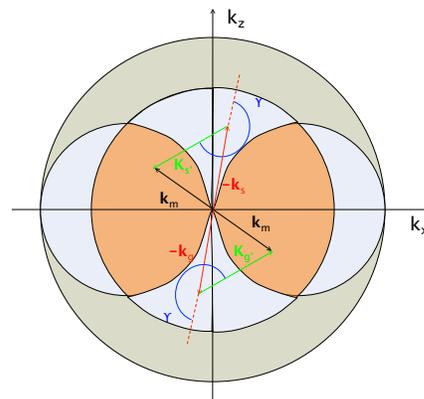
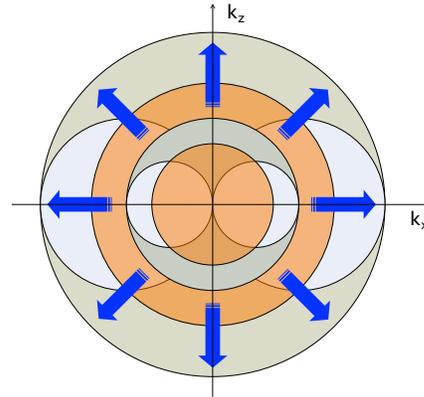


Figure 6: The circles shown in Figure 3 with their copies scaled down by a factor of two superimposed onto them. These smaller circles correspond to plane waves with half the frequency of the previous one. Wideband data illuminate the regions of the wavenumber plane that are swept as the circles continuously expand from the small ones to the large ones. [NR].

biondo1/. Reflections-TwoFreq



First-iteration FWI images

Figure 8a shows the search direction at the first iteration of a FWI process applied to dataset #1; we can clearly see the contributions of the reflected events and the diving waves superimposed onto each other. To help distinguish these two types of contributions to the search directions we approximately separated the corresponding events in the data domain by applying a linear muting as a function of offset that removed the diving waves from the data. Figure 8b shows the search direction computed by back-projecting the muted residuals, whereas Figure 8c shows its complement; that is, the search direction corresponding to the diving waves.

Figure 9 shows the wavenumber spectra computed from the images shown in Figure 8. The shape of these wavenumber spectra fits the graphical analysis presented in Figures 3 and 4. The characteristic eye-glasses that correspond to the light-blue circles in Figure 4 are clearly visible in Figure 8a as areas of amplitudes close to zero. As graphically illustrated by Figure 6, because the sources were wideband these circles are stretched horizontally and their borders are fuzzy.

For dataset #1 the velocity perturbations are very small; therefore, we expect the images shown in Figure 8 are good approximations of the true perturbations within the respective wavenumber regions. Figures 10a and 10b compare the bandpassed true perturbations to the imaged reflections, respectively. To facilitate the comparison Figure 10b shows the same image as shown in Figure 8b. To generate Figure 10a we applied a mask in the wavenumber domain that was designed to attenuate all the wavenumbers except within the region where we would expect the reflected events to be imaged. We can notice that wherever the random perturbations align to create semi-coherent events, these events are consistent between the two displays. Because velocity increases with depth the wavelengths increase with depth in the image (Figure 10b). Furthermore, because of the limitations of a surface-bound acquisition geometry, the angular bandwidth drastically decreases with depth in the actual image (Figure 10b).

Figure 10c shows the wavenumber spectrum of the image shown in Figure 10a and has similarities with Figure 9b. The lower amplitudes at high horizontal wavenumbers in

Figure 7: Frequency spectrum of the wavelet used to model the datasets. [CR]

biondo1/. Spec-Wavelet-Grad

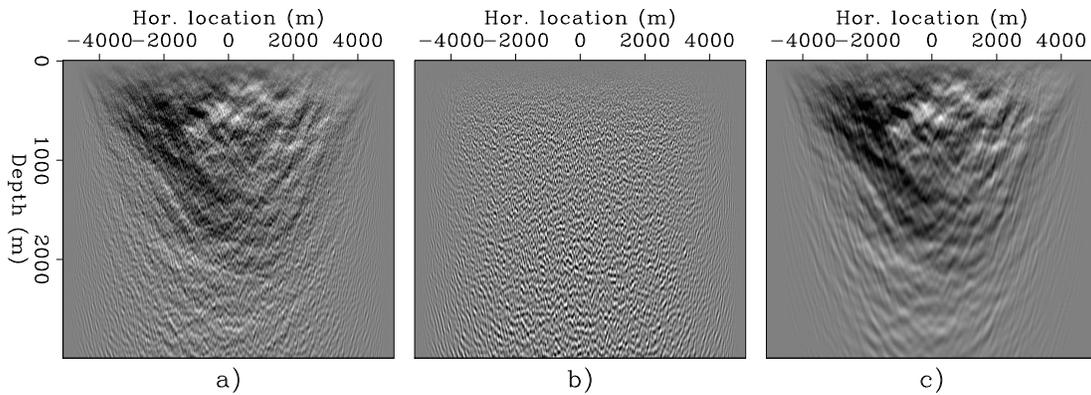
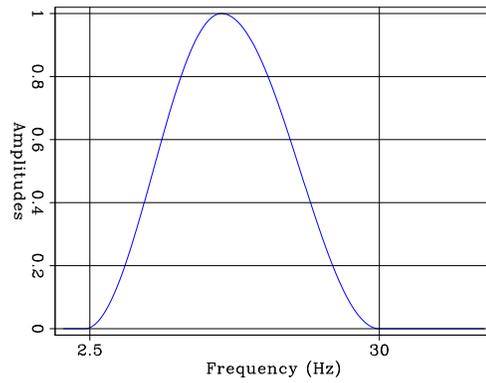


Figure 8: Search directions at the first iteration of a FWI process applied to dataset #1. To produce panel a) all components of the data (reflections and diving waves) are backprojected into the model. To produce panels b) and c) reflections (b) and diving wave (c) are separately backprojected into the model. [CR] biondo1/. Shape-all

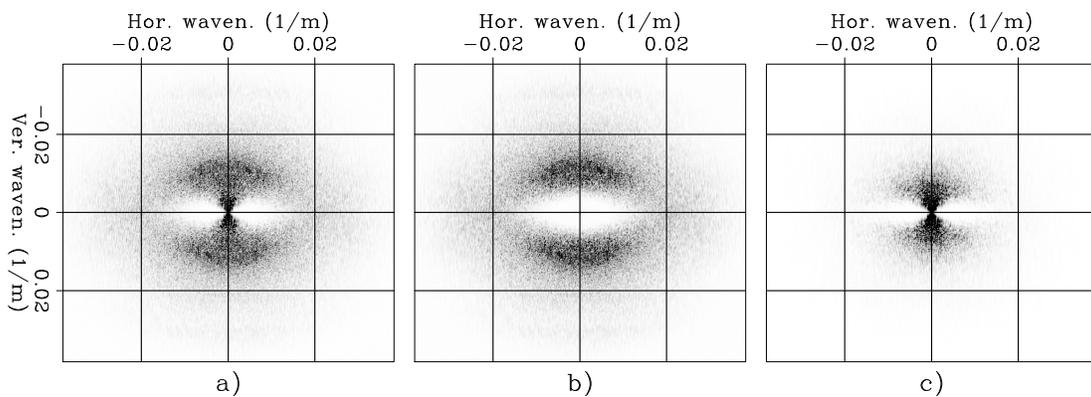


Figure 9: Wavenumber spectra computed from the images shown in Figure 8 and corresponding to a) reflections and diving waves, b) reflections, and c) diving waves. [CR]

biondo1/. Spec-all

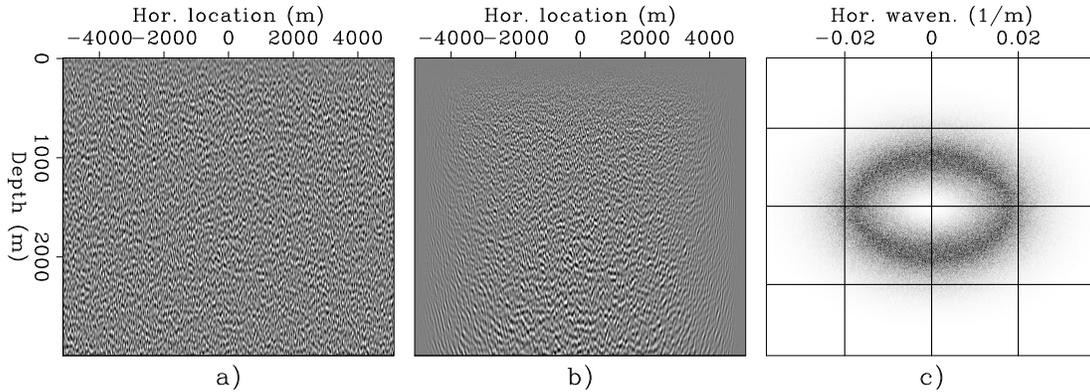


Figure 10: Bandpassed true perturbations (panel a) compared to the imaged reflections (panel b). Wherever the random perturbations align to create semi-coherent events, these events are consistent between the two images. Panel c) shows the wavenumber spectrum of panel a). [CR] biondo1/. Comp-refl

Figure 9b with respect to Figure 10c are caused by the loss of angular bandwidth with depth.

Similarly to Figure 10, Figure 11 compares bandpassed true perturbations to the imaged diving waves, respectively. In the shallow section there is an approximate correspondence between the imaged velocity anomalies and the true bandpassed perturbations. Figure 11c shows the wavenumber spectrum of the image shown in Figure 11a. As for the previous Figure, to facilitate the comparison between the bandpassed true perturbations and the estimated image, Figure 11b shows the same image as shown in Figure 8c.

Second-iteration FWI images

From the analysis presented in the previous section, we would expect that at the second iteration of a FWI procedure the reflection-tomography component of the gradient will fill some of the wavenumber-components missing from the images after the first iteration. Figure 12 shows the effects of the reflection-tomography component when FWI is applied to dataset #2. Figure 12b shows the search direction of the second iteration of FWI after applying a lowpass filter in the wavenumber domain to remove the most of the reflection components from the image. This image was generated by back-projecting the residuals after applying a linear mute to remove the diving-waves arrivals. We applied the same mute that we applied to the first-iteration residuals to generate the reflection image shown in Figure 8b. The muting of the residuals and the lowpass of the image were applied to isolate the reflection-tomography contributions as much as possible. These contributions tend to have lower amplitudes and would be easily missed if not isolated from the stronger contributions of the reflections and diving waves.

Figure 12a shows the true perturbations after applying the same lowpass filter applied to the second-iteration search direction. Comparing Figure 12a with Figure 12b we can match several narrow velocity anomalies. These anomalies are better resolved in the reflection-tomography image than in the corresponding diving-waves image obtained at the

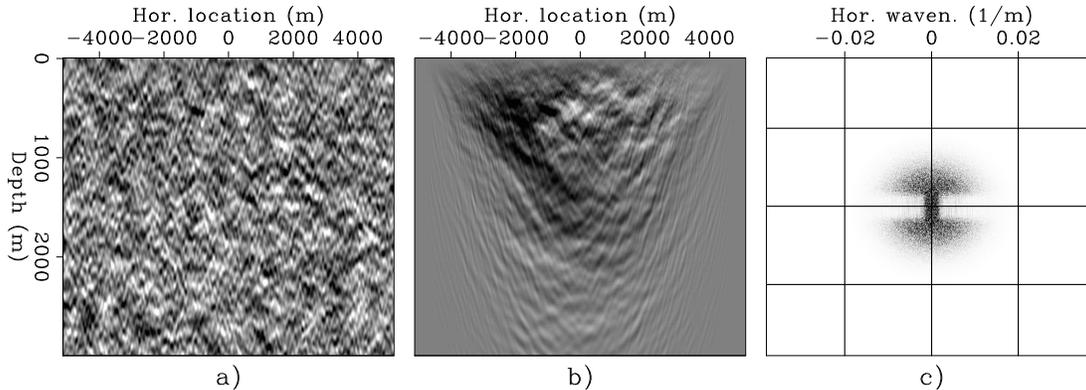


Figure 11: Bandpassed true perturbations (panel a) compared to the imaged diving waves (panel b). In the shallow section there is an approximate correspondence between the imaged velocity anomalies and the true bandpassed perturbations. Panel c) shows the wavenumber spectrum of panel a). [CR] `biondo1/. Comp-div`

first iteration of FWI applied to dataset #1 (Figure 12c.) This increase in lateral resolution is reflected in the wavenumber spectrum of the reflection-tomography image that is shown in Figure 13a. The spectra corresponding to the reflections and diving-waves images obtained after the first iteration are shown again in Figure 13b and 13c. This example shows that the contribution of reflection tomography is complementary to the one of diving waves and reflections. However, it also shows that a smaller region of the wavenumber plane is illuminated by the reflection tomography component than we would have expected from the graphic analysis presented in the previous section.

Figure 14 compares the lowpassed second-iteration search directions obtained for datasets #1, #2, and #3. Multiple scattering caused by the random velocity perturbations causes a slow down of the recorded reflections and causes the reflection-tomography component of the velocity updates at the second iteration to be biased towards negative updates. Because the kinematic error is very small for dataset #1 the image in Figure 14a is contaminated by reflections. In contrast, the stronger multiple-scattering caused by the larger velocity perturbations in datasets #2 and #3 causes an overall slow down of the recorded reflections and thus it enhances the tomographic component. Figure 14c shows similar features as Figure 14b, but it is more biased towards negative velocity updates because events are slowed down by multiple-scattering more in dataset #3 than in dataset #2.

FWI images

When we perform many iterations of FWI on the whole data; that is, without separating reflections from diving waves as in the experiments shown above, all the three components analyzed above (reflections, diving waves and reflection tomography) simultaneously contribute to the inversion process. If the initial velocity error is sufficiently small, the FWI process converges to an optimal model that approximates the true model for all the wavenumbers that are illuminated by the data.

Figure 15 shows the results of applying FWI to all the data components (reflections

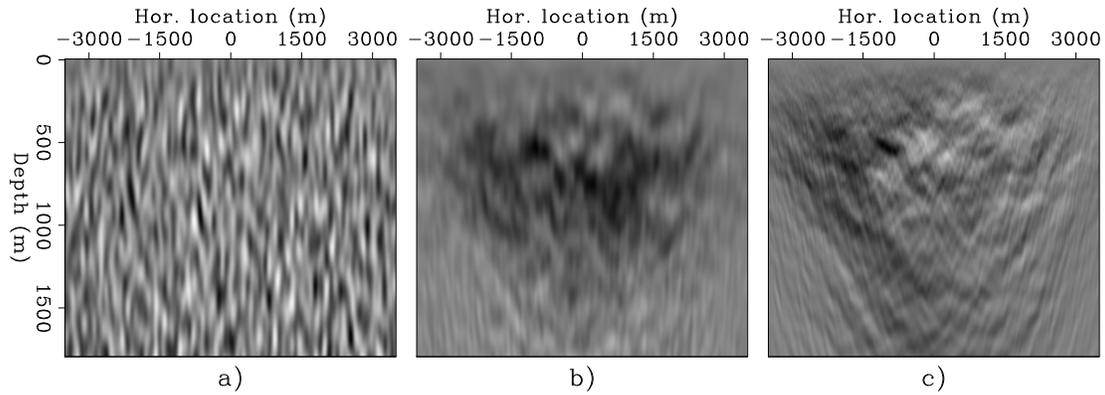


Figure 12: Bandpassed true perturbations (panel a) compared to the "reflection tomography" contributions to the second search direction of a FWI process (panel b). Panel c) shows the shallow part of the image obtained from the first iteration of FWI applied to diving waves in dataset #1. (Figure 11b.) [CR] [biondo1/. Tomo-Wind](#)

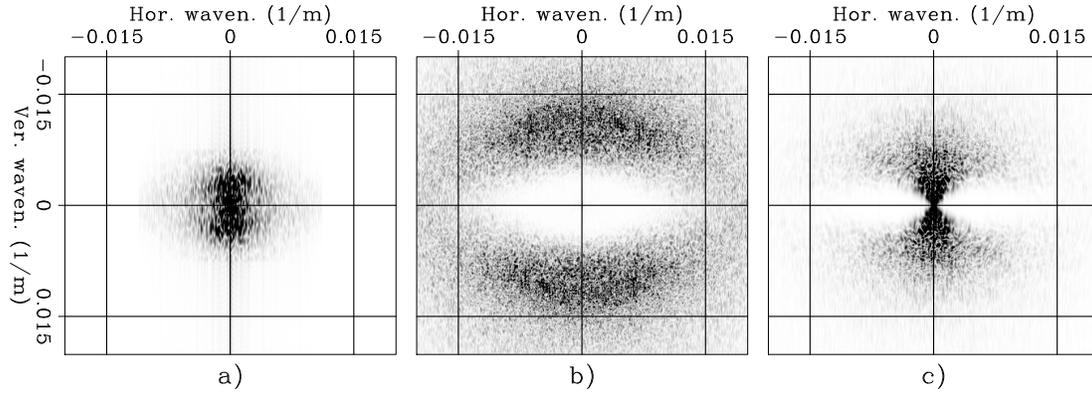


Figure 13: Wavenumber spectra computed from the images shown in: a) Figure 12b, b) Figure 8b, and c) Figure 8c. [CR] [biondo1/. Spec-tomo](#)

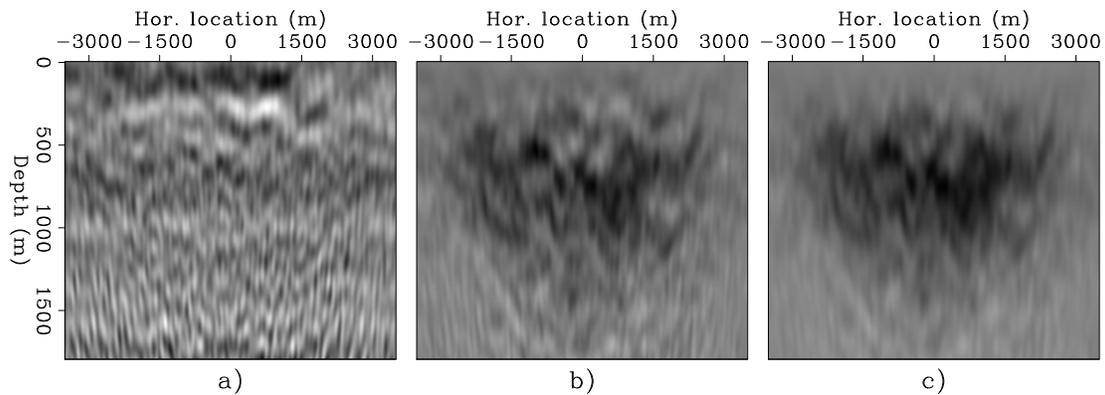


Figure 14: Figure 14 compares the lowpassed second-iteration search directions obtained by applying FWI to datasets #1, #2, and #3. Notice the increasing (left to right) bias towards negative updates related to the slow down caused by increasing multiple scattering in the data. [CR] [biondo1/. Comp-Tomo-Wind](#)

and diving waves) for dataset #1 after 10, 50, and 150 iterations of FWI using a BFGS solver. Figure 16 shows the wavenumber spectra of the models shown in Figure 15. The first 10 iterations solve for the shallow long-wavelength anomalies, mostly driven by the diving waves in the data. Iterations between 10 and 50 reach to the deeper part of the model and start to bring up the short wavelengths directly illuminated by the reflections. The last 100 iterations equalize the amplitudes of the model in depth and across the wavenumbers by resolving some of the intermediate wavelengths at deeper depths.

The spectral holes with the shape of eyeglasses are squeezed by the the later FWI iterations compared to the first ones. However, they are persistent indicating that there are some wavelengths of the model that cannot be resolved from the surface data even in presence of low frequencies, large offsets, and a vertical velocity gradient. Figure 17 shows windows around the origin of the spectra shown in Figure 16. It shows that the FWI process has not resolved model components with vertical wavelengths longer than 400 m and horizontal wavelengths between 150 m and 600 m.

DISCUSSIONS AND FUTURE WORK

The graphic analysis of the wavenumber illumination by reflections, diving waves, and reflection tomography shows that FWI has the potential to image all scales of the velocity model, if cycle-skipping does not occur. However, the synthetic-data examples shows that there is a region of the wavenumber plane that is not illuminated by the data. The most likely culprit for this limitation is that reflection tomography does not resolve all the wavenumbers that the graphic analysis we presented would indicate. A more complete theory that formally explains double-scattered events should be developed to quantitatively analyze the illumination provide by reflection tomography and help develop methods to overcome this limitations.

REFERENCES

- Wu, R. and M. N. Toksoz, 1987, Diffraction tomography and multisource holography applied to seismic imaging: *Geophysics*, **52**, 11–25.

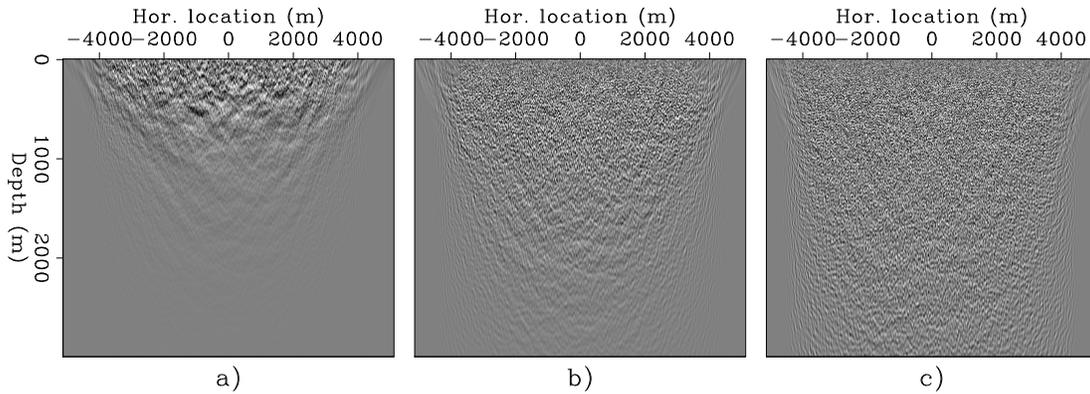


Figure 15: Velocity models obtained by FWI applied to dataset #1 after: a) 10 iterations, b) 50 iterations, and c) 150 iterations. [CR] `biondo1/. Image-pert`

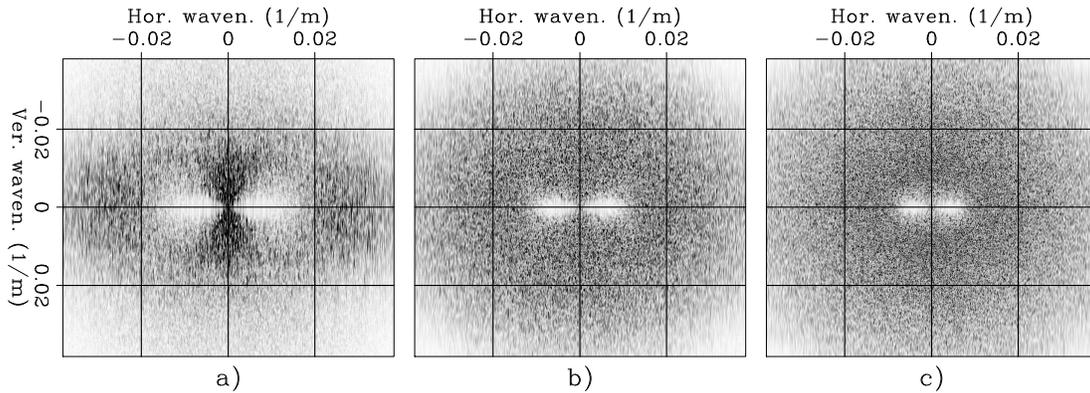


Figure 16: Wavenumber spectra of the velocity models obtained by FWI applied to dataset #1 after: a) 10 iterations, b) 50 iterations, and c) 150 iterations. [CR] `biondo1/. Spec-pert`

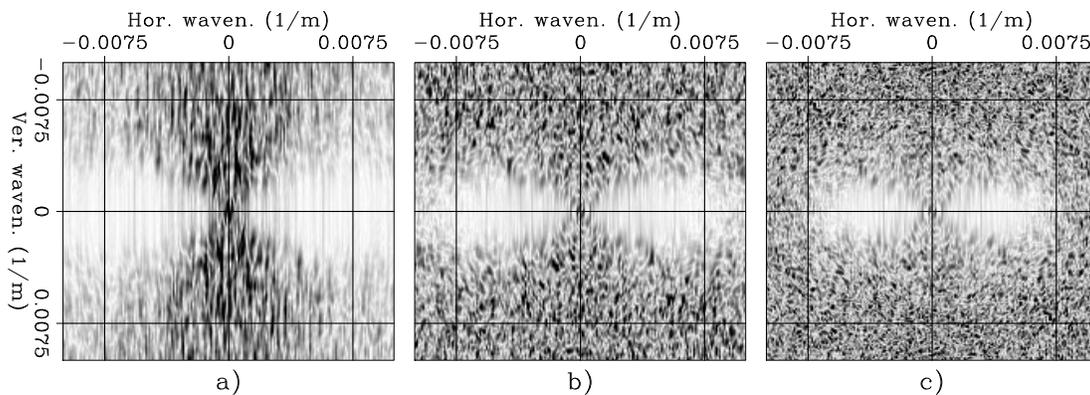


Figure 17: Windows of the wavenumber spectra shown in Figure 16 that focuses on the area of the wavenumber plane that are least resolved by the FWI process. [CR] `biondo1/. Spec-wind-pert`

Representing salt bodies with radial basis functions

Taylor Dahlke

ABSTRACT

In this work we show how radial basis functions can be used to sparsely represent the implicit surface used to represent salt bodies. We show that this methodology is effective even when the model parameter reduction is roughly 2% of the original model size. This is important to making shape optimization effective for 3D velocity models. When the Hessian of a modified FWI objective function is used for shape optimization, we must invert the Newton system for the search direction. When using iterative methods like conjugate gradient for this, the reduction in parameters improves the speed and stability of this inversion.

INTRODUCTION

Previously, Kadu et al. (2016) demonstrated how radial basis functions (RBFs) can be used to help define salt features using level sets. In their work, they chose to use a regular gridding for the RBF centers. This has the advantage of allowing the same resolution of updates to all areas of the salt model. Assuming that a reasonably good initial salt pick is chosen, we can further assume that any changes to this salt boundary will occur within its vicinity. Following this, we prefer to concentrate our resolution (and subsequently the RBF centers) around the areas where we actually expect updating to occur, i.e, the edges of the initial salt body picks. For this reason, we apply an approach that builds a probability distribution around the edges of the initial salt picks, and then randomly assigns RBF centers based on this probability distribution. This builds a ‘cloud’ of RBF centers that is densest where we expect the updates to occur and gives us enhanced resolution in these areas using far fewer parameters than a regular grid approach would require to achieve the same resolution. In this paper we first show how we create this probability distribution. Then we describe the linearization that we used to perform the non-linear inversion, and last we show results that demonstrate the efficacy of the method on the Cardamom salt model.

INVERSION

To use a sparse representation of our implicit surface as the model in an FWI type workflow, we first have to begin with some initial model. We find this by starting with some initial binary guess of salt/no-salt in the spatial field such as the salt overlay shown Figure 1. Next we use a non-linear inversion workflow to find the RBF parameters that best fit that 2D spatial salt mapping. We define the function that maps from the RBF parameter (λ) space to the implicit surface (ϕ) space as:

$$\phi(\lambda; \epsilon, r) = \sum_i^{N\lambda} \lambda_i \exp^{-(\epsilon r)^2}, \quad (1)$$

where r is the radial distance from the RBF center and ϵ is a constant that influences the taper of the Gaussian kernel. With this formulation of the implicit surface, we can then map from the implicit surface (ϕ) to a velocity field (m) using the non-linear Heaviside function (H):

$$m(\lambda) = H(\phi(\lambda)). \quad (2)$$

We can approximate the Heaviside as a differentiable function \hat{H} , which allows us to take its derivative as the smooth Dirac-delta like term $\hat{\delta}$, and thus find a gradient for equation 2:

$$\Delta m = \hat{\delta}(\phi(\lambda)) \sum_i^{N\lambda} \exp^{-(\epsilon r)^2} \Delta \lambda. \quad (3)$$

With this, we can code up linear forward and adjoint operators that allow us to transform between perturbations in the velocity and RBF parameters. Using this with the non-linear forward (equation 2), we can use a non-linear solver to invert for the correct RBF parameters (λ). I used a python-based, out-of-core solver developed by Ettore Biondi and Guillaume Barnier to perform the inversion itself (Biondi and Barnier (2017)).

APPLICATION

In order to preserve efficiency of the algorithm, we keep the RBF parameter ϵ constant, and thus use the same Gaussian function for each RBF kernel. This means we only solve for the weights (λ) applied to each of these kernels. Because this RBF kernel is static, we can calculate it beforehand. Since the value of the RBF diminishes with radial distance, we only use a small section of the full RBF that is in a region of relatively close radius to the center. Far-radius regions of the RBF are negligibly small, and so the computational cost of storing and using a larger kernel is a fruitless exercise.

Parameter choices

Because we choose to define our RBF kernel before inversion, we need to find parameter values for ϵ , for the size of the kernel footprint, and for the density of the RBF centers themselves (see Figure 3 showing probability mapping and Figure 4 showing the resulting RBF centers). These must be determined manually beforehand, and the interplay between the parameters must be analyzed. For example, if we choose a ϵ that is too large, the RBF will taper off rapidly. If a regular gridding of RBF centers is used, it is relatively easy to calculate a ϵ value that will allow for full spatial coverage via overlapping RBF functions. However, since we choose a randomized centering for each RBF, we need to check to see that our ϵ value is small enough that the summation of all RBF functions has complete

coverage over the spatial domain we are working with. However, if it's too small, then we smooth out the radial basis functions, and thus decrease the resolution of the implicit surface we create. Alternatively, we could keep our ϵ value and RBF centers the same, and increase the footprint of the kernel itself. We show the impact of varying ϵ with a fixed kernel footprint and RBF centers in Figures 2(a) and 2(b). In Figure 2(a) where we use $\epsilon = 2.25$, we get RBF kernels that taper off quickly, leaving little to no overlap in areas where the RBF are more sparsely centered. However, if we decrease ϵ to 0.25, we reduce this tapering and get full coverage in the area of interest (Figure 2(b)). This does reduce the precision slightly around the boundaries themselves since the smoother RBF mean we have less resolution in our aggregated implicit surface. Overall, the parameters that tune the minimum probability of a center occurring, ϵ , and kernel footprint size must be balanced to achieve both full coverage, low number of parameters (RBF centers), and resolution of the resulting aggregate surface. This tuning can be done relatively quickly by doing simple inversions on small test sections of the full model.

Cardamom salt model

The Cardamom field is in the Gulf of Mexico, and lies about 360 km south-west of New Orleans, Louisiana in approximately 830 m of water. The reservoir itself sits beneath thick layers of salt in rock more than 6 km below the sea floor. We choose a section of the velocity model provided to us by Shell that has a notable salt protrusion in it as an example.

Beginning with a salt model from the Cardamom field dataset (Figure 1), we are able to build a probability density map that favors putting RBF centers near the original picked boundary (Figure 3). From this, we are able to generate random RBF positions (Figure 4). Using these RBF centers, we then perform a conjugate gradient inversion to find the proper weighting of the RBF kernels in order to best fit our starting model. The inversion converges relatively quickly (Figure 5), and produces a result that is relatively close to the desired matching model. Figure 2(b) shows that the matching model and the resulting inverted model generated from sparse RBF parameters are quite similar.

CONCLUSIONS

In this paper we showed that radial basis functions can be used to significantly reduce the number of parameters necessary to represent a 2D salt model. For the Cardamom example we achieve 98% reduction in model parameters while still closely matching the original boundary described in the fully-gridded space. The benefit of using random locations for RBF centers is that we can achieve a higher resolution implicit surface than if we were to use a regular gridded model for the same number of model parameters. However, we need to take the time to choose our parameters like ϵ carefully in order to maintain full coverage, high model reduction, and final surface resolution. In future work, I hope to show how shape optimization can incorporate this approach on a larger 3D cube of the same Cardamom velocity model for the purposes of model refinement according to the FWI objective function.

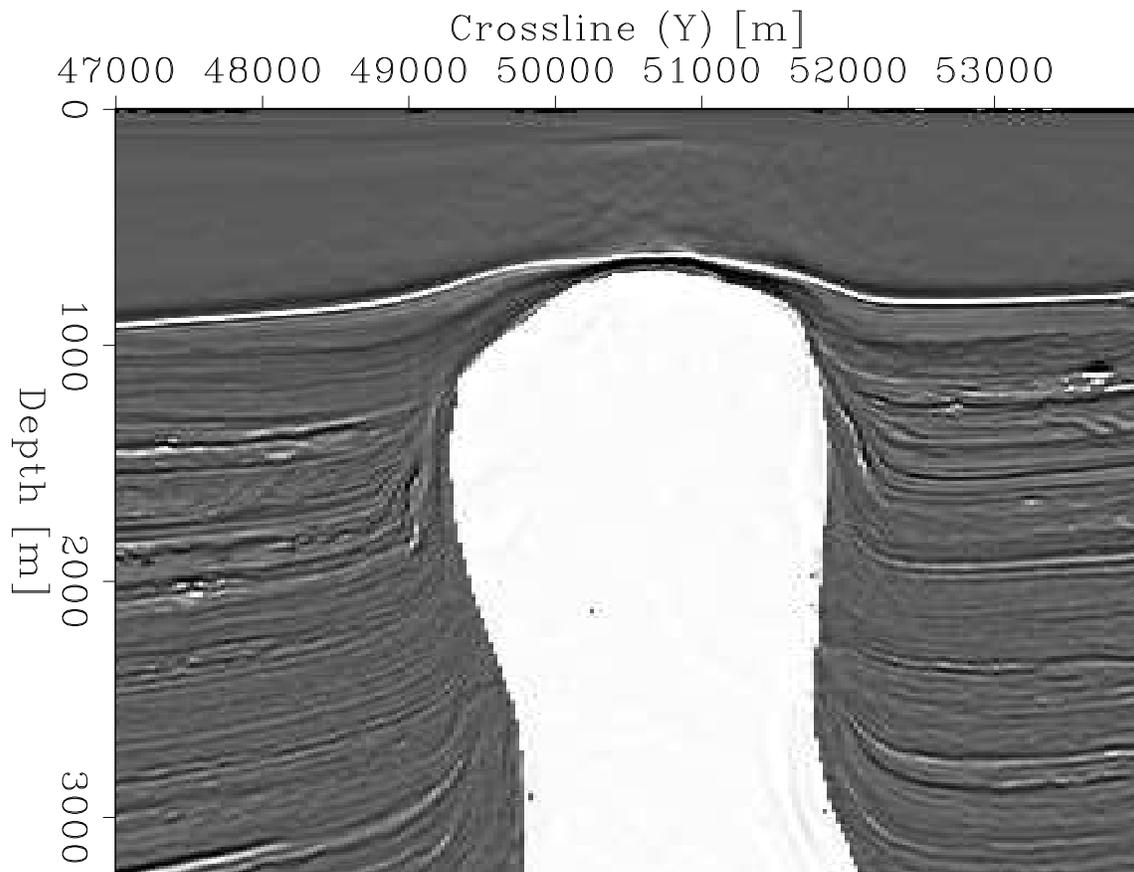


Figure 1: Overlay of salt model used by Shell and the corresponding RTM image. [ER]
taylor1/. both

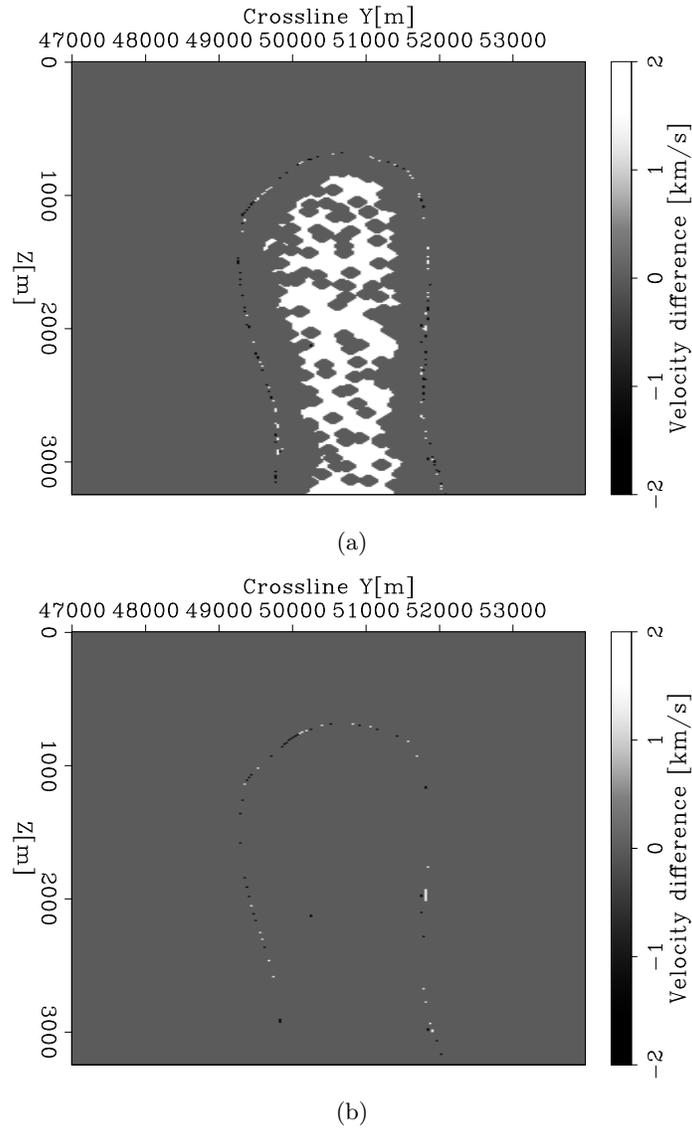


Figure 2: Differences between original salt model and the resulting model produced by the radial basis function representation. (a) shows fitted salt model using $\epsilon = 2.25$ value, while (b) shows fitted salt model using $\epsilon = 0.25$ value. Both cases used 98% fewer model parameters than the original full-grid scheme. Background velocity is 2.5 km/s and salt velocity is 4.5 km/s. [ER] `taylor1/.rbfinv-diff-sparse,rbfinv-diff-full`

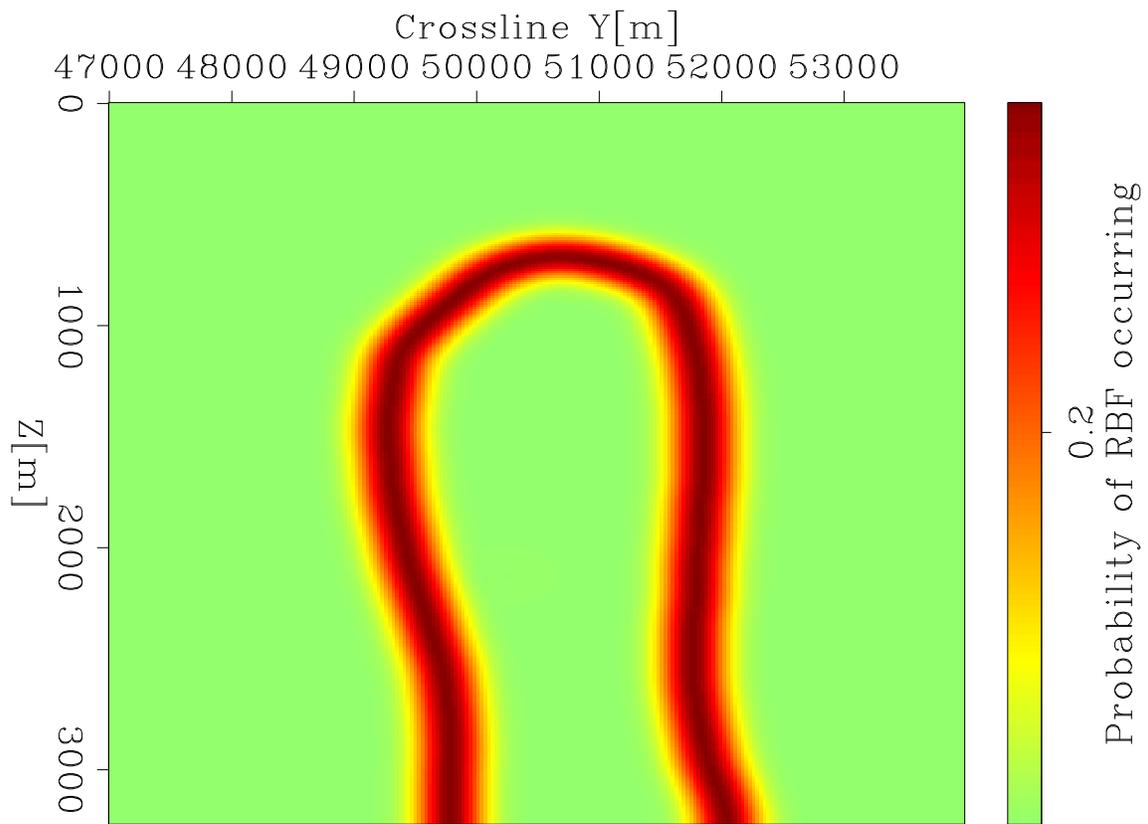


Figure 3: The probability distribution that was used to randomly position the radial basis function centers. [ER] `taylor1/. centers-dist`

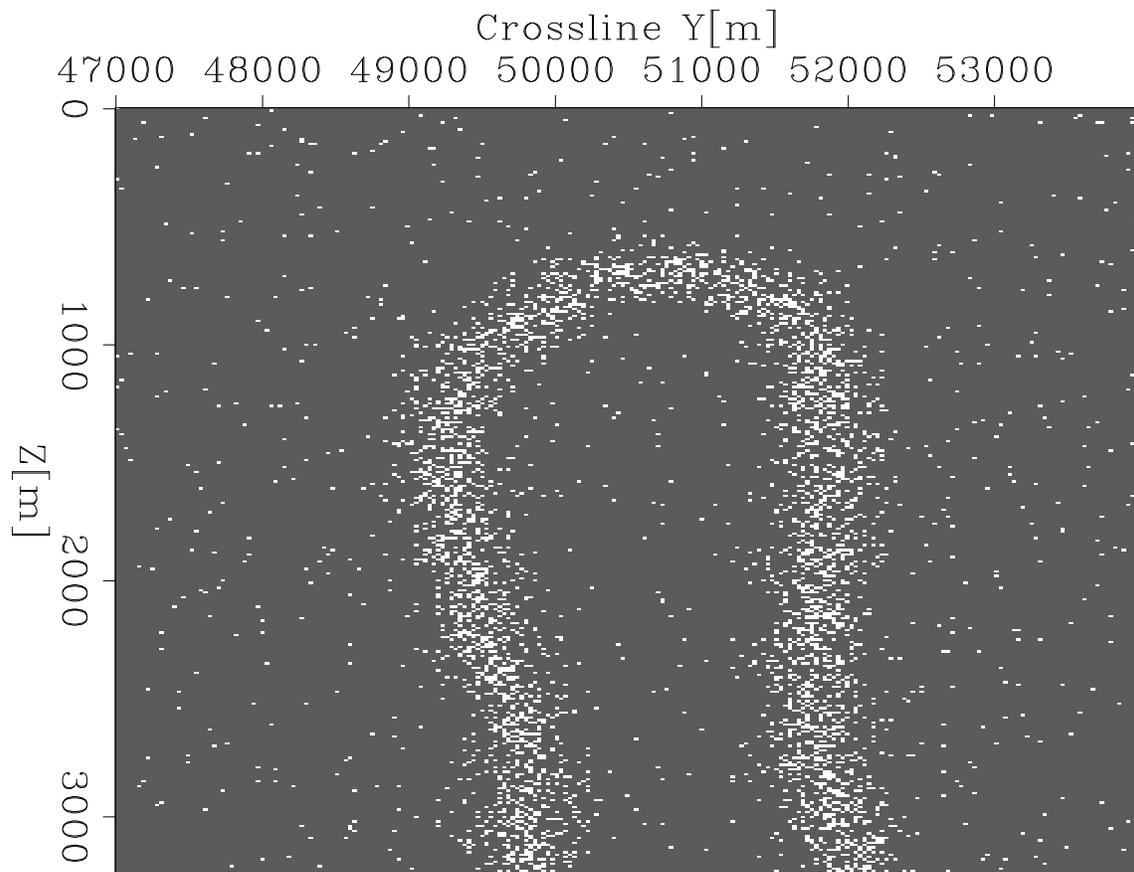


Figure 4: Center points for radial basis functions used to construct the implicit surface.
[ER] `taylor1/. centers`

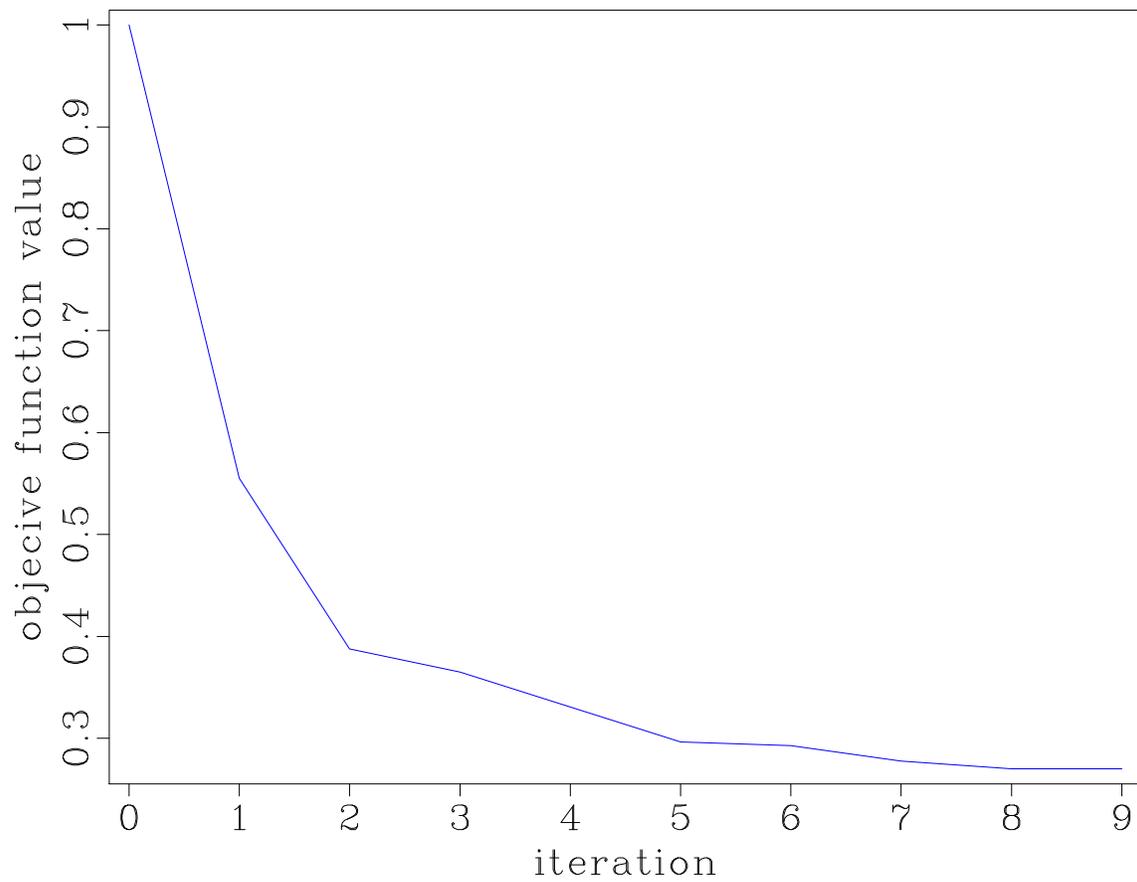


Figure 5: Objective function from the non-linear inversion used to find the RBF parameters.

[ER] `taylor1/. objfunc`

ACKNOWLEDGEMENTS

I would like to acknowledge Shell Oil for providing the RTM data and salt model used in this work. I would like to thank Bonnie Jones, Jan Douma, and their affiliates at Shell for their help in accessing and working with this dataset.

REFERENCES

- Biondi, E. and G. Barnier, 2017, A flexible out-of-core solver for linear/non-linear problems: SEP-Report, **168**.
- Kadu, A., T. van Leeuwen, and W. A. Mulder, 2016, Salt reconstruction in full waveform inversion with a parametric level-set method: CoRR, **abs/1610.00251**.

Using the Hessian of a radial basis formulation for level set inversion

Taylor Dahlke, Biondo Biondi, and Robert Clapp

ABSTRACT

Salt bodies provide complex imaging challenges because of their geometry and reflective properties due to the (often) sharp contrast of wave speed between salt and sediments. Level sets are a useful tool to define and refine discrete boundaries of salt using an implicit surface to describe them. Furthermore, we can represent the implicit surface using a sparse representation based on radial basis functions (RBFs). Using linear operators to map from RBF parameter space to wave speed space, we develop a new formulation of the Full Waveform Inversion (FWI) objective function, and then take the second derivative to get a formulation of its Hessian. We can then solve the corresponding Newton system to find a search direction. The sparse representation offered by the RBF scheme means that a truncated iterative inversion is intrinsically faster due to the large reduction in model parameters that we need to solve for. We demonstrate the efficacy of using the Gauss-Newton approximation of this Hessian, as well as explore the limitations of using the full Hessian formulation for finding a search direction.

INTRODUCTION

Previous work done by Kadu et al. (2016) demonstrated how radial basis functions can be used to help define salt features using level sets. Further, there is a growing field of literature describing the use of level sets as a means to track boundaries for shape optimization schemes (Li et al. (2010), Lewis et al. (2012), Guo and Hoop (2013), Santosa (1996), Osher and Sethian (1988), Burger (2003)), including in the domain of seismic imaging. Our recent work explored the use of the Hessian of a full-grid level set formulation (see Dahlke et al. (2017a)), showing promise as a means to find search directions for a FWI-type work flow. This approach included the inversion of a Newton system that uses a full Hessian formulation of the level set-FWI objective function. For large problems like we find in typical 3D seismic imaging, the large number of model parameters makes inverting this system prohibitively expensive. We introduce a sparse parameterization of the level set problem, which allows us to invert a Newton system that is based on roughly two-orders of magnitude fewer model parameters than before. This speeds up the convergence of iterative methods like conjugate gradient. In this work, we derive this formulation, and then demonstrate on 2D synthetic models using the Gauss-Newton Hessian approximation. After this, we demonstrate using the full Hessian and discuss the practical limitations and potential benefits of that approach.

DERIVATION

Shape optimization

The first step of this derivation is to describe the model space that we are working with. We will call our velocity model m , which we define as:

$$m(\phi_{i,j}, b_{i,j}) = H(\phi_{i,j})(c_{\text{salt}} - b_{i,j}) + b_{i,j}, \quad (1)$$

where $H(\circ)$ is the Heaviside function, $m(\phi_{i,j}, b_{i,j})$ is the velocity value, $\phi_{i,j}$ is the implicit surface value, and $b_{i,j}$ is the background velocity value at 2D spatial position (i, j) . We generalize these parameters for the entire spatial domain (ignoring i, j), and expand this definition with a Taylor series as:

$$m_1 = m_0 + \left. \frac{\partial m}{\partial \phi} \right|_{m_0} \Delta \phi + \left. \frac{\partial m}{\partial b} \right|_{m_0} \Delta b + \dots \quad (2)$$

This approximation is only valid when the Taylor Series we describe converges with the addition of increasingly higher order terms. For the Heaviside function, this is not the case, since the function is not differentiable in original form. For this reason, we must use a smoothed approximation of the Heaviside function, such as:

$$H(\phi) \approx \frac{1}{2} \left[1 + \frac{2}{\pi} \arctan\left(\frac{\pi\phi}{\epsilon}\right) \right].$$

By truncating the series in equation 2 and ignoring higher order terms, we can create a linear approximation to this smooth approximation for the perturbation of the velocity model m with respect to ϕ and b :

$$\Delta m \approx \frac{\partial m(\phi_o, b_o)}{\partial \phi} \Delta \phi + \frac{\partial m(\phi_o, b_o)}{\partial b} \Delta b. \quad (3)$$

This can be written as a matrix operation:

$$\begin{aligned} \Delta m &\approx \begin{bmatrix} \frac{\partial m(\phi_o, b_o)}{\partial \phi} & \frac{\partial m(\phi_o, b_o)}{\partial b} \end{bmatrix} \begin{bmatrix} \Delta \phi \\ \Delta b \end{bmatrix} \\ \Delta m &\approx \begin{bmatrix} \frac{\partial m(\phi_o, b_o)}{\partial \phi} & \frac{\partial m(\phi_o, b_o)}{\partial b} \end{bmatrix} \Delta p, \end{aligned}$$

where we define operator D as:

$$\begin{aligned} D &= \begin{bmatrix} \frac{\partial m(\phi_o, b_o)}{\partial \phi} & \frac{\partial m(\phi_o, b_o)}{\partial b} \end{bmatrix} \\ &= [\delta(\phi_o)(c_s - b) \quad 1 - H(\phi_o)]. \end{aligned} \quad (4)$$

This operator D ultimately scales and masks the parameter fields $\Delta \phi$ and Δb . With this new approximation of the perturbation in our velocity model, the application of our Born operator to our new model parameter space $\Delta p = [\Delta \phi \quad \Delta b]^T$ is given by:

$$\Delta d \approx BD\Delta p.$$

Alternatively, we can find the update gradient for our model parameters by applying the adjoint operation:

$$\Delta p \approx D^T B^T \Delta d.$$

Similarly we can find the application of the Hessian to the search direction as:

$$D^T H D \Delta p \approx -D^T B^T \Delta d. \quad (5)$$

In equation 5, we can substitute H with either the full or Gauss-Newton Hessian. Previous work by Fichtner (2010) shows that the full Hessian of the FWI objective function can be constructed by summing a WEMVA component with the Gauss-Newton component of the Hessian. It is this formulation of the full Hessian application that we use. The method we propose solves equation 5 for Δp using a conjugate gradient algorithm.

Sparsifying with Radial Basis Functions

The thesis of the work done in Kadu et al. (2016) is to replace a regular grid parametrization of the implicit surface ϕ with a surface described as an aggregate of many RBFs, resulting in a much sparser model. We build upon this idea by clustering the spatial locations of the radial basis functions around the areas we expect to see updating occur. This allows us to use far fewer RBF parameters to attain a higher resolution around the salt boundary than we would if we used the regular gridding described in Kadu et al. (2016):

$$\phi(\lambda; \epsilon, r) = \sum_i^{N\lambda} \lambda_i \exp^{-(\epsilon r)^2} \quad (6)$$

where λ is the new model parameter, r is the radial distance from the RBF center i , and ϵ controls the sharpness of the RBF taper (constant). Further, we limit the support of each RBF to improve the efficiency of computing the implicit surface ϕ built from the aggregated RBFs. These details and more are described in Dahlke et al. (2017b). In regards to this work however, our operator D must be modified to account for this additional linear transformation:

$$\begin{aligned} D &= \begin{bmatrix} \frac{\partial m(\phi_o, b_o)}{\partial \phi} \frac{\partial \phi}{\partial \lambda} & \frac{\partial m(\phi_o, b_o)}{\partial b} \end{bmatrix} \\ &= \begin{bmatrix} \delta(\phi_o)(c_s - b) \exp^{-(\epsilon r)^2} & 1 - H(\phi_o) \end{bmatrix} \end{aligned} \quad (7)$$

Further, our model space has also changed to be:

$$\Delta p = \begin{bmatrix} \Delta \lambda \\ \Delta b \end{bmatrix}.$$

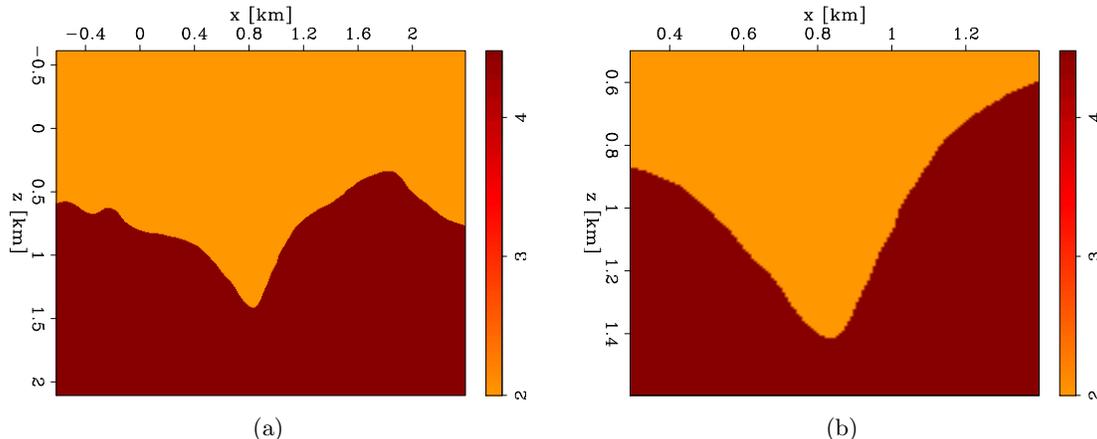


Figure 1: The full model (a) used for propagation (acquisition geometry from (0,0) to (0,1.75)). A close up of the canyon area (b). [ER] `taylor2/. sigsbee-full,single-guess`

APPLICATION TO THE SIGSBEE CANYON MODEL

Single canyon perturbation

For the first application example, we select a portion of the upper Sigsbee canyon model (Figure 1(b)). We perturb the left hand side of the canyon (Figure 2) so that we can get secondary scattering against the opposite canyon wall (shown in Figure 3). It is this secondary scattering that the full Hessian is expected to be able to recover, as opposed to the Gauss-Newton Hessian approximation, which relies on only first-order Born scattering. We use this model because it should be able to offer a comparison between the two Hessian formulations of reasonable significance. For this model we use an acquisition geometry of 38 shots evenly spaced, with 230 receivers. We used a Ricker wavelet with a central frequency of 15 Hz. For this case as well as the double perturbation case, we assume that $\Delta b = 0$, and so invert for a model defined as $\Delta p = \Delta \lambda$.

Double canyon perturbation

For the second application example, we use the same true model based on the upper Sigsbee salt (Figure 1(b)). In this case, we perturb both the left and right hand sides of the canyon (Figure 7). This will offer a further complexity to the secondary scattering of the model (shown in Figure 8). The same acquisition geometry and wavelet were used for this example as the first model.

DISCUSSION ON THE FULL HESSIAN

Benefits

The Gauss-Newton Hessian approximation is based only on the first-order scattering that the Born operator captures. This is limiting in cases where secondary scattering is more

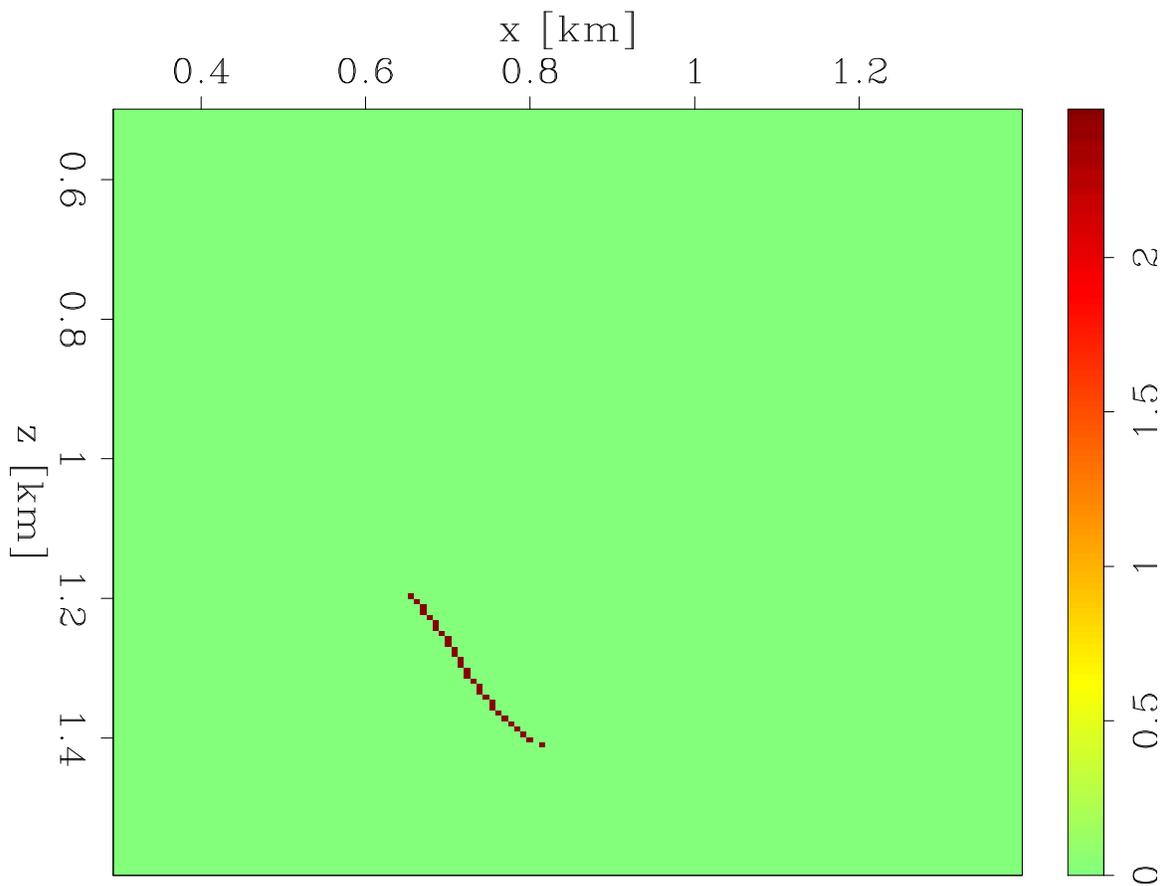


Figure 2: The single canyon perturbation of the Sigsbee model. [ER] `taylor2/. single-pert`

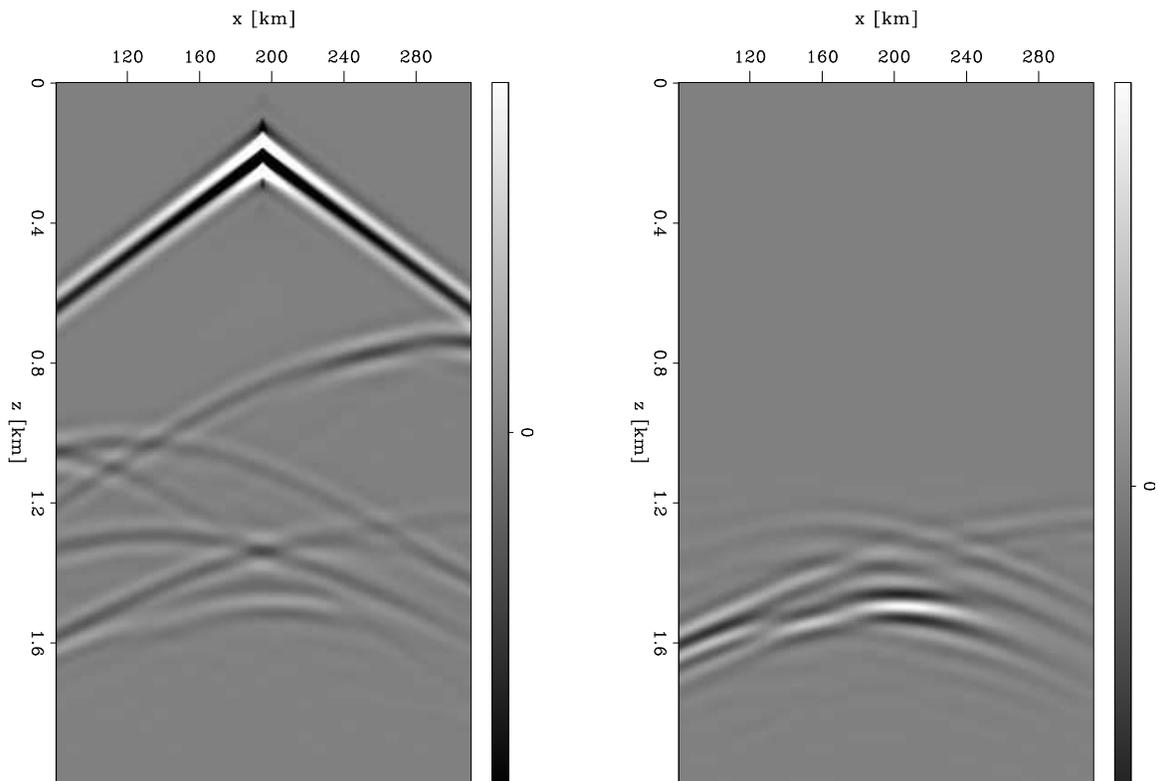


Figure 3: The data generated from the center shot on the true model (left). The center shot residual between the true data and the data generated from an initial guess that had a single canyon side perturbation (right). [CR] `taylor2/. centershot-analysis-single`

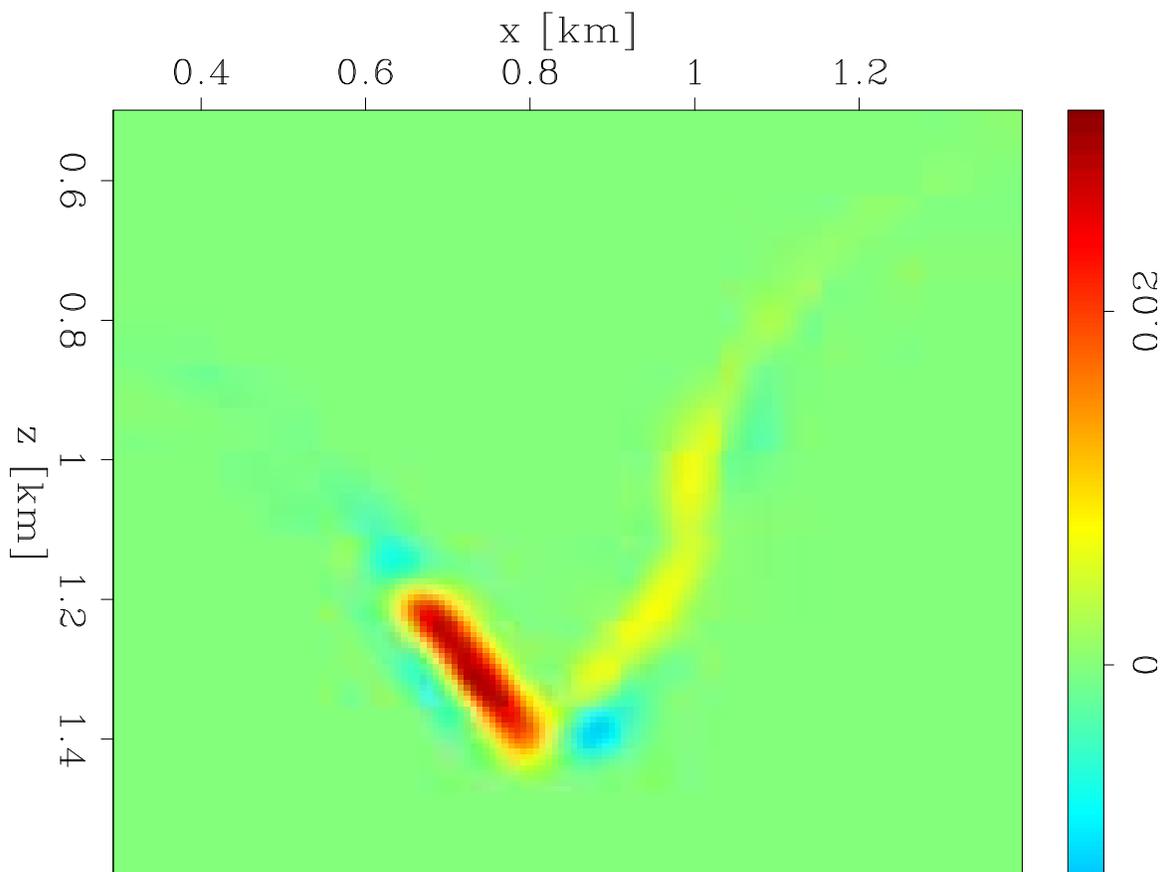


Figure 4: The inversion result after using the Gauss-Newton approximation of the Hessian on the single perturbation model. [CR] `taylor2/. single-final-gn`

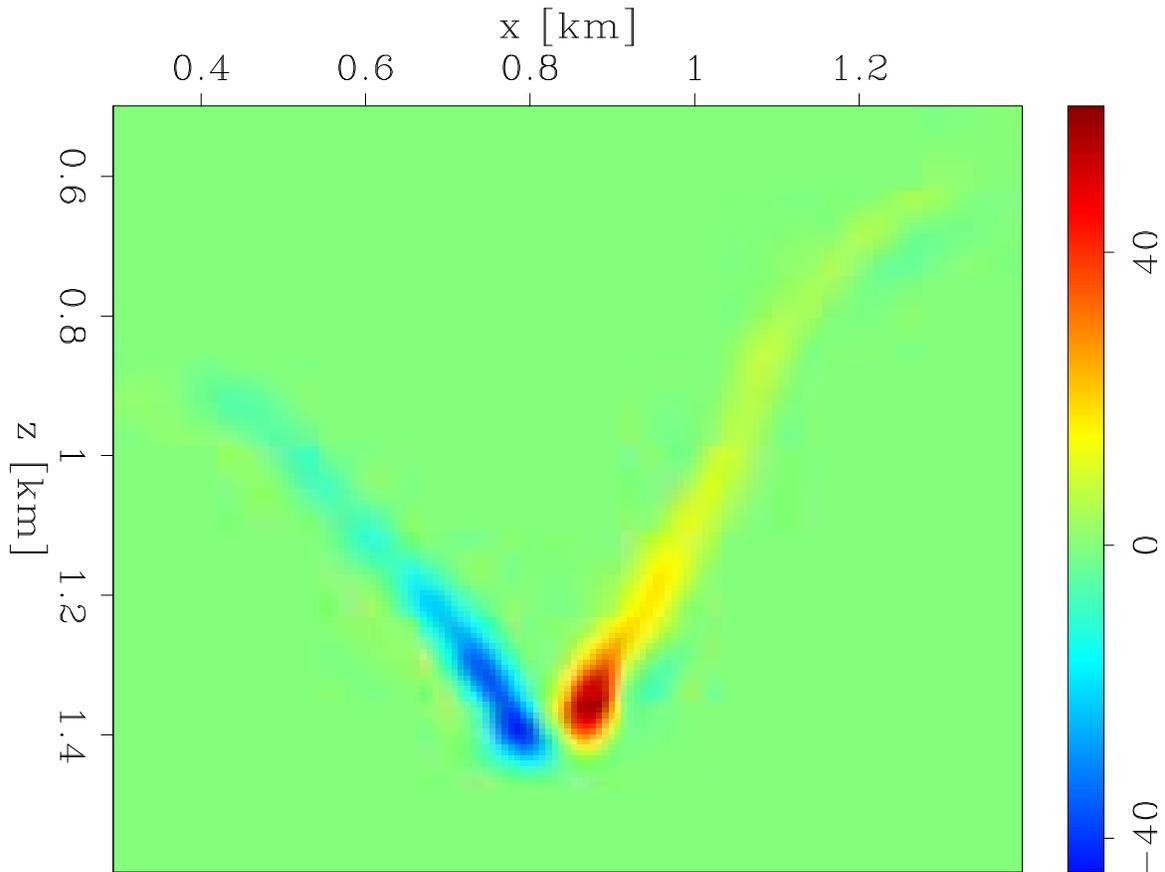


Figure 5: The inversion result after using the full Hessian on the single perturbation model.

[CR] `taylor2/. single-final-full`

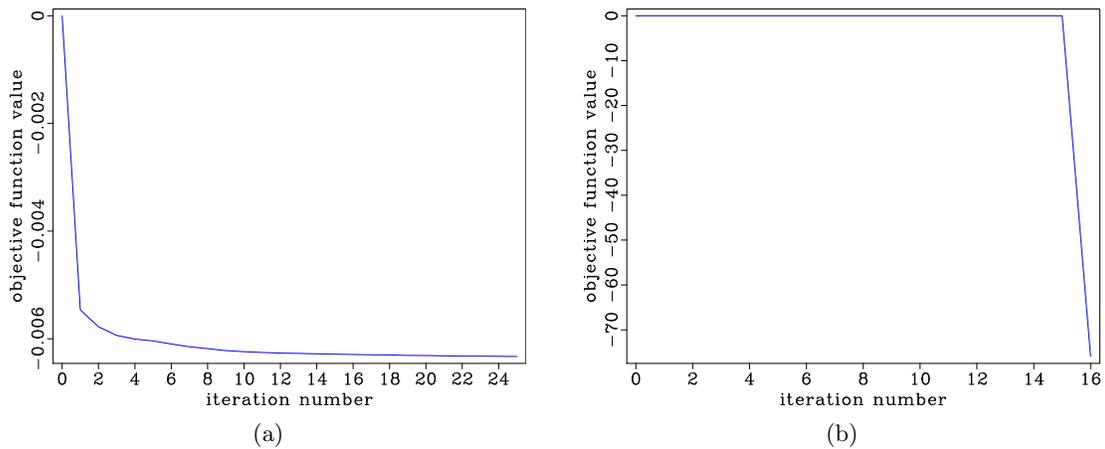


Figure 6: The objective functions from the inversions of the single canyon perturbation model using the Gauss-Newton Hessian (a), and the full Hessian (b).

[CR] `taylor2/. objfunc-single-gn,objfunc-single-full`

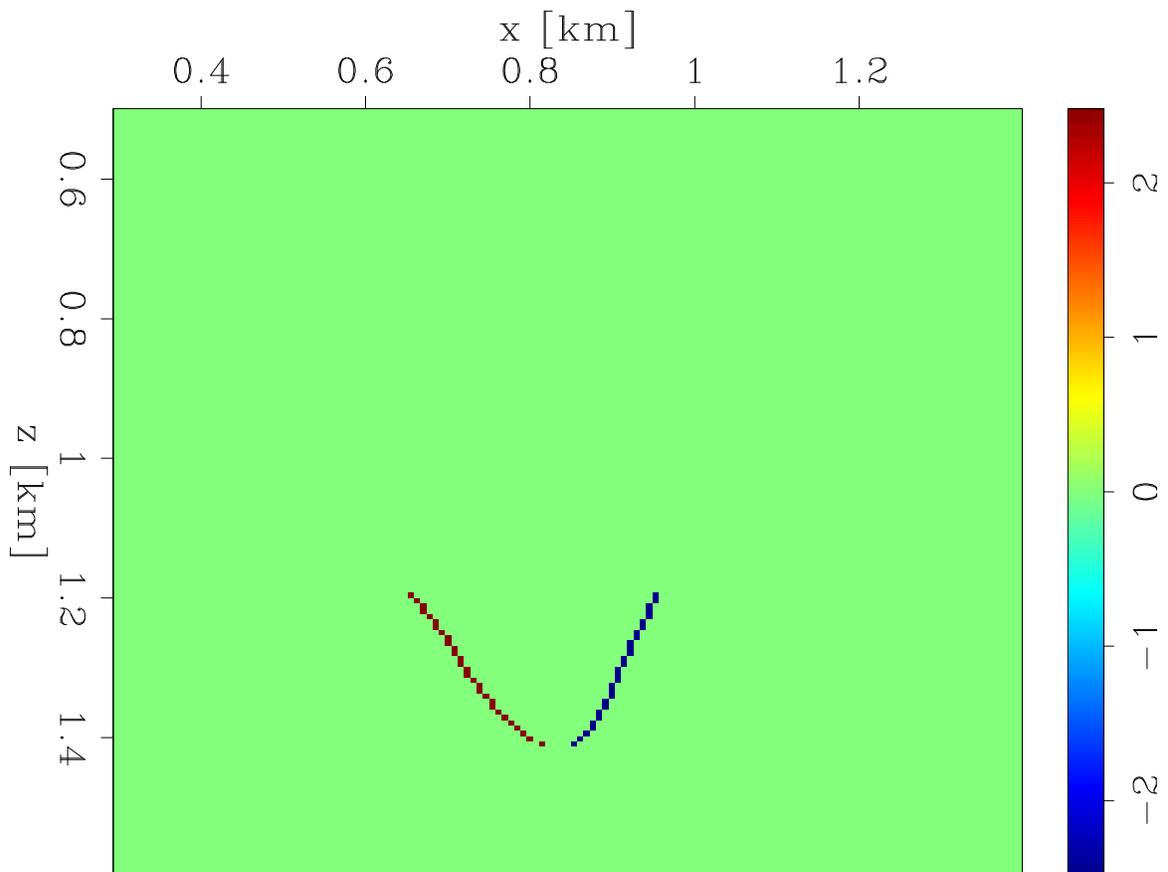


Figure 7: The double canyon perturbation of the Sigsbee model. [ER]
taylor2/. double-pert

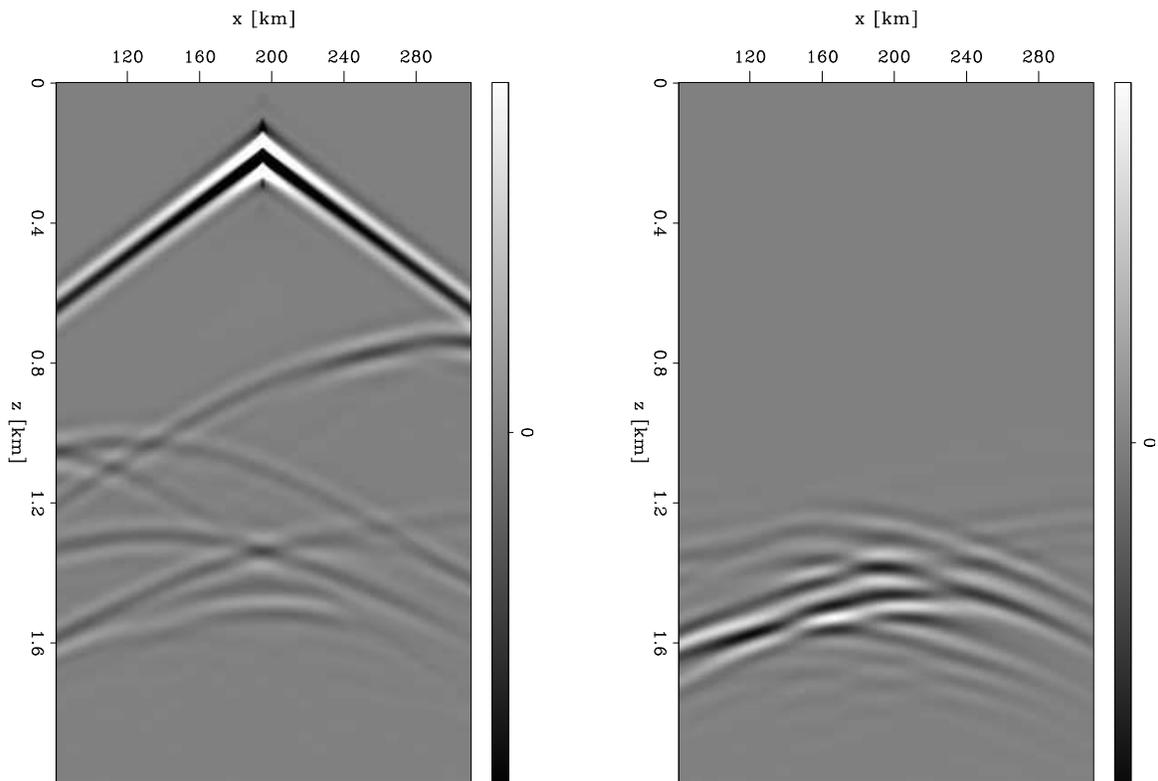


Figure 8: The data generated from the center shot on the true model (left). The center shot residual between the true data and the data generated from an initial guess that had a double canyon side perturbation (right). [CR] `taylor2/. centershot-analysis-double`

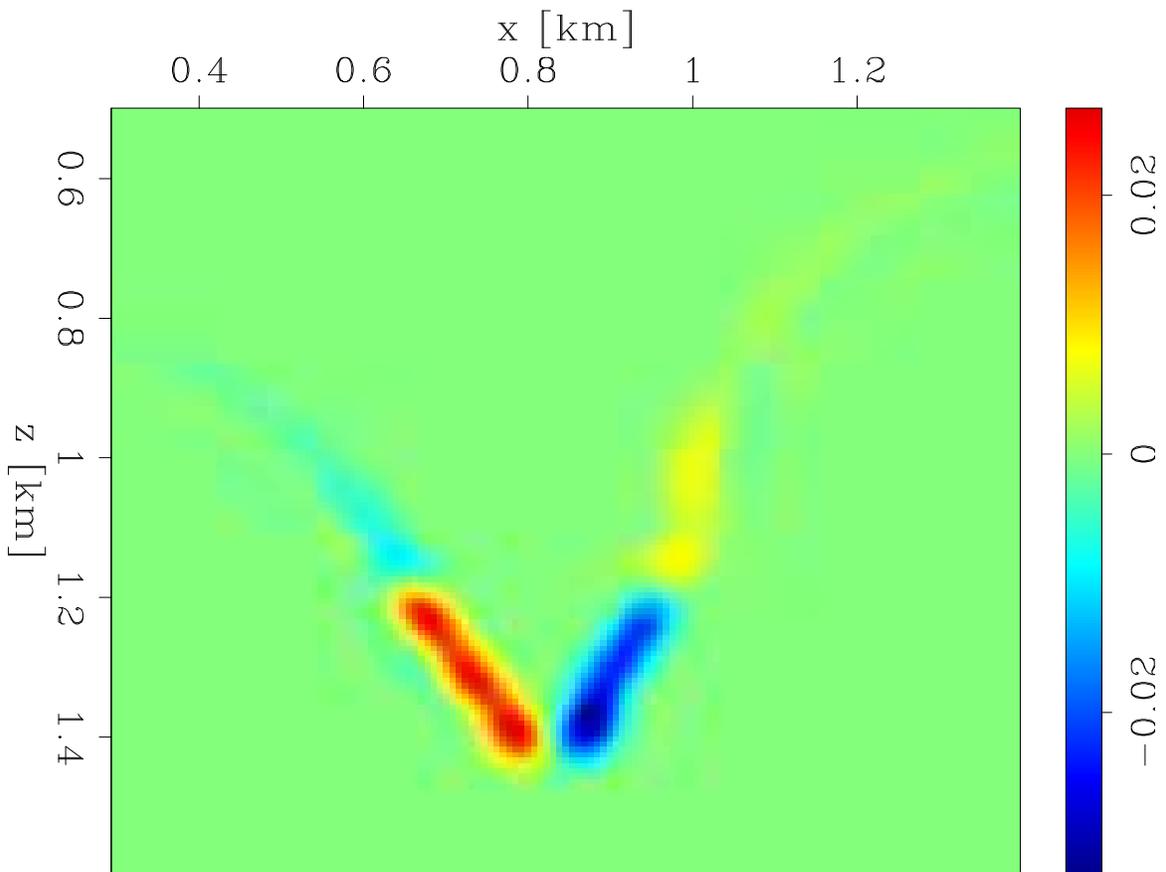


Figure 9: The inversion result after using the Gauss-Newton approximation of the Hessian on the double perturbation model. [CR] `taylor2/. double-final-gn`

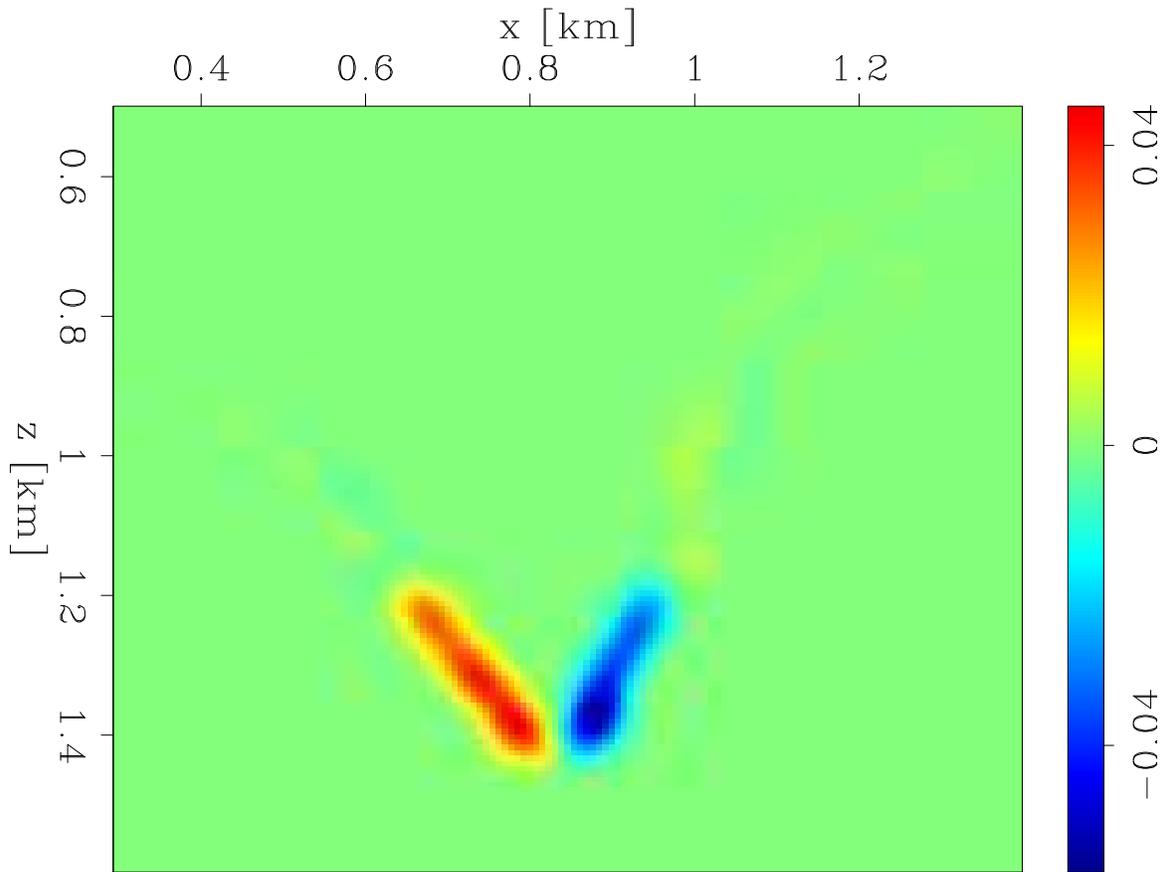


Figure 10: The inversion result after using the full Hessian on the double perturbation model. [CR] `taylor2/. double-final-full`

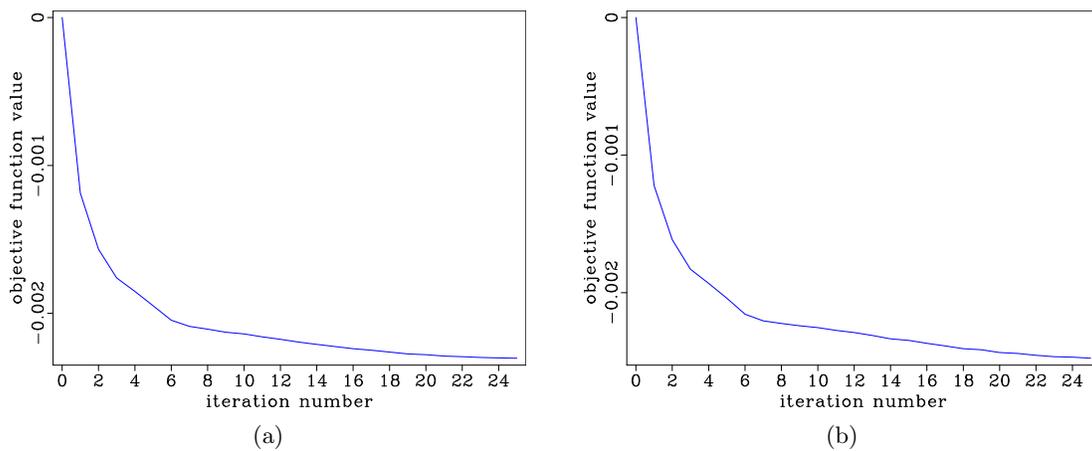


Figure 11: The objective functions from the inversions of the double canyon perturbation model using the Gauss-Newton Hessian (a), and the full Hessian (b). [CR] `taylor2/. objfunc-double-gn,objfunc-double-full`

prominent, such as in salt canyons. The advantage of using the full Hessian is that its application incorporates a WEMVA term that accounts for this second order scattering. When we look at the results from the double canyon perturbation model and compare the Gauss-Newton (Figure 10) and the full Hessian results (Figure 9), we can see a slight improvement in the focusing of the energy in the full Hessian results. This improved search direction should lead to better convergence in the non-linear inversion scheme as well.

Limitations

However, we find that this improvement is not found for all models, since the Hessian is model dependent. The single perturbation example results are much different. While the Gauss-Newton Hessian system inverts quite nicely (Figures 4 and 6(a)), the full Hessian inversion explodes part way through (Figures 5 and 6(b)). Because the full Hessian operator is not inherently positive semi-definite like the Gauss-Newton Hessian is, it is possible that the operator has negative eigenvalues, which can lead to instability during inversion. This was the case in the single canyon perturbation example.

There are a number of ways this can be alleviated. One standard method is to use the Levenberg-Marquardt method Weisstein (2017) of regularizing the operator with a scaled identity matrix. However, in order to use this method properly, the correct scaling of the identity matrix must be used. If too large of a scaling is used, the operator becomes more like the identity matrix, negating the potential benefit of inverting the full Hessian system to begin with. If the scaling is too small, the system will still be ill-conditioned. The ideal scaling is slightly more than the value of the most negative eigenvalue of the operator. This makes the operator positive definite. Since our model (and as a result, our Hessian) is very large, it is impractical to store or factorize the Hessian matrix to determine the most negative eigenvalue through traditional non-iterative linear algebra methods.

Power Iteration Method

The most practical way to find the best scaling is by using the power iteration method outlined in Larson (2012) to find the maximum absolute-valued eigenvalue (positive in the case shown for Figure 12). After this has been found, we shift the operator by the negative of this value to find a new maximum absolute-valued eigenvalue. The difference between this value and the first one derived is the magnitude of the most negative eigenvalue. We experimented with this method, but found the results of this effort to be minimal, and at notable computational cost. Figure 12 shows that in practice at least 30 iterations (and so ~ 30 forward full Hessian operator applications) were necessary for each of the two power iteration searches. Once these searches were complete and a proper shift was found, we found that the results of using this Levenberg-Marquardt shift were almost imperceptible from the Gauss-Newton results. Furthermore, since the Hessian operator is model-dependent (and so changes with each outer loop iteration of FWI), we would need to perform these power iteration steps each time we were to invert the Newton system.

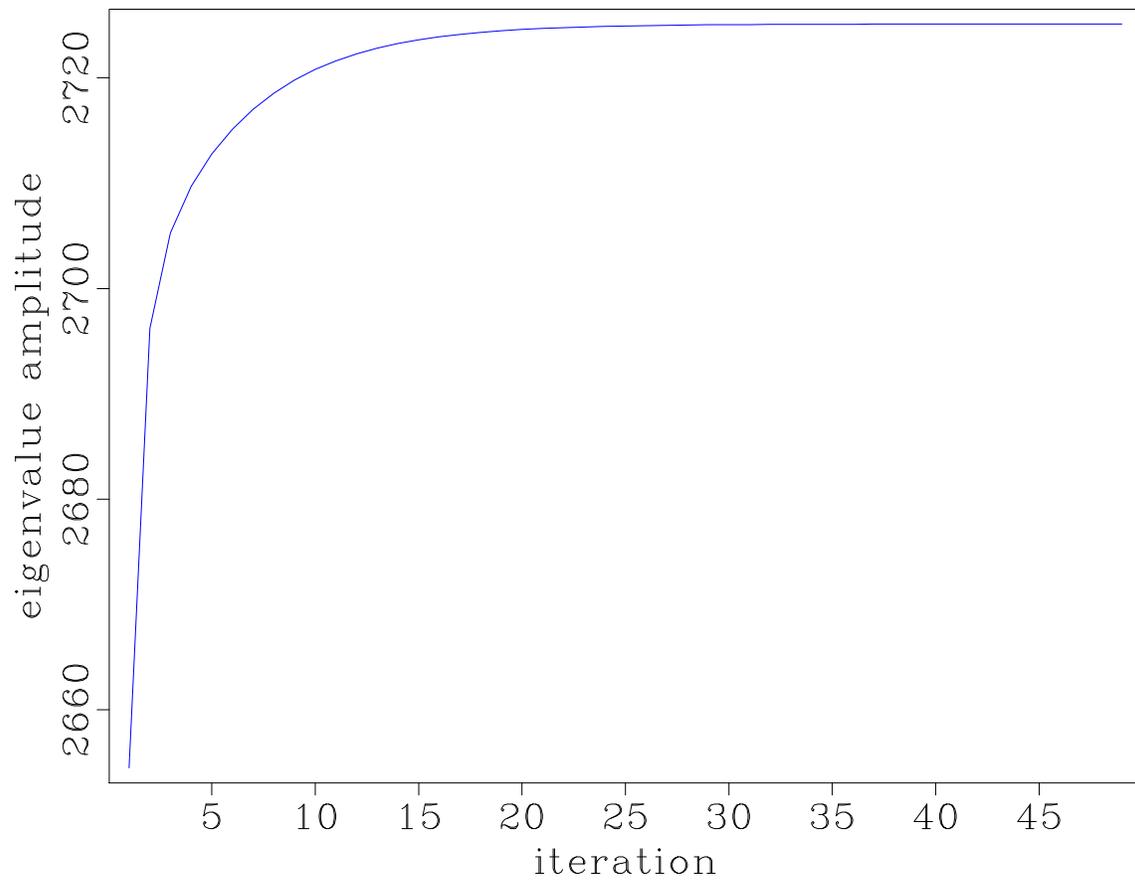


Figure 12: The power iteration curve showing the maximum absolute value approximated eigenvalues of the full Hessian operator used on the single canyon perturbation model.

[CR] `taylor2/. powerit1`

CONCLUSIONS

We successfully invert the Newton system for an objective function that is based on a sparse radial basis function parametrization. We find that the full Hessian formulation for the radial-basis function level sets provides marginal improvement over the Gauss-Newton Hessian for certain models where it inverts stably. However, this improvement is not necessarily worth the effort of ensuring a stable inversion result, which can be very costly for methods like Levenburg-Marquadt combined with power iterations.

REFERENCES

- Burger, M., 2003, A framework for the construction of level set methods for shape optimization and reconstruction: Interfaces and Free boundaries, **5**, 301–330.
- Dahlke, T., B. Biondi, and R. Clapp, 2017a, Application of full hessian on level set formulation and parameter inversion: SEP-Report, **168**.
- , 2017b, Representing salt bodies with radial basis functions: SEP-Report, **170**.
- Fichtner, A., 2010, Full seismic waveform modelling and inversion. Advances in Geophysical and Environmental Mechanics and Mathematics: Springer Berlin Heidelberg.
- Guo, Z. and M. V. D. Hoop, 2013, Shape optimization and level set method in full waveform inversion with 3d body reconstruction: SEG Technical Program Expanded Abstracts, 1079–1083.
- Kadu, A., T. van Leeuwen, and W. A. Mulder, 2016, Salt reconstruction in full waveform inversion with a parametric level-set method: CoRR, **abs/1610.00251**.
- Larson, R., 2012, Elementary linear algebra: Cengage Learning.
- Lewis, W., B. Starr, D. Vigh, et al., 2012, A level set approach to salt geometry inversion in full-waveform inversion: 2012 SEG Annual Meeting.
- Li, C., C. Xu, C. Gui, and M. Fox, 2010, Distance regularized level set evolution and its application to image segmentation: Image Processing, IEEE Transactions on, **19**, 3243–3254.
- Osher, S. and J. A. Sethian, 1988, Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations: Journal of computational physics, **79**, 12–49.
- Santosa, F., 1996, A level-set approach for inverse problems involving obstacles: ESAIM Controle Optim. Calc. Var, **1**, 17–33.
- Weisstein, E. W., 2017, Levenberg-marquardt method: From MathWorld—A Wolfram Web Resource.

Implementing Wave-Equation Migration Velocity Analysis Within Linearized Waveform Inversion with Velocity Updating: Considerations and Challenges

Alejandro Cabrales-Vargas

ABSTRACT

Wave-Equation Migration Velocity Analysis is one of the fundamental processes for performing Linearized Waveform Inversion with Velocity Updating, and also the most computationally intense. We recently proposed the implementation of the former by means of employing Random Boundary Conditions for storage alleviation, at the cost of performing extra wavefield propagations. We show the result of this implementation. However, the scattered source wavefield and the scattered receiver wavefield depend on the direction of propagation of the corresponding wavefields that originate them. Therefore, the source wavefield must propagate forward in time when scattering. Likewise, the receiver wavefield must propagate backward in time when scattering. This restriction leads to the fact that we require twelve propagations per iteration plus one, instead of the eight iterations plus one that we had initially expected. Additionally, Random Boundary Conditions can introduce random noise that could potentially harm the inversion result if they are not properly implemented.

INTRODUCTION

Some colleagues and I recently proposed Linearized Waveform Inversion with Velocity Updating (LWIVU) (Cabrales-Vargas et al., 2016a,b, 2017) as a new inversion technique aimed at improving the subsurface reflectivity, which allows the subsurface velocity model to vary. Such variability is not intended to correct reflector positioning, but to improve amplitudes affected by the accumulated effect of inaccuracies in velocity or slowness, therefore yielding more confidence in the estimation of the subsurface reflectivity. Cabrales-Vargas et al. (2017) discuss some aspects of the LWIVU processing components, such as the Gauss-Newton Hessian construction by means of point-spread functions. Gauss-Newton Hessian can be precomputed, stored, and applied “on the fly,” interpolating as needed. On the contrary, Wave-Equation Migration Velocity Analysis (WEMVA) (Biondi and Sava, 1999; Biondi, 2006) has to be performed twice at each iteration during the inversion. Cabrales-Vargas et al. (2017) propose the use of Random Boundary Conditions (RBC) (Clapp, 2009) in WEMVA to prevent saving wavefields in disk and the corresponding I/O access. In Reverse-Time Migration (RTM), the price of using such RBC is an extra propagation of the source wavefield. On the same grounds, the estimated number of wavefield propagations within LWIVU was four for each WEMVA step (Cabrales-Vargas et al., 2016a,b, 2017).

However, I have found that a single application of WEMVA demands seven propagations when implemented with RBC, not five (counting an initial propagation of the source wavefield). It signifies twelve propagations per LWIVU iteration instead of eight (WEMVA

is performed twice per iteration.) In the first section of this report I discuss the reason for the additional propagations. Next, I implement the WEMVA operator in a simple two-layer model to verify the effects of the RBC, and an alternative implementation using Energy Imaging Conditions (EIC) (Rocha et al., 2016).

WAVE-EQUATION MIGRATION VELOCITY ANALYSIS WITH RANDOM BOUNDARY CONDITIONS

The WEMVA process represents a linear operator that maps perturbations in the slowness squared field into perturbations in the migrated image. If the process is performed using zero subsurface offset, the WEMVA operator is self adjoint; thus, the same operator retrieves a perturbation in slowness squared from a perturbation in the image.

The WEMVA process can be split into the following steps:

- Forward propagation of the source wavefield in background slowness field
- Backward propagation of the receiver wavefield in background slowness field
- Scattering of the source wavefield upon the perturbation in the image or in the background model
- Scattering of the receiver wavefield upon the perturbation in the image or in the background model
- Zero-lag time cross-correlation of the source wavefield and scattered receiver wavefield
- Zero-lag time cross-correlation of the receiver wavefield and scattered source wavefield

Let us assume that we can store disk the propagated wavefields. We can execute the WEMVA process as shown in Algorithm 1. This procedure demands four propagations

Algorithm 1 WEMVA implementation saving both source and receiver wavefields

- Forward *propagate* the source wavefield and store; then, scatter upon perturbation and forward *propagate* the scattered source wavefield.
 - Backward *propagate* the receiver wavefield and store; then, scatter upon perturbation and backward *propagate* the scattered receiver wavefield.
 - Perform cross-correlations.
-

(indicated in italics): two propagations in the background model, and two propagations after scattering. Notice that only the source and the receiver wavefields need to be stored, not the scattered wavefields.

Now let us assume that we can only store one wavefield. We begin with the source wavefield for simplicity. In this case we can proceed as indicated in Algorithm 2. Notice that now we need to perform two extra propagations compared to the previous case. This is the price the we must pay for preventing the storage of more than one wavefield at a time.

Algorithm 2 WEMVA implementation storing one wavefield at a time

- Forward *propagate* the source wavefield and store it.
 - Backward *propagate* the receiver wavefield and scatter upon perturbation; then, backward *propagate* the scattered receiver wavefield “on the fly.”
 - Crosscorrelate the scattered receiver wavefield with the stored source wavefield as the former is propagated.
 - Delete the source wavefield.
 - Backward *propagate* the receiver wavefield and store it.
 - Forward *propagate* the source wavefield and scatter upon perturbation; then, forward *propagate* the scattered source wavefield “on the fly.”
 - Crosscorrelate the scattered source wavefield with the stored receiver wavefield as the former is propagated.
-

We can go further and prevent the storage of the propagated wavefields whatsoever using RBC to ensure the reversibility of propagations, similar to the RTM case (Clapp, 2009). In such a case, we proceed as indicated in Algorithm 3. Notice that by using RBC

Algorithm 3 WEMVA implementation storing none of the wavefields (using RBC)

- Forward *propagate* the source wavefield; then save the last two time frames.
 - Backward *propagate* the receiver wavefield and scatter upon perturbation; then, backward *propagate* the scattered receiver wavefield “on the fly.”
 - At the same time, backward *repropagate* the source wavefield and cross-correlate with the scattered receiver wavefield. Save the last two frames of the receiver wavefield.
 - Forward *propagate* the source wavefield and scatter upon perturbation; then, forward *propagate* the scattered source wavefield “on the fly.”
 - At the same time, forward *re-propagate* the receiver wavefield and cross-correlate with the scattered source wavefield.
-

we have to pay the price of performing an extra propagation with respect to storing one wavefield, or three with respect to storing both wavefields. This is the implementation that I will employ for the WEMVA step in LWIVU. In the last reports (Cabralés-Vargas et al., 2016a,b, 2017) we had estimated fewer propagations because we had assumed that propagation of the scattered wavefields was independent of the time direction. In other words, we (**incorrectly**) reasoned as indicated in Algorithm 4.

In summary, considering that the LWIVU process demands two WEMVA implementations (one for the gradient in model space and the other for its projection onto the data space), we obtain the following number of propagations for each case:

Algorithm 4 Wrong implementation

- Forward *propagate* the source wavefield; then, save the last two time frames.
 - Backward *propagate* the receiver wavefield and scatter upon perturbation; then, backward *propagate* the scattered receiver wavefield “on the fly.”
 - At the same time, backward *repropagate* the source wavefield and scatter upon perturbation; then, backward propagate the scattered source wavefield. Crosscorrelate corresponding wavefields as they are propagated backwards in time.
-

- Storing both wavefields: Two propagations per iteration + two initial propagations of the source and the receiver wavefields.
- Storing one wavefield: Eight propagations per iteration + one initial propagation of the source wavefield.
- Storing none of the wavefields (using RBC): Twelve propagations per iteration + one initial propagation of the source wavefield.

The significant increase in the number of wavefield propagations when avoiding their storage is because storing prevents their recomputation throughout the process. Only the scattered wavefields are recomputed because the perturbations change as iterations progress. On the contrary, storing none of the wavefields demands recomputation of all wavefields as needed.

SYNTHETIC EXAMPLES

In this section I show the application of the WEMVA operator and the adjoint using a simple two-layer model with a Gaussian positive-velocity anomaly (Figure 1). I first isolate the anomaly to obtain the corresponding perturbation in background slowness squared (Figure 2), which is negative. Then, I apply forward WEMVA and obtain the corresponding perturbation in the image (Figure 3). Notice the presence of the low-wavenumber tomographic component, as well as the virtual absence of random artifacts which can potentially be produced by the RBC. Next, I apply adjoint WEMVA to the perturbation in the image for recovering an approximation to the original perturbation in the background (Figure 4). The perturbation in the image maps back into a shape that resembles the original Gaussian anomaly, although the amplitude is wrong because an inversion process is needed for recovering the original amplitudes. Notice that the aforementioned tomographic component maps onto a reflector resembling the perturbation in the image, which amplitude obscures the approximated anomaly. These results are similar to those obtained with tapering boundary conditions (Cabrales-Vargas et al., 2016b), although I have not yet set forth an inversion in the present case.

I repeat the experiment replacing the cross-correlation imaging conditions (CIC) (Claerbout, 1992) with EIC. Figure 5 shows that the tomographic component has been attenuated in forward WEMVA. Likewise, after applying adjoint WEMVA with EIC the unwanted reflector was virtually removed (Figure 6), although some mild random artifacts can be seen

Figure 1: Two-layer velocity model with Gaussian anomaly. [ER]
[alejandro1/. model](#)

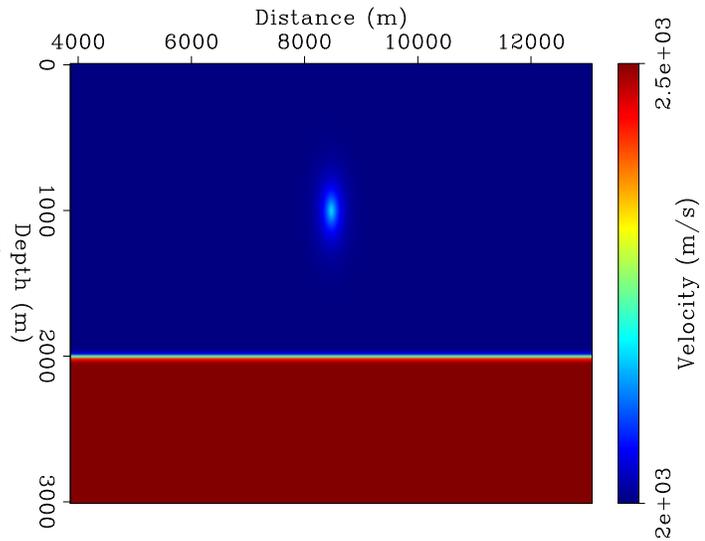


Figure 2: Gaussian anomaly expressed in slowness squared. [ER]
[alejandro1/. gauss](#)

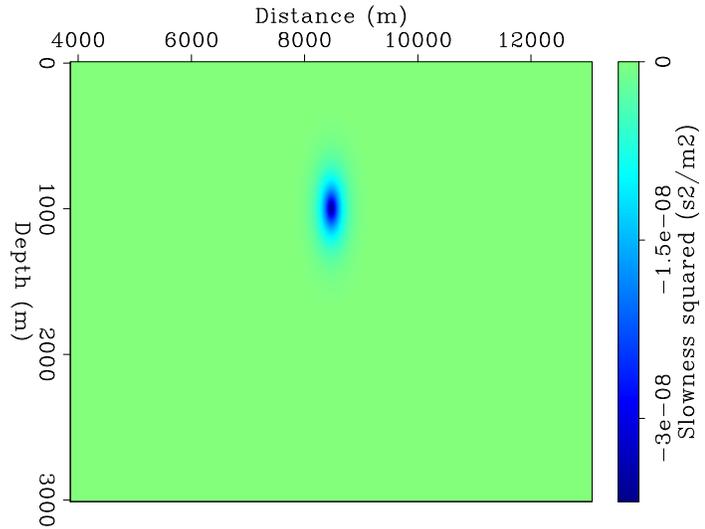


Figure 3: Perturbation in the image after applying forward WEMVA to the Gaussian anomaly. [CR]
[alejandro1/. wemva](#)

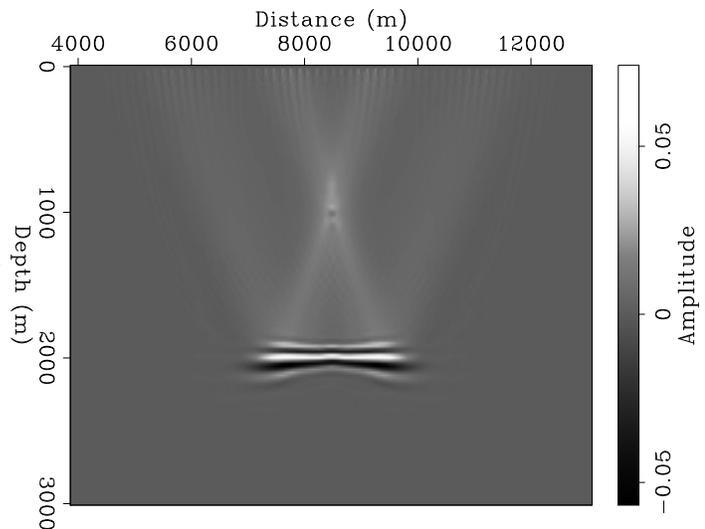
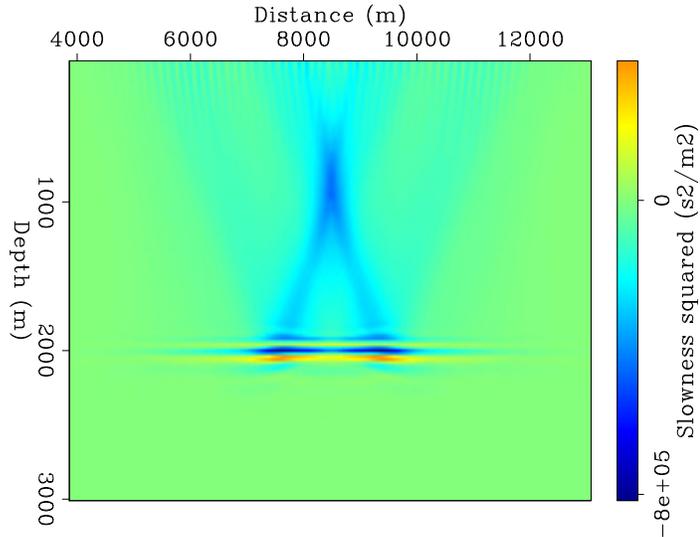


Figure 4: Approximation to the perturbation in the background slowness after applying adjoint WEMVA to the perturbation in the image (Figure 3).

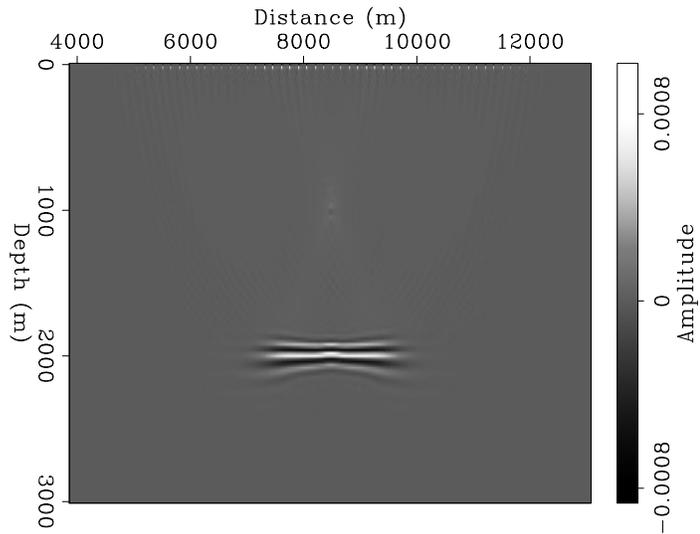
[CR] [alejandrol/. wemva-adj](#)



now that the reflector's amplitude no longer obscures the estimated anomaly. This separation of tomographic and reflectivity components can potentially help in producing more reliable results during the inversion. Nonetheless, my implementation of WEMVA using EIC has not passed the dot-product test yet.

Figure 5: Perturbation in the image after applying forward WEMVA to the Gaussian anomaly using EIC.

[CR] [alejandrol/. wemva-eic](#)



CONCLUSIONS

I rectified the number of propagations by iteration that LWIVU requires when the WEMVA step is performed with RBC. The restriction of time directionality during the computation of the scattered wavefields makes the number of propagations rise to seven for a single application of WEMVA, and twelve per iteration plus one within LWIVU, where two WEMVA applications are required.

Using RBC does not introduce significant random artifacts to the WEMVA results. The

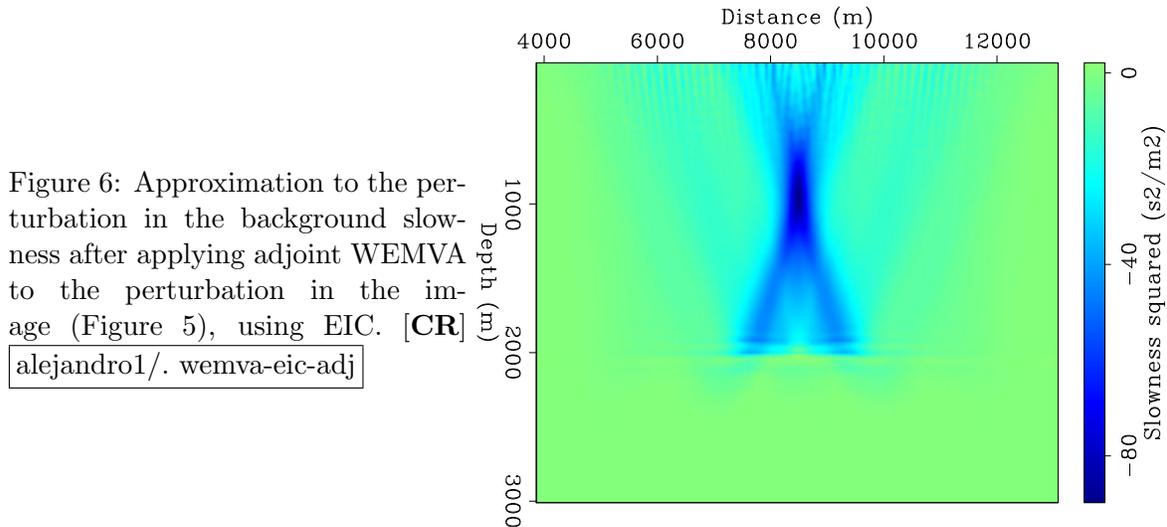


Figure 6: Approximation to the perturbation in the background slowness after applying adjoint WEMVA to the perturbation in the image (Figure 5), using EIC. [CR] [alejandro1/. wemva-eic-adj](https://doi.org/10.1190/1.5111111)

incorporation of EIC attenuates both the tomographic component in the perturbation in the image and the reflectivity component in the perturbation in the background model. I still have to revisit this variation to verify whether it passes the dot-product test.

ACKNOWLEDGEMENT

Alejandro Cabrales would like to thank Petróleos Mexicanos for financing his PhD studies.

REFERENCES

- Biondi, B., 2006, 3-D seismic imaging: Society of Exploration Geophysicists.
- Biondi, B. and P. Sava, 1999, Wave-equation migration velocity analysis: SEG Technical Program Expanded Abstracts, 1723–1726.
- Cabrales-Vargas, A., B. Biondi, and R. Clapp, 2016a, Linearized waveform inversion with (small) velocity updating: SEP-Report, **163**, 189–196.
- , 2016b, Linearized Waveform Inversion with Velocity Updating: Theory and first results: SEP-Report, **165**, 63–92.
- , 2017, Our progress towards linearized waveform inversion with velocity updating (LWIVU): SEP-Report, **168**, 151–162.
- Claerbout, J. F., 1992, Earth soundings analysis: Processing versus inversion: Blackwell Scientific Publications, Inc.
- Clapp, R., 2009, Reverse time migration with random boundaries: SEG Technical Program Expanded Abstracts, 2809–2813.
- Rocha, D., N. Tanushev, and P. Sava, 2016, Acoustic wavefield imaging using the energy norm: Geophysics, **81**, no. 4, S151–S163.

A 2D Helmholtz equation solver library based on C++ and SuiteSparse

Rahul Sarkar and Biondo Biondi

ABSTRACT

We developed a 2D Helmholtz equation solver library in C++ based on the SuiteSparse library for sparse linear algebra. The solver library was successfully tested for correctness in this paper by running a suite of simple 2D examples. We first used the library to solve the Helmholtz equation in a homogenous medium for different frequency bands. The test was then repeated for an inhomogenous two layer medium, for the same frequency bands. We also performed Born linearization tests with the code and demonstrated that the error is quadratic in the magnitude of the velocity perturbation, in agreement with the theoretical prediction. Finally we performed extended linearized Born forward modeling tests using the Helmholtz code to demonstrate its use in Tomographic Full Waveform Inversion (TFWI) applications.

INTRODUCTION

Solving the wave equation in the frequency domain has many advantages from the perspective of seismic inversion. In particular, this is useful if we have a large number of shots, in which case the cost of pre-factoring and storing the LU decomposition of the matrix arising in the numerical solution of the Helmholtz equation is justified as it has to be performed only once for each frequency, and then the factors can be reused for all the shots. The solution to the Helmholtz equation is then obtained by performing a forward and backward solve for each shot. This idea also extends naturally to the TFWI setting introduced by Biondi and Almomin (2014), where one needs to solve the wave equation to get the background wave field which is then convolved with the extended model to get the secondary source to be propagated using Born modeling. In fact, one of the main motivations for this work is to be able to perform 3D TFWI at low frequencies below 5Hz, which can fit in present day computer memory for reasonably large model sizes. In addition to this, the convolution operation in time domain TFWI becomes a multiplication operation in the frequency domain. The same is also true of the gradient computation for certain objective functions such as in the case of FWI.

In this paper, we follow the discretization scheme introduced by Liu and Ying (2016). The code that we have developed can handle three kinds of boundary conditions: Dirichlet, Neumann and Sommerfeld radiation through a PML (perfectly matched layer) approach as described in Liu and Ying (2016). The PML is implemented to have the same amount of attenuation across different frequency bands, and is also independent of the number of PML cells. We have implemented the Helmholtz solver in C++ using the SuiteSparse library (Davis, 2004, 2006) for performing the sparse factorization.

THEORY

In this section, we provide a very brief introduction to the Helmholtz equation, and outline the numerical scheme that we use to solve it in this paper. We establish the connection of the Helmholtz equation with the acoustic wave equation, derive formulas for the Born linearization of the Helmholtz equation, and also derive the frequency domain formulation of TFWI extended Born forward modeling.

The Helmholtz equation

The Helmholtz equation in d dimensions is given by

$$\left(\nabla^2 + \frac{\omega^2}{c^2(\mathbf{x})} \right) \hat{u}(\mathbf{x}, \omega) = \hat{f}(\mathbf{x}, \omega), \quad (1)$$

where $\mathbf{x} \in \mathbb{R}^d$, $\nabla^2 = \sum_{i=1}^d \frac{\partial^2}{\partial x_i^2}$ represents the d -dimensional Laplacian operator, $\omega \in \mathbb{R}$ is the angular frequency, $c(\mathbf{x})$ is the velocity of the medium that depends on the position \mathbf{x} , $\hat{f}(\mathbf{x}, \omega)$ is the forcing function, and $\hat{u}(\mathbf{x}, \omega)$ is the solution to the Helmholtz equation. For purposes of this paper, we will be only concerned with $d = 2$, i.e we will only consider the 2D Helmholtz equation.

Typically, the Helmholtz equation is solved on a bounded domain $\mathbf{\Gamma}$ with suitable boundary conditions on the boundary of the domain $\mathbf{d}\mathbf{\Gamma}$, which forms a submanifold of $\mathbf{\Gamma}$ of codimension 1. Three types of boundary conditions are extremely common, namely *DBC* (*Dirichlet Boundary Condition*), *NBC* (*Neumann Boundary Condition*) and *RBC* (*Robin Boundary Condition*). These boundary conditions have the following analytical form

$$\begin{aligned} \hat{u}(\mathbf{x}, \omega) \Big|_{\mathbf{d}\mathbf{\Gamma}} &= \hat{u}_D(\mathbf{x}, \omega) & (DBC) \\ \frac{\partial \hat{u}(\mathbf{x}, \omega)}{\partial \mathbf{n}} \Big|_{\mathbf{d}\mathbf{\Gamma}} &= \hat{u}_N(\mathbf{x}, \omega) & (NBC) \\ \left(a\hat{u}(\mathbf{x}, \omega) + b \frac{\partial \hat{u}(\mathbf{x}, \omega)}{\partial \mathbf{n}} \right) \Big|_{\mathbf{d}\mathbf{\Gamma}} &= \hat{u}_R(\mathbf{x}, \omega) & (RBC), \end{aligned} \quad (2)$$

where $\frac{\partial}{\partial \mathbf{n}}$ is the normal derivative operator pointing outward with respect to the boundary $\mathbf{d}\mathbf{\Gamma}$.

However in many cases such as commonly encountered in frequency domain seismic inversion, one wishes to solve the Helmholtz equation on an unbounded domain. But even in such applications, the forcing function (also known as the “source”) is still supported on a bounded domain, and one is interested in the solutions representing outgoing waves instead of the solutions representing incoming waves, i.e one wants the “source” to be a source and not a sink. To enforce this condition, a commonly used boundary condition is the Sommerfeld radiation boundary condition (see Sommerfeld (1949); Johnson (2008) for details), given by

$$\lim_{r \rightarrow \infty} \frac{\partial \hat{u}(\mathbf{x}, \omega)}{\partial \mathbf{r}} - \frac{i\omega}{c_0} \hat{u}(\mathbf{x}, \omega) = 0, \quad (3)$$

where we have assumed that the velocity of the medium is constant outside a bounded set Γ , i.e. $c(\mathbf{x}) \Big|_{\mathbb{R}^d \setminus \Gamma} = c_0$, and $\frac{\partial}{\partial \mathbf{r}}$ denotes the derivative operator along the unit normal in the radial direction. In practice, one has to solve the Helmholtz equation on a truncated domain in a computer, where the Sommerfeld radiation boundary condition is implemented using some form of PML condition, which we discuss next.

Numerical scheme to solve the Helmholtz equation

The numerical scheme implemented for this paper is taken from Liu and Ying (2016). Both directions are treated in a manner similar to one another, and thus we only discuss what is done along the X direction. Suppose our computational domain along the X direction is given by $[0, l]$. Let η denote the PML width, and let us first discuss the case when we have PML boundaries on both ends of the computational domain. Thus we have two PML zones given by $[0, \eta)$ and $(l - \eta, l]$. Following Liu and Ying (2016), we introduce the auxiliary functions

$$\begin{aligned} \sigma(x) &:= \begin{cases} \frac{C}{\eta} \left(\frac{x-\eta}{\eta} \right)^2, & x \in [0, \eta), \\ 0, & x \in [\eta, l - \eta], \\ \frac{C}{\eta} \left(\frac{x-l+\eta}{\eta} \right)^2, & x \in (l - \eta, l], \end{cases} \\ s(x) &:= \left(1 + i \frac{\sigma(x)}{\omega} \right)^{-1}, \end{aligned} \quad (4)$$

where C is a positive constant that does not depend on ω , and η is typically around one wavelength.

With this notation, the Helmholtz equation along X direction with PML is modified to be

$$\begin{aligned} \left(\left(s(x) \frac{\partial}{\partial x} \right)^2 + \frac{\omega^2}{c^2(x)} \right) \hat{u}(x, \omega) = \hat{f}(x, \omega), \quad \forall x \in [0, l], \\ \text{with } \hat{u}(0, \omega) = \hat{u}(l, \omega) = 0, \text{ and } \hat{f}(x, \omega) = 0 \text{ in the PML zone.} \end{aligned} \quad (5)$$

The above equation 5 is discretized using a standard second-order central difference numerical scheme as follows

$$\frac{s_i}{h_x} \left(\frac{s_{i+1/2}}{h_x} (\hat{u}_{i+1} - \hat{u}_i) - \frac{s_{i-1/2}}{h_x} (\hat{u}_i - \hat{u}_{i-1}) \right) + \frac{\omega^2}{c_i^2} \hat{u}_i = \hat{f}_i, \quad \forall 1 \leq i \leq n, \quad (6)$$

where the subscript i means that the corresponding quantity is evaluated at $x = ih_x$, and $h_x = \frac{1}{n+1}$ is the grid size along X. Note that here we have used the same PML width on both sides, but it is possible to use different PML widths also, as long as they are kept to at least one wavelength. Then the discretization scheme gets modified accordingly. Finally one should also note that it is possible to discretize equation 5 using a higher order discretization scheme which is relatively straightforward to perform, but we do not explore this in our paper.

When one has DBC on any side, we set $\eta = 0$ for that side. The case of NBC is similar, and we first set $\eta = 0$ for that side followed by using a central difference scheme for the points

on the boundary. The right hand sides are also modified by the DBC and NBC, depending on the numerical values supplied for the boundary conditions.

Performing the above discretization leads to a matrix equation of the form $\mathbf{A}\hat{\mathbf{u}} = \hat{\mathbf{f}}$. The matrix \mathbf{A} is extremely sparse and has at most 5 non-zero elements per row in the case of the 2D Helmholtz equation. We use SuiteSparse to compute the sparse LU factorization of such a system and then one can solve the same equation for different right hand sides, such as for different shots.

Connection of the Helmholtz equation with the wave equation

To understand how the Helmholtz equation arises in frequency domain seismic inverse problems, let us consider the constant density acoustic wave equation in a medium with heterogenous velocities given by

$$\frac{1}{c^2(\mathbf{x})} \frac{\partial^2 u(\mathbf{x}, t)}{\partial t^2} - \nabla^2 u(\mathbf{x}, t) = -f(\mathbf{x}, t), \quad (7)$$

where $u(\mathbf{x}, t)$ is the solution to the wave equation, $-f(\mathbf{x}, t)$ is the time domain forcing function (also called the “source”) and $c(\mathbf{x})$ denotes the heterogenous velocity field as defined earlier.

To see the connection with the Helmholtz equation, we simply Fourier transform equation 7 in time and we obtain the Helmholtz equation in equation 1, where $\hat{u}(\mathbf{x}, \omega)$ and $\hat{f}(\mathbf{x}, \omega)$ are the Fourier transforms of $u(\mathbf{x}, t)$ and $f(\mathbf{x}, t)$ respectively. Thus the solution to the time domain wave equation is completely determined if we know the solution to the corresponding Helmholtz equation for all $\omega \in \mathbb{R}$, by Fourier duality. In practice, the source $f(\mathbf{x}, t)$ is band limited and thus one only needs to solve the Helmholtz equation for a limited range of frequencies.

Linearization of the Helmholtz equation

We next derive formulas for the Born linearization of the Helmholtz equation, for a fixed angular frequency ω . Let us consider the velocity field $c(\mathbf{x})$ and the corresponding solution to the Helmholtz equation $\hat{u}(\mathbf{x}, \omega)$ appearing in equation 1. We now perturb the velocity field by adding a small perturbation $\delta c(\mathbf{x})$, and so the new velocity is given by $c(\mathbf{x}) + \delta c(\mathbf{x})$. As a result of this perturbation, the solution to the Helmholtz equation also changes and let us denote it by $\hat{u}(\mathbf{x}, \omega) + \delta \hat{u}(\mathbf{x}, \omega)$. These perturbed quantities also satisfy the Helmholtz equation, and so we have

$$\left(\nabla^2 + \frac{\omega^2}{(c(\mathbf{x}) + \delta c(\mathbf{x}))^2} \right) (\hat{u}(\mathbf{x}, \omega) + \delta \hat{u}(\mathbf{x}, \omega)) = \hat{f}(\mathbf{x}, \omega). \quad (8)$$

The Born linearization formula expresses the first order relationship between the perturbed quantities $\delta \hat{u}(\mathbf{x}, \omega)$ and $\delta c(\mathbf{x})$. To achieve this goal, we first note that to first order we have the following result

$$\frac{\omega^2}{(c(\mathbf{x}) + \delta c(\mathbf{x}))^2} \approx \frac{\omega^2}{c^2(\mathbf{x})} \left(1 - 2 \frac{\delta c(\mathbf{x})}{c(\mathbf{x})} \right). \quad (9)$$

Then using equation 9, and subtracting equation 1 from equation 8 and retaining only the first order terms we get

$$\left(\nabla^2 + \frac{\omega^2}{c^2(\mathbf{x})}\right) \delta \hat{u}(\mathbf{x}, \omega) = 2\omega^2 \frac{\delta c(\mathbf{x})}{c^3(\mathbf{x})} \hat{u}(\mathbf{x}, \omega). \quad (10)$$

The above equation 10 is the linearized Born formula for the Helmholtz equation.

TFWI extended linearized Born forward modeling

A great advantage of solving the wave equation in the frequency domain is that TFWI extended Born linearized forward modeling also becomes straightforward. We first recall the time domain wave equation for the same as formulated in Biondi and Almomin (2014),

$$\frac{1}{c^2(\mathbf{x})} \frac{\partial^2 u_e(\mathbf{x}, t)}{\partial t^2} - \nabla^2 u_e(\mathbf{x}, t) = \int_{\mathbb{R}} 2 \frac{\partial^2 u}{\partial t^2}(\mathbf{x}, t - \tau) \frac{\delta c_e(\mathbf{x}, \tau)}{c^3(\mathbf{x})} d\tau, \quad (11)$$

where $u(\mathbf{x}, t)$ solves equation 7, $\delta c_e(\mathbf{x}, t)$ is the extended velocity model, and $u_e(\mathbf{x}, t)$ is the extended Born linearized forward modeled data.

We can Fourier transform equation 11 with respect to time to get the corresponding Helmholtz equation in the frequency domain given by

$$\left(\nabla^2 + \frac{\omega^2}{c^2(\mathbf{x})}\right) \hat{u}_e(\mathbf{x}, \omega) = \frac{2\omega^2}{c^3(\mathbf{x})} \hat{u}(\mathbf{x}, \omega) \delta \hat{c}_e(\mathbf{x}, \omega), \quad (12)$$

where $\hat{u}_e(\mathbf{x}, \omega)$ is the Fourier transform of $u_e(\mathbf{x}, t)$, and $\delta \hat{c}_e(\mathbf{x}, \omega)$ is the Fourier transform of $\delta c_e(\mathbf{x}, t)$. As can be seen from equation 12, in order to compute $\hat{u}_e(\mathbf{x}, \omega)$, we only need to change the forcing term in the Helmholtz equation and moreover the forcing term is just a point wise multiplication for each frequency ω . Thus, we have been able to reduce the expensive convolution operation in time domain to a trivial multiplication operation, while at the same time the Helmholtz operator on the left remains unchanged which means that the matrix factorization can be reused.

NUMERICAL EXPERIMENTS

In this section, we present some numerical tests in 2D performed with the 2D Helmholtz equation solver library. For the first four tests, our simulation geometry consists of 500×500 cells along X and Z directions. The top left corner of the geometry has the coordinates (0 km, 0 km). The simulation box dimensions along the two directions are 10 km \times 10 km, which gives a grid spacing of 0.02 km along both the X and Z directions. In the first and second tests we solve the Helmholtz equation in a homogenous medium and a heterogenous two layer medium, respectively. In the third numerical experiment, we test the accuracy of the Born linearization, while in the fourth experiment we test the free-surface boundary condition for the top surface which is encountered for example while dealing with the air-water interface. For the fifth and final test, we perform extended linearized Born forward modeling in a homogenous background medium with the extended model given by a point scatterer. In all the cases, we plot both the real and imaginary parts of the solution to the Helmholtz equation.

Homogenous velocity field

In this experiment, we use a constant homogenous velocity $c(\mathbf{x}) = 1$ km/s. We place the source at the center of the model at (5 km, 5 km). We use PML boundary conditions on all sides of the model. We show the results for two different values of angular frequency — Figures 1(a) and 1(b) show the results for $\omega = 20$ rad/sec, and Figures 2(a) and 2(b) show the results for $\omega = 40$ rad/sec. Visual inspection of the figures reveal agreement with what we expect of the solutions - the wavelength of the solution for $\omega = 20$ rad/sec is twice that of the wavelength of the solution for $\omega = 40$ rad/sec. Moreover, we see that the solutions are spherically symmetric which is what we expect.

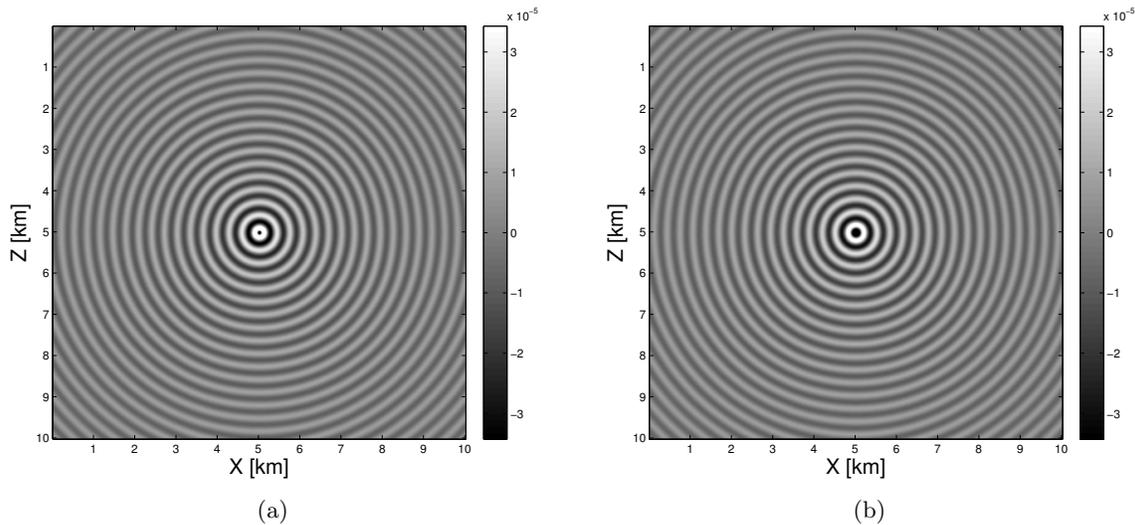


Figure 1: Solution to the Helmholtz equation with a constant velocity medium $c(\mathbf{x}) = 1$ km/s, for $\omega = 20$ rad/sec, and a unit impulse source placed at (5 km, 5 km) : a) Real part, b) Imaginary part. [ER] rahull/. real-homog-20,imag-homog-20

Heterogenous velocity field

In this experiment, we use a heterogenous two-layer velocity model that is constant horizontally, given by $c(\mathbf{x}) = 1$ km/s for $z = 0-5$ km, and $c(\mathbf{x}) = 2$ km/s for $z = 5-10$ km. We place the source at the coordinates (5 km, 2.5 km). We again use PML boundary conditions on all sides of the model and show the results for two different values of angular frequency — Figures 3(a) and 3(b) show the results for $\omega = 20$ rad/sec, and Figures 4(a) and 4(b) show the results for $\omega = 40$ rad/sec. The figures reveal superposition between direct waves from the source and reflected waves from the interface in the top layer, while we only have transmitted waves in the bottom layer. Also the wavelength of the solution is twice in the second layer as compared to the first layer, which is expected theoretically.

Linearized Born scattering

In the linearized Born scattering tests, we design the experiment using a constant velocity background medium given by $c(\mathbf{x}) = 1$ km/s with the source placed at the center of the

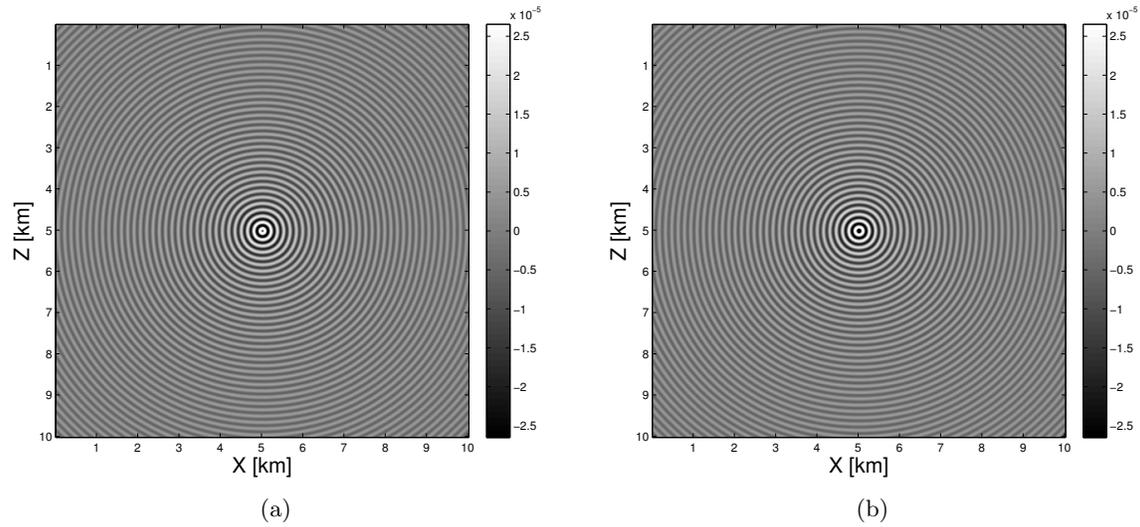


Figure 2: Solution to the Helmholtz equation with a constant velocity medium $c(\mathbf{x}) = 1$ km/s, for $\omega = 40$ rad/sec, and a unit impulse source placed at (5 km, 5 km) : a) Real part, b) Imaginary part. [ER] rahul1/. real-homog-40,imag-homog-40

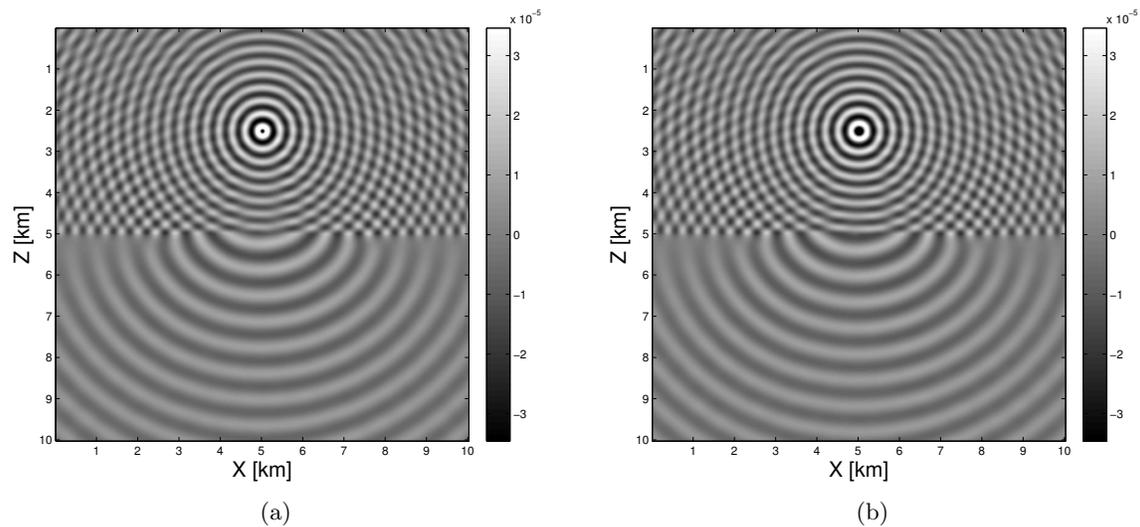


Figure 3: Solution to the Helmholtz equation at $\omega = 20$ rad/sec with a two layer (horizontally constant) medium given by $c(\mathbf{x}) = 1$ km/s for $z = 0-5$ km, and $c(\mathbf{x}) = 2$ km/s for $z = 5-10$ km, and a unit impulse source placed at (5 km, 2.5 km) : a) Real part, b) Imaginary part. [ER] rahul1/. real-hetero-2layer-20,imag-hetero-2layer-20

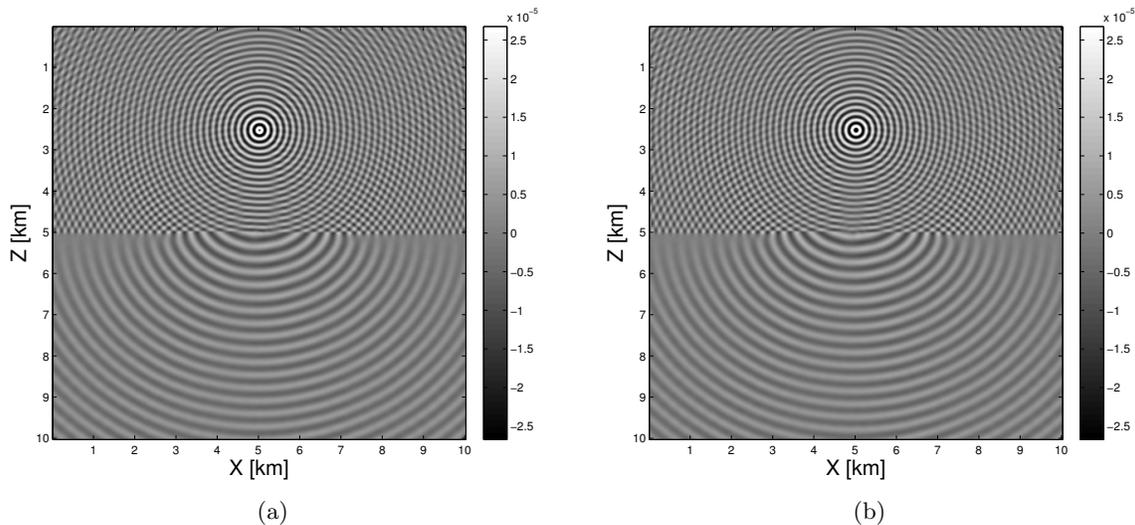


Figure 4: Solution to the Helmholtz equation at $\omega = 40$ rad/sec with a two layer (horizontally constant) medium given by $c(\mathbf{x}) = 1$ km/s for $z = 0-5$ km, and $c(\mathbf{x}) = 2$ km/s for $z = 5-10$ km, and a unit impulse source placed at $(5 \text{ km}, 2.5 \text{ km})$: a) Real part, b) Imaginary part. [ER] rahul1/. real-hetero-2layer-40,imag-hetero-2layer-40

model at $(5 \text{ km}, 5 \text{ km})$. The Helmholtz equation is solved to obtain the background solution $\hat{u}(\mathbf{x}, \omega)$. We use PML boundary conditions on all sides of this model for this test, and set the angular frequency $\omega = 20$ rad/sec. Next, we create a velocity perturbation at a single point in the model at $(5 \text{ km}, 7.5 \text{ km})$ of 10% (0.1 km/s), and zero everywhere else. This is our $\delta c(\mathbf{x})$.

We first create a new velocity model by adding the perturbation to the background velocity, given by $\tilde{c}(\mathbf{x}) = c(\mathbf{x}) + \delta c(\mathbf{x})$. The Helmholtz equation is then solved to get the new solution given by $\tilde{u}(\mathbf{x}, \omega)$. We then calculate the true perturbed solution as $\widetilde{\delta u}(\mathbf{x}, \omega) = \tilde{u}(\mathbf{x}, \omega) - \hat{u}(\mathbf{x}, \omega)$. The real and imaginary components of the true perturbed wave field generated this way are plotted in Figures 5(a) and 5(b) respectively.

Next, we perform linearized Born modeling using equation 10 to get a first order estimate of the true perturbed solution, given by $\delta \hat{u}(\mathbf{x}, \omega)$. The real and imaginary components of $\delta \hat{u}(\mathbf{x}, \omega)$ are plotted in Figures 6(a) and 6(b) respectively. As we can see from the colorbars, the amplitudes of $\widetilde{\delta u}(\mathbf{x}, \omega)$ are much lower compared to $\delta \hat{u}(\mathbf{x}, \omega)$. However, the solutions resemble each other upon visual inspection. The errors can be attributed to the relatively large magnitude of the velocity perturbation, which is expected to decrease as we reduce the magnitude of the perturbation at a quadratic rate. This is what we demonstrate next.

We proceed by varying the velocity perturbation $\delta c(\mathbf{x})$ placed at the same location as before, but this time varying it from 1%, 2%, ..., 10%. The quantities $\widetilde{\delta u}(\mathbf{x}, \omega)$ and $\delta \hat{u}(\mathbf{x}, \omega)$ are recalculated for each case and then we calculate the 2-norm of the difference $\|\widetilde{\delta u}(\mathbf{x}, \omega) - \delta \hat{u}(\mathbf{x}, \omega)\|_2$. The results are plotted in Figure 7. As we can see from the image, the relationship is approximately quadratic.

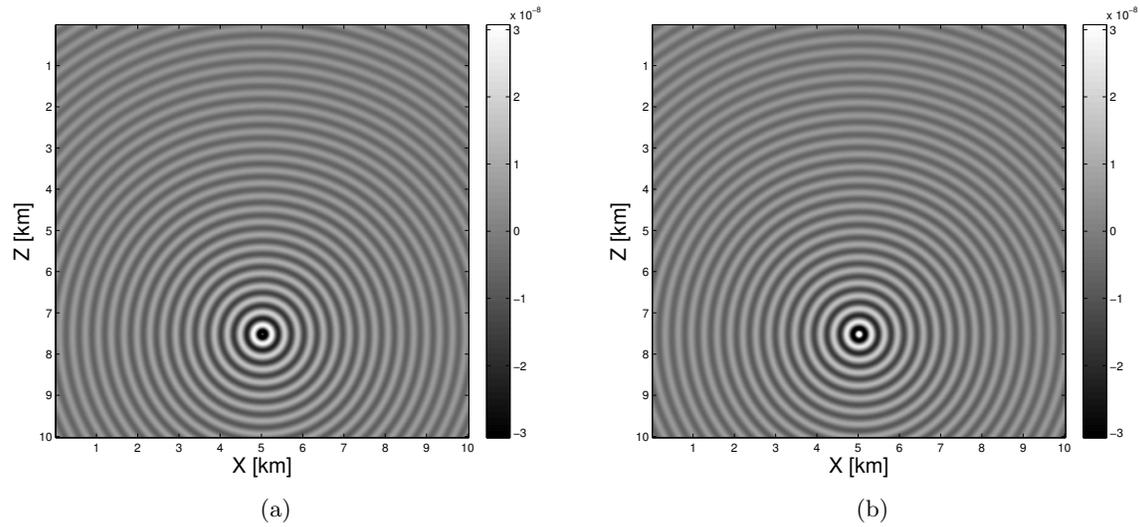


Figure 5: True difference between the solutions to the Helmholtz equation $\hat{u}(\mathbf{x}, \omega)$ and $\tilde{\hat{u}}(\mathbf{x}, \omega)$ at $\omega = 20$ rad/sec, formed respectively using a constant velocity background medium given by $c(\mathbf{x}) = 1$ km/s, and that with a perturbed medium obtained by adding 0.1 km/s to the background at (5 km, 7.5 km). The primary source is a unit impulse located at (5 km, 5 km). The quantity plotted is the perturbed wave field $\delta\tilde{\hat{u}}(\mathbf{x}, \omega) = \tilde{\hat{u}}(\mathbf{x}, \omega) - \hat{u}(\mathbf{x}, \omega)$: a) Real part, b) Imaginary part. [ER] rahul1/. real-diff,imag-diff

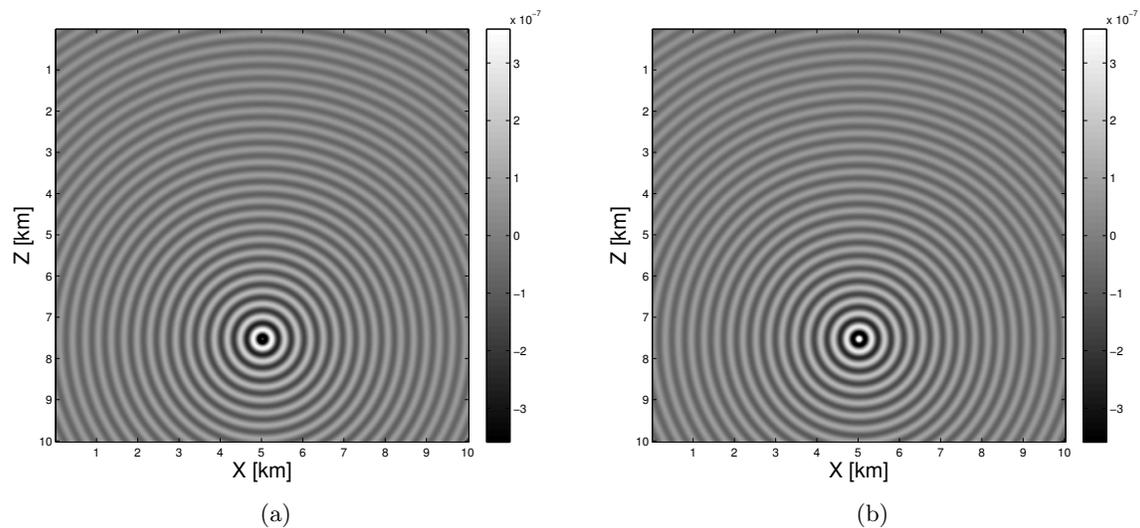


Figure 6: First order estimate of the perturbed wave field using Born modeling $\delta\tilde{\hat{u}}(\mathbf{x}, \omega)$ at $\omega = 20$ rad/sec, with a scatterer located at (5 km, 7.5 km) of 0.1 km/s : a) Real part, b) Imaginary part. [ER] rahul1/. real-born,imag-born

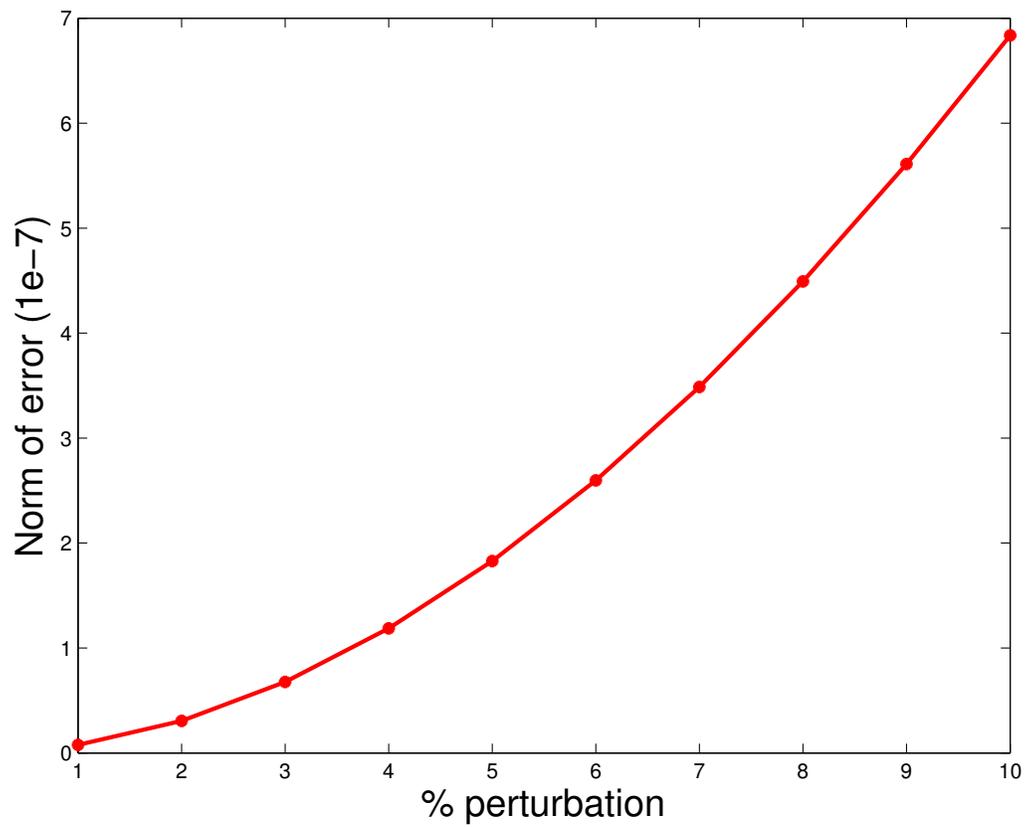


Figure 7: Plot of the norm of the error $\|\widetilde{\delta\hat{u}}(\mathbf{x}, \omega) - \delta\hat{u}(\mathbf{x}, \omega)\|_2$ vs $\frac{\delta c(\mathbf{x})}{c(\mathbf{x})} \%$, where $\widetilde{\delta\hat{u}}(\mathbf{x}, \omega)$ is the first order Born scattered wave field and $\delta\hat{u}(\mathbf{x}, \omega)$ is the true perturbed wave field.

[CR] rahul1/. norm-error

Free surface modeling

In this experiment, we use a constant homogenous velocity $c(\mathbf{x}) = 1$ km/s, and place the source at the center of the model at (5 km, 5 km), just like the first numerical experiment. We use PML boundary conditions on the left, right and bottom sides of the model. But to simulate the air-water interface on the top of the model, we use the homogenous Neumann boundary condition on the top side. This is expected to generate reflections from the top surface. We show the results for two different values of angular frequency — Figures 8(a) and 8(b) show the results for $\omega = 20$ rad/sec, and Figures 9(a) and 9(b) show the results for $\omega = 40$ rad/sec.

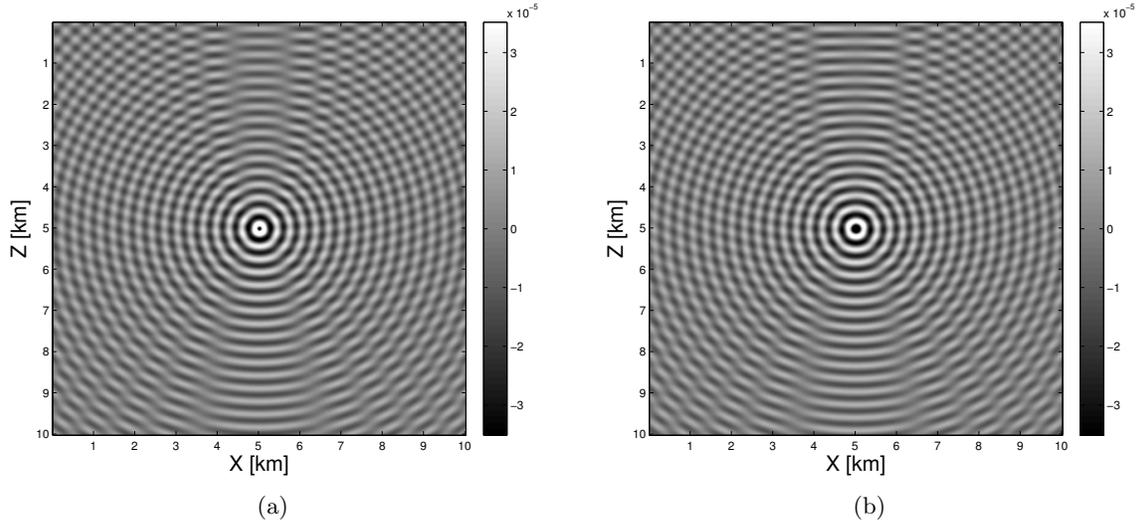


Figure 8: Solution to the Helmholtz equation with a constant velocity medium $c(\mathbf{x}) = 1$ km/s, for $\omega = 20$ rad/sec, and a unit impulse source placed at (5 km, 5 km), and homogeneous Neumann Boundary Conditions on the top side : a) Real part, b) Imaginary part.

[ER] rahull/. real-homog-freesurface-20,imag-homog-freesurface-20

TFWI forward modeling

The goal of the last and final experiment is to perform TFWI linearized extended Born forward modeling using the Helmholtz equation, as described in equation 12. We will also attempt to reconstruct the time domain recorded waveform in this experiment. We start by using a constant homogenous velocity medium with $c(\mathbf{x}) = 5$ km/s as the background, and place the source at the center of the model. However this time instead of using a point source, we choose a spatially varying Gaussian source (in both X and Z directions) centered at $(x_0 = 5$ km, $z_0 = 5$ km) and standard deviation $\sigma = 0.1$ km, with the source injected in phase at all the spatial points. Thus, if we denote the time domain wavelet as $w(t)$, then the time domain source denoted by $f(\mathbf{x}, t)$ is given as

$$f(\mathbf{x}, t) = w(t) \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-x_0)^2 + (z-z_0)^2}{2\sigma^2}\right). \quad (13)$$

We use a Ricker wavelet $w(t)$ with peak frequency $f_p = 10$ Hz, sampling interval $\Delta t = 8$ ms,

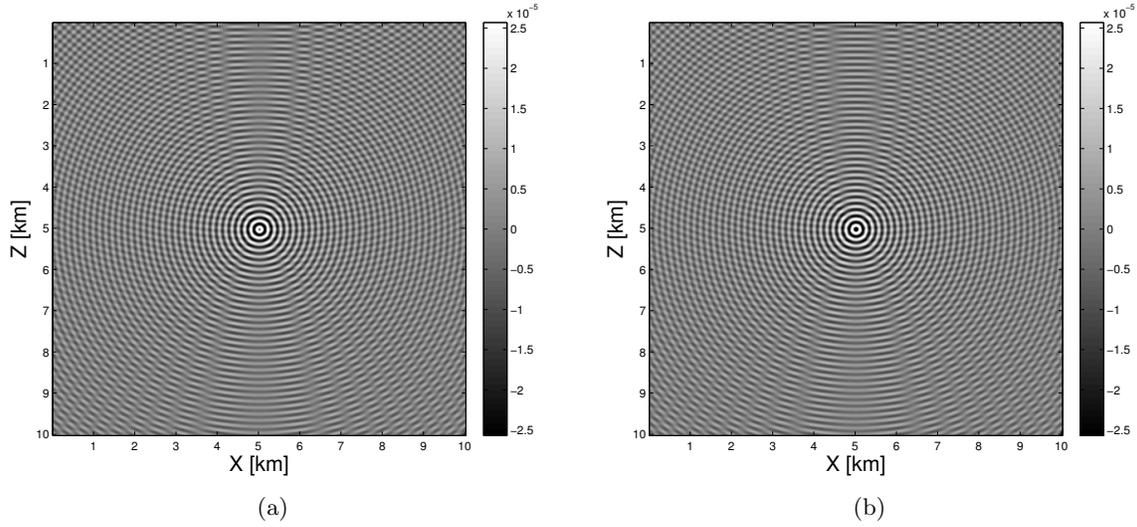


Figure 9: Solution to the Helmholtz equation with a constant velocity medium $c(\mathbf{x}) = 1$ km/s, for $\omega = 40$ rad/sec, and a unit impulse source placed at (5 km, 5 km), and homogeneous Neumann Boundary Conditions on the top side : a) Real part, b) Imaginary part.

[ER] rahul1/. real-homog-freesurface-40,imag-homog-freesurface-40

number of samples $N_t = 250$ and a delay of $\delta t = 0.2$ s. This gives us a total modeling time of $N_t \Delta t = 2$ s. The functional form of the wavelet is given by

$$w(t) = (1 - 2\pi^2 f_p^2 (t - \delta t)^2) \exp(-\pi^2 f_p^2 (t - \delta t)^2). \quad (14)$$

The wavelet $w(t)$ and its amplitude spectrum are plotted in Figures 10(a) and 10(b) respectively, while the spatially varying Gaussian field for source injection is shown in Figure 10(c).

The source function $f(\mathbf{x}, t)$ is then discrete Fourier transformed in time, which gives us an angular frequency sampling of $\Delta\omega = \frac{2\pi}{N_t \Delta t}$. The Fourier transformed coefficients of $f(\mathbf{x}, t)$ correspond to the N_t angular frequencies given by the ordered set $\{(-\frac{N_t}{2} + 1) \Delta\omega, \dots, 0, \dots, \frac{N_t}{2} \Delta\omega\}$, where the angular frequencies are spaced by units of $\Delta\omega$. This gives us the set of frequencies for which we need to solve the Helmholtz equation. However, because the time domain solution to the wave equation is purely real, the solution to the Helmholtz equation satisfies the following symmetry condition

$$\hat{u}(\mathbf{x}, -\omega) = \hat{u}(\mathbf{x}, \omega)^*, \quad (15)$$

which means that we only need to solve the Helmholtz equation for the positive frequencies, and the solutions for the negative frequencies are obtained simply by complex conjugation. Once all the solutions for the different frequencies have been found by solving the Helmholtz equation, we can inverse discrete Fourier transform the solution to get the time domain wave field.

In order to demonstrate this process, we place our receivers at a depth given by $z = 2.5$ km, and for all grid points in the X direction. We first solve for the background wave field using

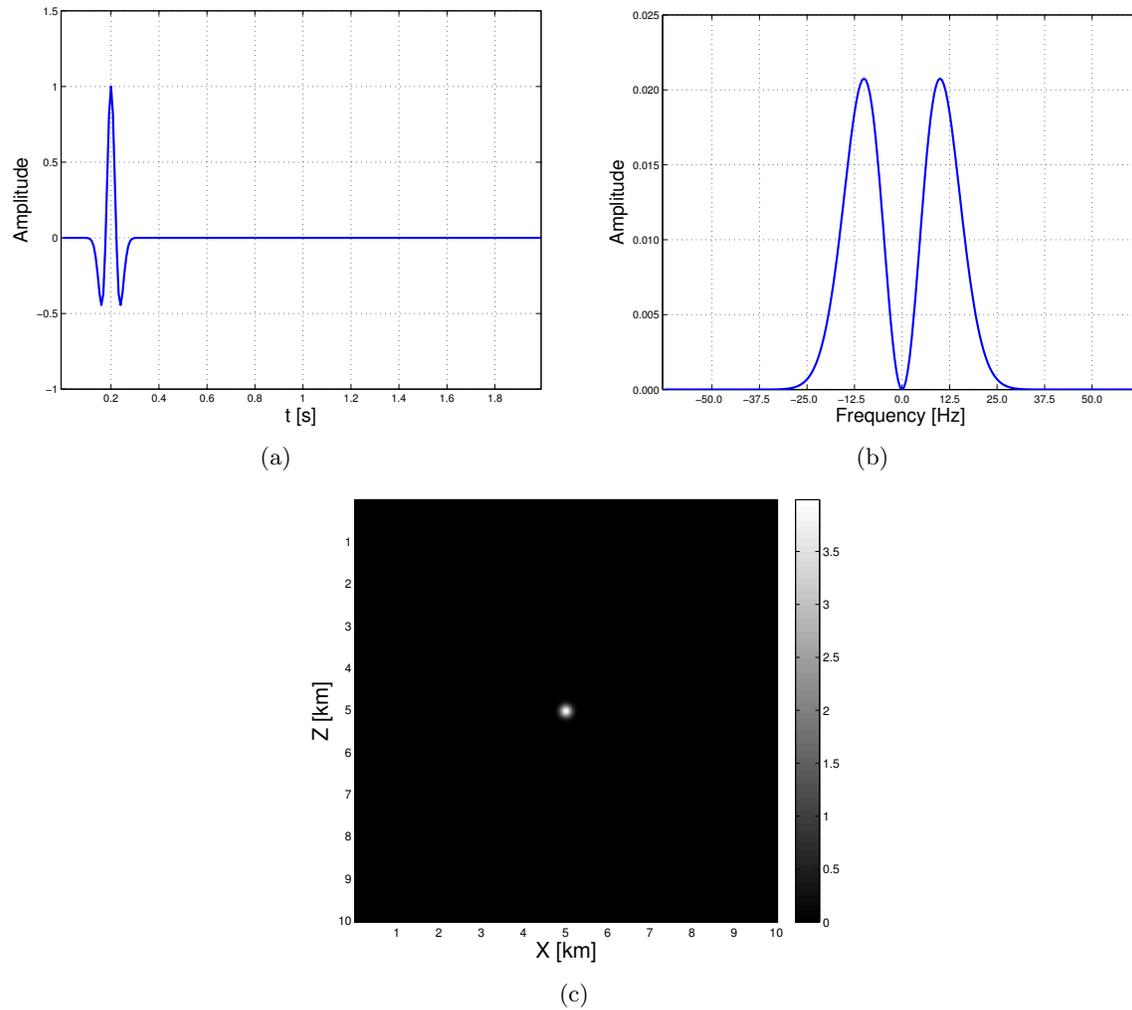


Figure 10: Source function $f(\mathbf{x}, t)$ used for the TFWI experiment : a) Ricker wavelet in time $w(t)$, b) Amplitude spectra of $w(t)$, c) Spatially varying Gaussian field used for source injection. [CR] rahul1/. ricker-wavelet,ricker-spectra,gauss-smooth

this process and the resulting time domain wave field at the receiver locations is shown in Figure 11.

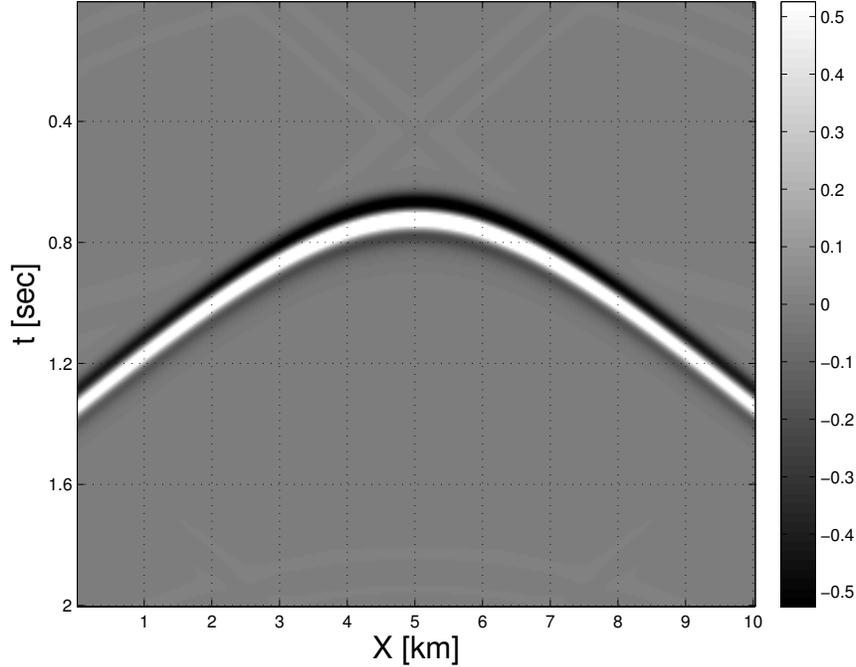


Figure 11: Background wave field at receiver depth $z = 2.5$ km for a constant homogenous velocity medium $c(\mathbf{x}) = 5$ km/s, formed by inverse Fourier transforming the frequency domain solutions of the Helmholtz equation. [CR] rahul1/. real-reconstruction

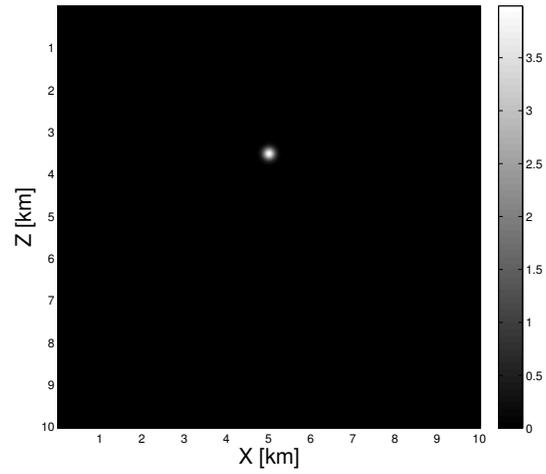
Next for the extended linearized Born modeling, we use an extended velocity model of the following form

$$\delta c_e(\mathbf{x}, t) = v(t) \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - x_1)^2 + (z - z_1)^2}{2\sigma^2}\right), \quad (16)$$

where we have chosen a product representation for the extended model with the time signature given by $v(t)$, and is modulated in space by a Gaussian centered at $(x_1 = 5$ km, $z_1 = 3.5$ km) with standard deviation $\sigma = 0.1$ km, as shown in Figure 12. We try three different cases for $v(t)$ — (a) $v(t) = \delta_0(t)$, (b) $v(t) = \delta_a(t)$ and (c) $v(t) = \delta_{-a}(t)$, where $a = 0.5$ s.

To use the Helmholtz solver, we proceed by discrete Fourier transforming $\delta c_e(\mathbf{x}, t)$ to get $\delta \hat{c}_e(\mathbf{x}, \omega)$ for each case, and at the same set of discrete angular frequencies as in the case of the background wave field. Then the forcing term in equation 12 is easily calculated as a point-wise multiplication with the already solved background wave field to get $\frac{2\omega^2}{c^3(\mathbf{x})} \hat{u}(\mathbf{x}, \omega) \delta \hat{c}_e(\mathbf{x}, \omega)$ at each angular frequency to be solved for. Finally, we inverse discrete Fourier transform the solutions to the Helmholtz equation to get the time domain solution. These results are again displayed for all receivers at depth $z = 2.5$ km in Figures 13, 14 and 15 for the three cases with $v(t)$ given by $\delta_0(t)$, $\delta_a(t)$ and $\delta_{-a}(t)$ respectively.

The first case with $v(t) = \delta_0(t)$ corresponds to standard Born modeled wave field, while the second case with $v(t) = \delta_a(t)$ and the third case with $v(t) = \delta_{-a}(t)$ correspond to the cases where the Born modeled wave field is delayed by +0.5s and -0.5s, respectively.



(a)

Figure 12: Spatially varying Gaussian field component of the extended velocity model $\delta c_e(\mathbf{x}, t)$. [CR] rahul1/. gauss-pert-smooth

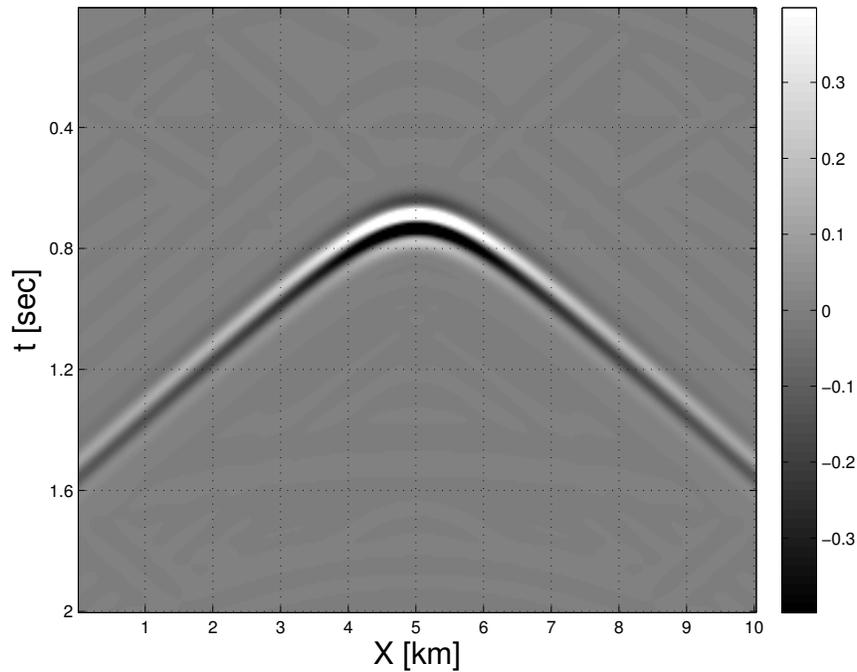


Figure 13: Extended linearized Born modeled wave field for the TFWI experiment recorded at receiver depth $z = 2.5$ km for $v(t) = \delta_0(t)$. [CR] rahul1/. real-extBorn1-reconstruction

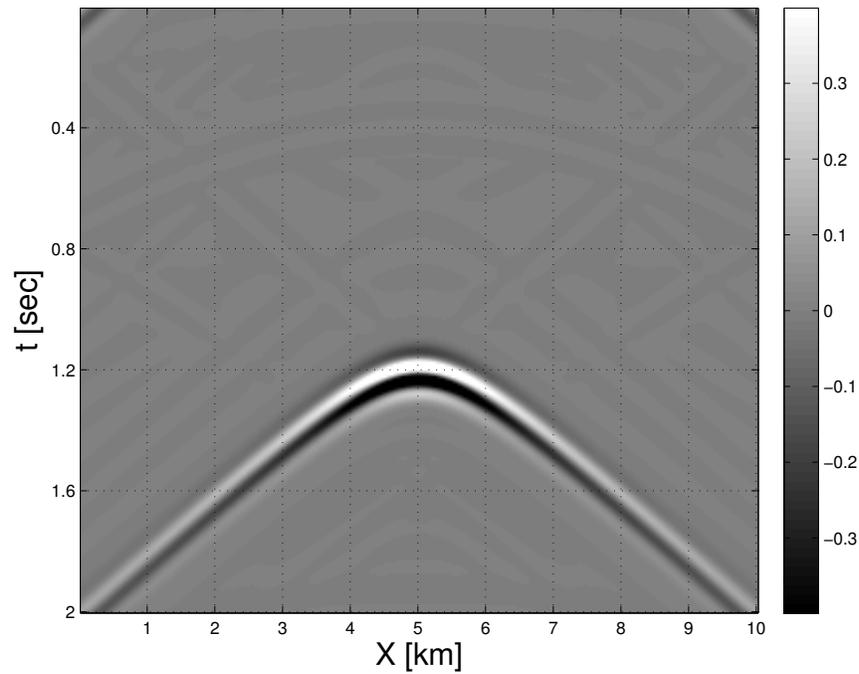


Figure 14: Extended linearized Born modeled wave field for the TFWI experiment recorded at receiver depth $z = 2.5$ km for $v(t) = \delta_a(t)$, where $a = 0.5$ s. [CR] rahul1/. real-extBorn2-reconstruction

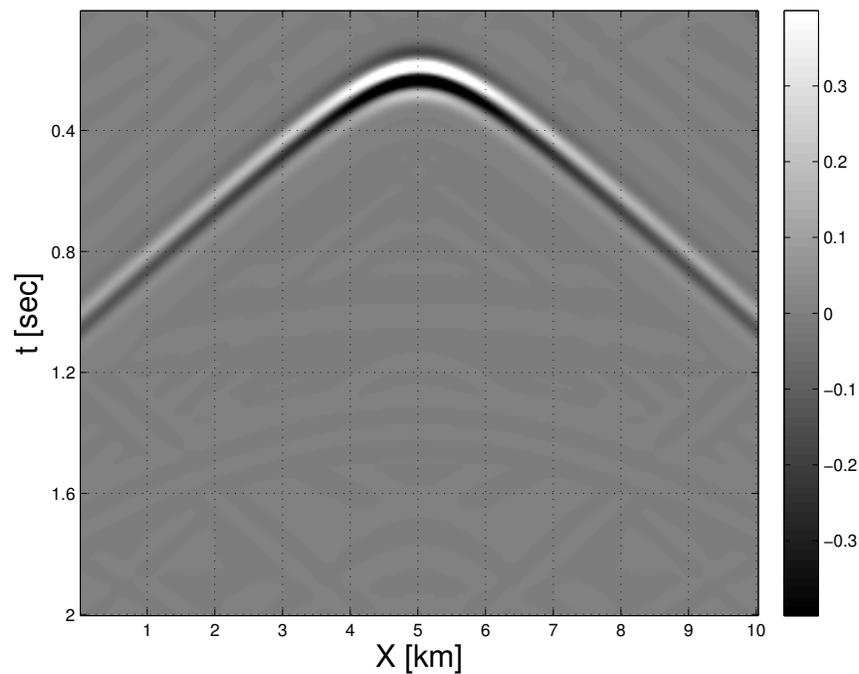


Figure 15: Extended linearized Born modeled wave field for the TFWI experiment recorded at receiver depth $z = 2.5$ km for $v(t) = \delta_{-a}(t)$, where $a = 0.5$ s. [CR] rahul1/. real-extBorn3-reconstruction

DISCUSSION AND FUTURE WORK

In this paper, we have developed a 2D Helmholtz equation solver library in C++. We provided some numerical examples by solving the Helmholtz equation using the library for homogenous and inhomogenous media, for different frequencies. We also performed a Born linearization test to verify the theoretical prediction that the error is indeed quadratic with respect to the magnitude of the velocity perturbation. Finally, we showed how extended linearized Born forward modeling can be performed in the frequency domain.

Future work with this library is aimed at extending it to solve the 3D Helmholtz equation. The eventual goal of this research project, started this summer, is to use the library to solve TFWI problem in the frequency domain which will likely bring enormous computational savings. The immediate near term goal is to integrate the Helmholtz solver with an inversion library and experiment with 2D TFWI.

ACKNOWLEDGEMENT

We would like to thank Prof. Lexing Ying and Prof. Andras Vasy for helpful discussions on the numerical and analytical aspects of the Helmholtz equation. We would also like to thank Prof. Stew Levin and Eileen Martin for proof-reading and suggesting valuable edits that greatly improved the quality of this paper.

REFERENCES

- Biondi, B. and A. Almomin, 2014, Simultaneous inversion of full data bandwidth by tomographic full waveform inversion: *Geophysics*, **79**, no. 3, WA129–WA140.
- Davis, T. A., 2004, Algorithm 832: Umfpack v4. 3—An unsymmetric-pattern multifrontal method: *ACM Transactions on Mathematical Software (TOMS)*, **30**, 196–199.
- , 2006, *Direct methods for sparse linear systems*: SIAM.
- Johnson, S. G., 2008, Notes on perfectly matched layers (PMLs): Lecture notes, Massachusetts Institute of Technology, Massachusetts, **29**.
- Liu, F. and L. Ying, 2016, Additive sweeping preconditioner for the Helmholtz equation: *Multiscale Modeling & Simulation*, **14**, 799–822.
- Sommerfeld, A., 1949, *Partial differential equations in physics*, volume **1**: Academic Press.

Permian Basin High Density Land Acquisition Experimental Dataset

Stuart Farris and Rustam Akhmadiev

ABSTRACT

An experimental, high density land survey has been made available to the Stanford Exploration Project. Herein describes an overview of the dataset including survey parameters and employed wavelet removal techniques. Wave modes are identified in preliminary shot gathers and possible research directions are discussed.

INTRODUCTION

Producing oil and gas from onshore reserves has relatively low financial overhead in comparison with offshore operations (Fagan, 1997). This makes onshore production enticing during periods of oil and gas commodity recessions. Unfortunately, seismic data collected onshore is notoriously noisy and produces poor images of the subsurface. The noise and imaging issues are largely attributed to the interactions between the seismic source and the shallow-unconsolidated subsurface (Al-Ali, 2007). Better understanding the noise introduced by the shallow subsurface may lead to improved processing, modeling, and imaging of onshore oil and gas reserves.

An experimental seismic dataset was made available to the Stanford Exploration Project (SEP) through Occidental Petroleum in an attempt to better understand the surface noise within onshore seismic acquisitions. This data is particularly exciting because it is a combination of 2D and 3D high density surveys. It also lies over the Delaware Basin, a subset of the Permian Basin which is a hotbed for unconventional oil and gas production in the United States. This report outlines the beginning of a long journey of processing and experimentation on this unique dataset that aims to improve the overall quality of onshore seismic imaging.

SURVEY OVERVIEW

The acquisition parameters consist of three receiver layouts and three shot lines. The receiver geophone layouts are two patches and one line all with single phone (point receiver) recordings. The patches are square and have 441 receivers in a square 21x21 grid. They are overlain on one another and will be referred to as the 7001 patch and 8001 patch. The 7001 patch has 33x33 foot spacing and the 8001 patch has 16.5x16.5 foot spacing. The third receiver layout is a two dimensional line that extends outward from the overlain patches. This line, referred to as 6001 line, has 640 receivers spaced 16.5 feet apart. Figure 1 shows the three layouts. Figure 2 zooms in to show a closer view of the overlain patches.

Two of the shot lines extend parallel to the 6001 line with 165 foot spacing. One of the parallel shot lines lies on top of 6001 line and is referred to as the near line. The other

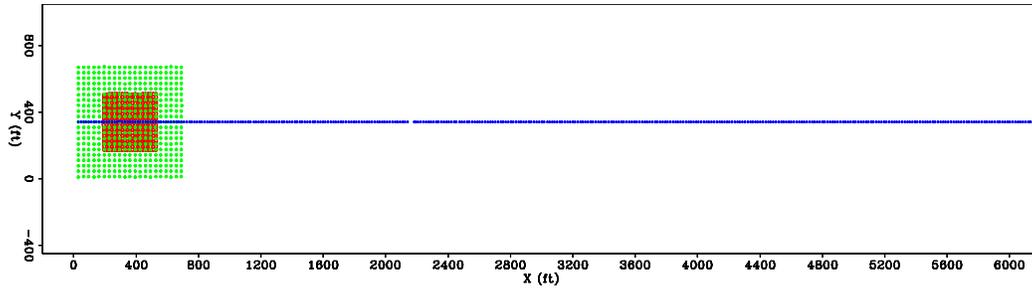


Figure 1: Receiver layouts 6001, 7001, 8001. [CR] `farris1/. geoRec`

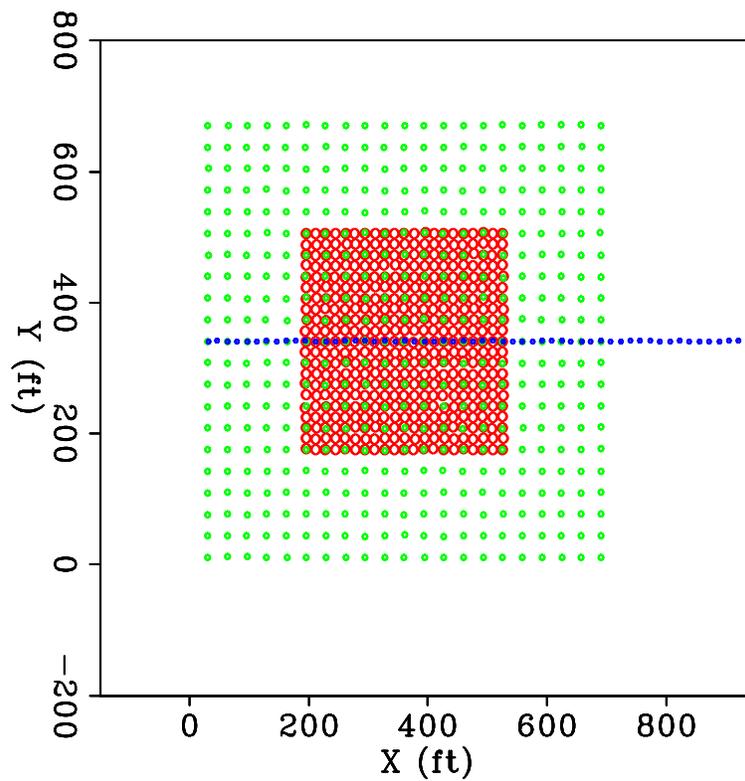


Figure 2: Zoomed view of receiver layouts 7001 and 8001. [CR] `farris1/. geoRecZoom`

parallel line, the far line, is offset from the 6001 line by about one mile to the north. The third shot line, the orthogonal line, runs perpendicular to the 6001 line on its western end.

Various source sweeps were used in each shot line. It is easiest to describe them in a list:

- Near Line
 - Test 1: Three sweeps per source location using four vibroseis trucks and a 16 second, 4-92 Hz linear sweep.
 - Test 2: Three sweeps per source location using one vibroseis truck and a 16 second, 4-92 Hz nominal sweep. Other three trucks were idle.
 - Test 3: Three sweeps per source location using four vibroseis trucks and a 16 second, 2-92 Hz linear sweep.
 - Test 4: Three sweeps per source location using four vibroseis trucks and a 16 second, 2-92 Hz dwell sweep.
 - Test 5: Noise test with four trucks shaking without pads in contact with ground. Performed every 16 source locations.
- Far Line
 - Test 6: Three sweeps per source location using four vibroseis trucks and a 16 second, 2-92 Hz linear sweep.
 - Test 7: Three sweeps per source location using four vibroseis trucks and a 16 second, 2-92 Hz dwell sweep.
- Orthogonal Line
 - Test 8: Three sweeps per source location using four vibroseis trucks and a 16 second, 2-92 Hz linear sweep.
 - Test 9: One sweep with one vibroseis truck every 41.25 feet using a 16 second, 2-92 Hz linear sweep.

DATA ORGANIZATION

The previous section outlined the source and receiver geometries of this dataset. Each source pattern was used for each receiver layout. The survey has nine source patterns and three receiver layouts. This means the dataset contains $3 \times 9 = 27$ source-receiver layout combinations. We perform the wavelet removal techniques described in this report on all combinations. In this way, the quality of each technique can be assessed on a variety of source sweeps and geometries.

WAVELET REMOVAL

Multiple wavelet removal techniques were attempted on the entire dataset with varying results. We discuss these techniques, their theoretical advantages, and their relative success. Figure 3 is included to introduce the provided raw data. From this data we hope to extract signal that represents coherent events and valuable surface noise information.

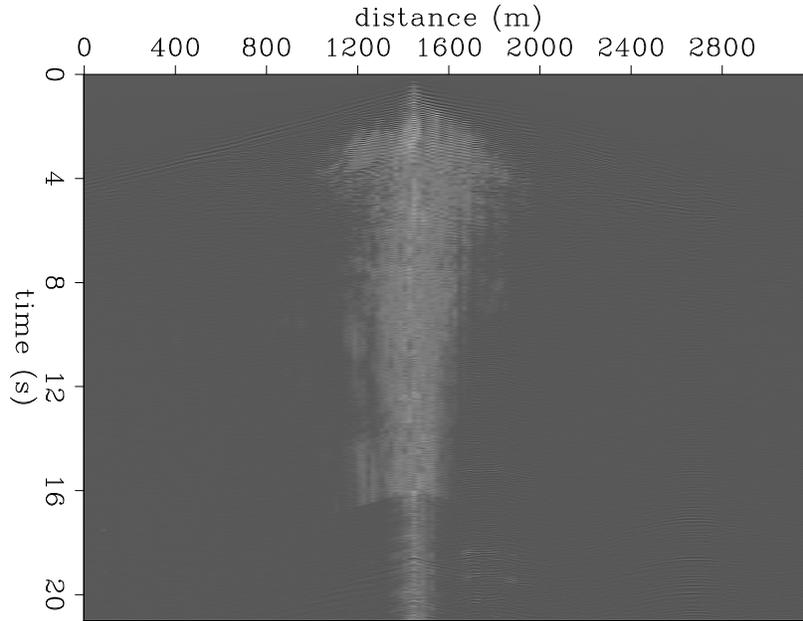


Figure 3: Uncorrelated shot gather from the two dimensional acquisition line. [CR]
 farris1/. uncorr2d

Correlation

The recorded vibroseis data can be represented in terms of a convolutional model, where the received signal is a convolution of the emitted source signature (sweep) and reflectivity of the earth. In frequency domain this operation corresponds to simple multiplication of their spectra. Meaning

$$d(t) = \sum_{\tau=0}^t s(t - \tau) * r(\tau) \quad (1)$$

$$D(\omega) = S(\omega) * R(\omega).$$

Where t is time, ω is angular frequency, $d(t)$ is recorded data in the time domain, $D(\omega)$ is recorded data in the frequency domain, $s(t)$ is source function in the time domain, $S(\omega)$ is source function in the frequency domain, and $R(\omega)$ is Earth's reflectivity in the frequency domain.

The most common initial step in the processing of vibroseis records is the correlation of the recorded seismic traces with the given sweep signal. This can be efficiently implemented using fast Fourier transform.

$$x(t) = \sum_{\tau=0}^t s(t + \tau) * d(\tau)$$

$$X(\omega) = \overline{S(\omega)} * D(\omega) = |S(\omega)| e^{-j\Phi_s(\omega)} * \underbrace{|S(\omega)| e^{j\Phi_s(\omega)} R(\omega)}_{D(\omega)} \quad (2)$$

$$X(\omega) = \underbrace{|S(\omega)|^2}_{\text{Klauder wavelet}} * R(\omega),$$

where $X(\omega)$ is the result of the correlation in the frequency domain. By taking the inverse Fourier transform of $X(\omega)$, we can retrieve the equivalent of the earth model convolved with a Klauder wavelet in the time domain (Yilmaz, 2001). Figure 4 illustrates the result of the correlation for the 6001 line and the 8001 patch, respectively. Since the correlation in frequency domain is periodic, the traces must be padded with array of zeros of appropriate size to avoid wraparound effect. Since the source signature is 16 seconds and the data was recorded for 21 seconds, the result of the correlation is truncated to 5 seconds.

The complex-conjugate source spectrum will have the same amplitude spectrum as the original but the phase will have the opposite sign. Therefore, cross-correlation will result into collapsing the original sweep by converting it into its zero-phase analogue (Klauder wavelet).

Deconvolution

Another way of approaching the wavelet removal problem is solving for reflectivity directly from the correlated data found from Equation 1. However, we need to try to stabilize the solution and avoid dividing by zero. This is usually done by adding a small percent(ϵ) of the maximum amplitude to the denominator. We can also guarantee that the denominator is positive by multiplying the numerator and the denominator by the complex conjugate of the source spectra. Thus,

$$R(\omega) = \frac{\overline{S(\omega)}D(\omega)}{S(\omega)\overline{S(\omega)} + \epsilon A_{max}}. \quad (3)$$

The results of deconvolution are displayed in Figure 5 for the 6001 line and the 8001 patch, respectively.

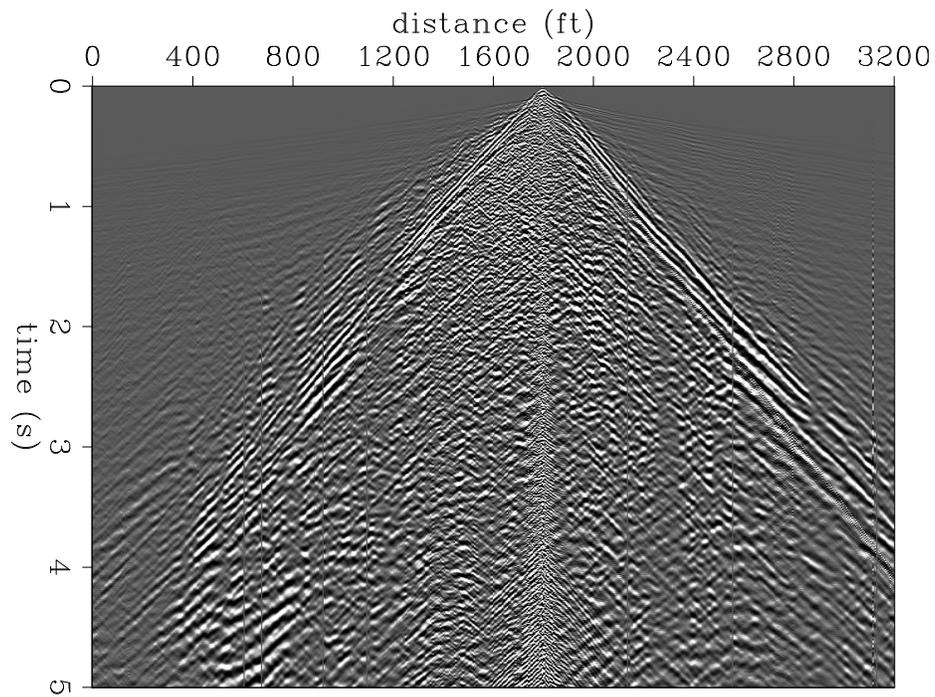
Minimum Phase Wavelet Shaping

Even when we avoid diving by zero, division itself is generally not a stable operation and can bring unwanted noise in the result. Therefore we attempt to reduce the amount of noise with Predictive Error Filter (PEF) deconvolution.

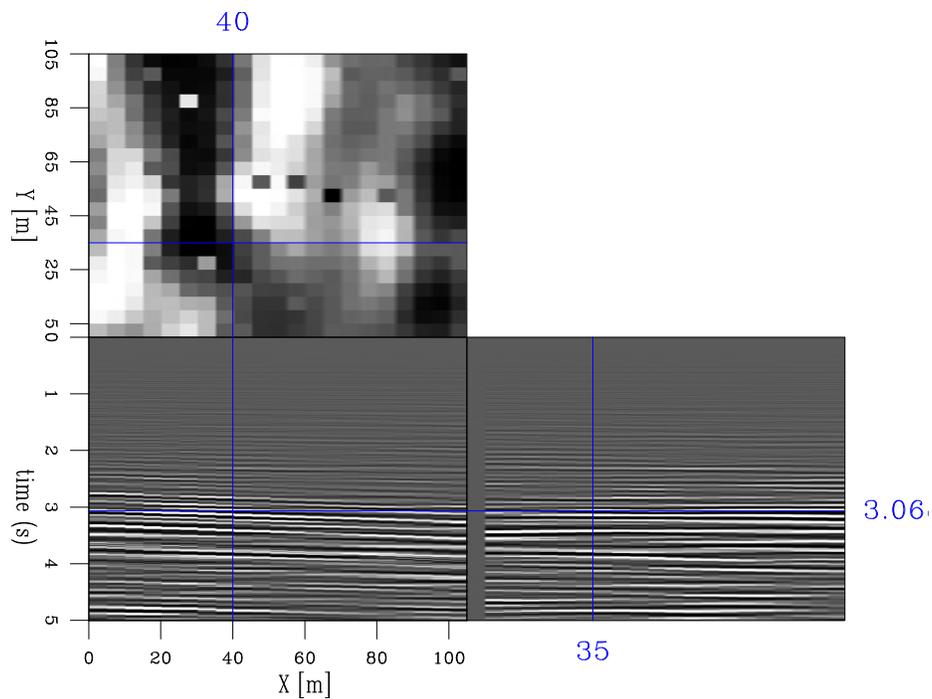
However, we have seen that correlation removes the phase of the original sweep out of the record and creates a zero-phase wavelet (autocorrelation of sweep or Klauder wavelet). Strictly speaking, deconvolution is stable only for a minimum-phase wavelet, since the inverse of such a wavelet is stable and causal. That is why prior to deconvolution we need to correct the signal for the phase so that the resulting wavelet is a minimum phase analogue of the original. In order to do this Kolmogoroff factorization is used (Claerbout, 2014). This procedure creates a minimum-phase wavelet based on its spectrum, and so it can be applied to the correlated traces in order to convert the Klauder wavelet to its minimum-phase analogue.

Prediction error filter

Using the minimum phase correction described above, PEF deconvolution can be applied to correlated traces to increase the temporal resolution of the image. In order to construct



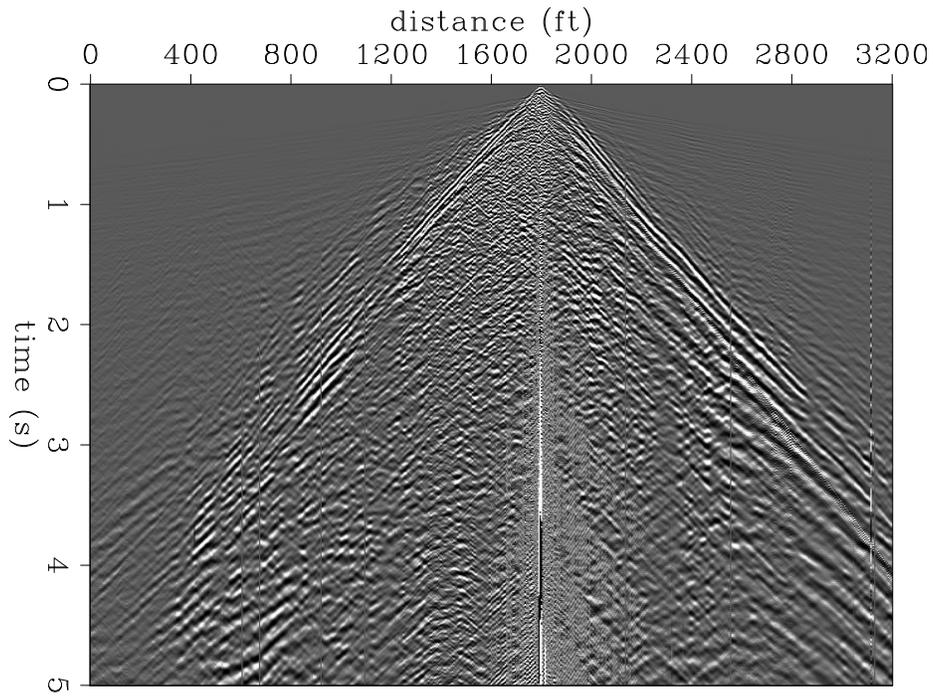
(a)



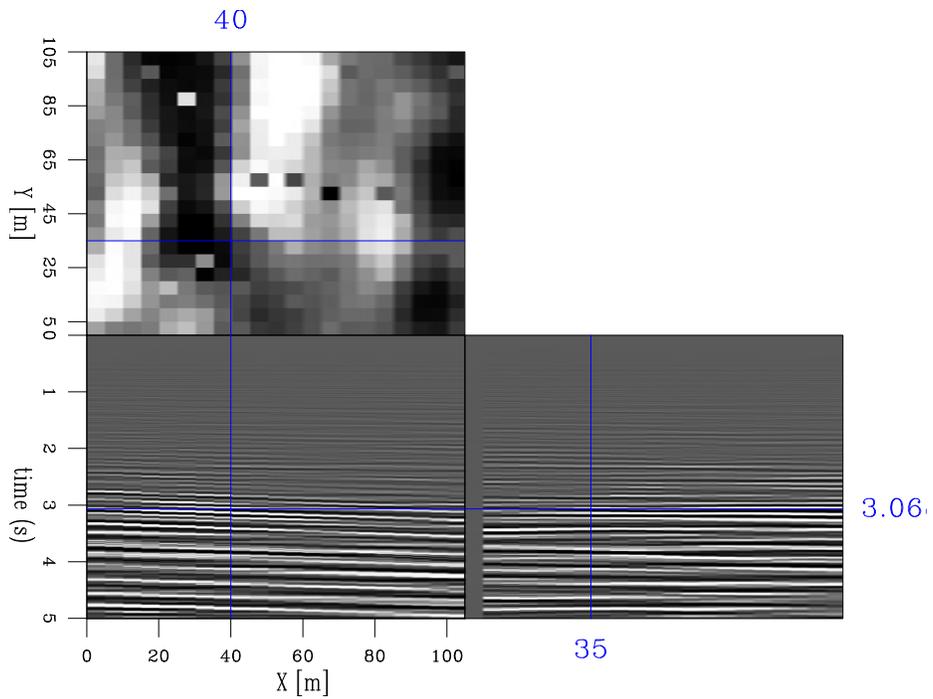
(b)

Figure 4: Correlation of shot gather from the (a) 6001 line and (b) 8001 patch. [CR]

farris1/.corr2d,corr3d



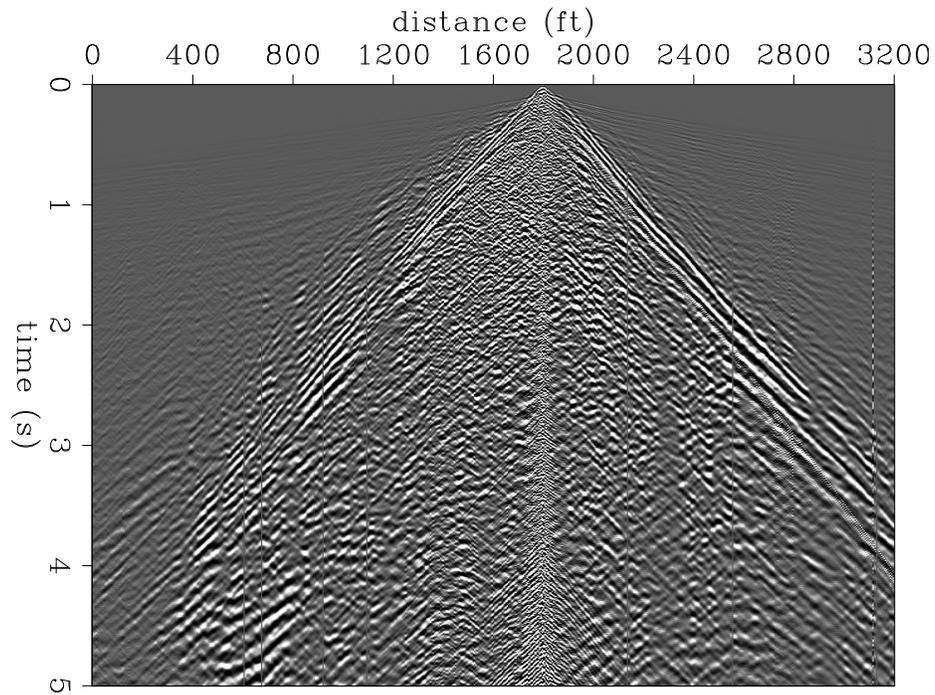
(a)



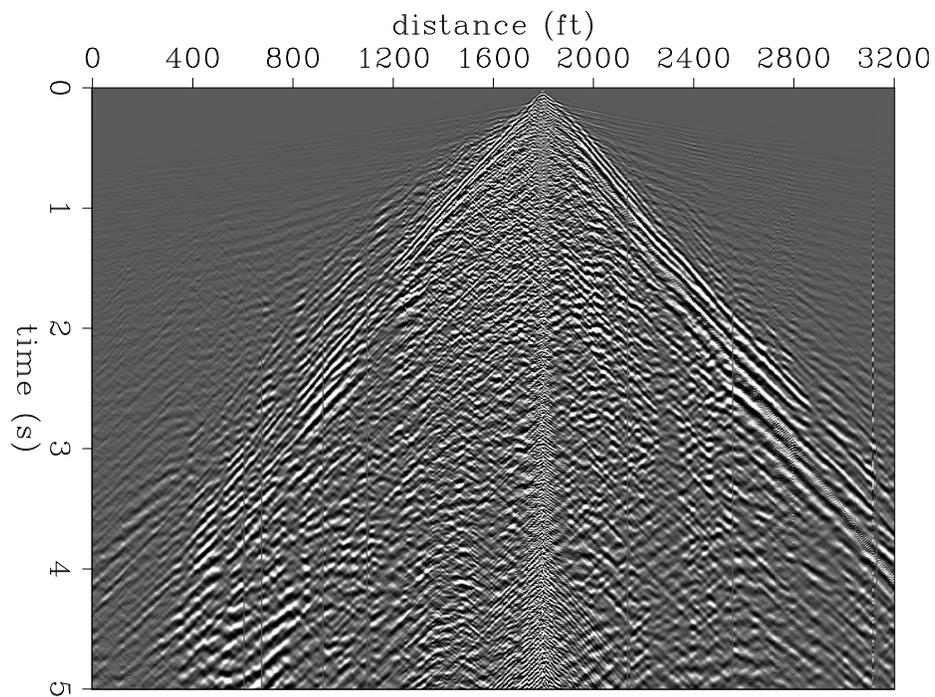
(b)

Figure 5: Decon of source wavelet from (a) 6001 line and (b) 8001 patch. [CR]

farris1/. decon2d,decon3d

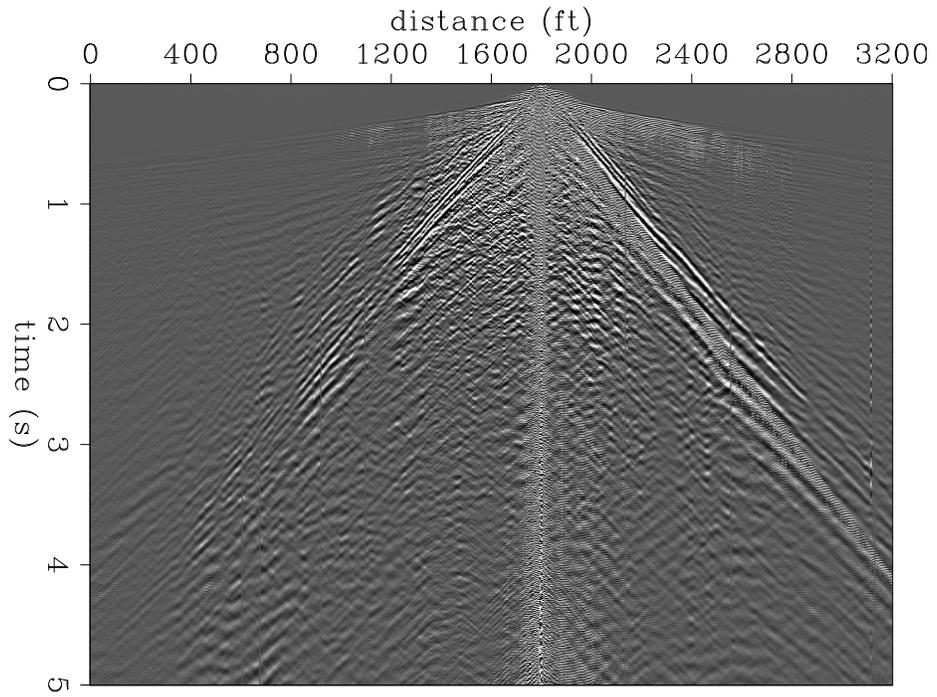


(a)

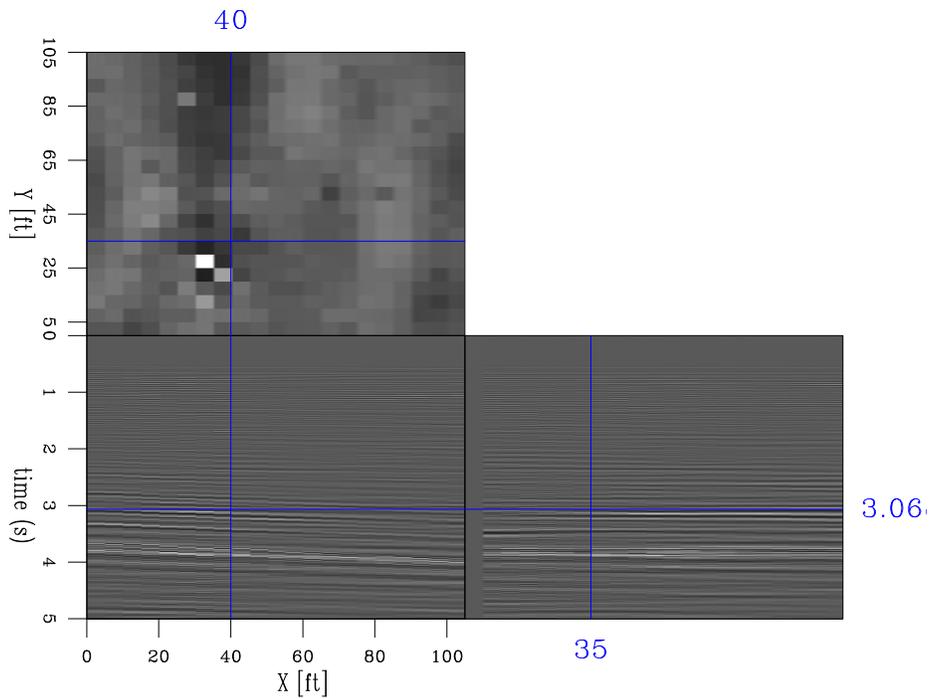


(b)

Figure 6: Correlation of shot gather from the 6001 line (a) before and (b) after min phase correction. [CR] `farris1/. corr2d,corrmin2d`



(a)



(b)

Figure 7: PEF decon from the (a) 6001 line and (b) 8001 patch.

[CR]

farris1/. wiener2d,wiener3d

the Prediction Error Filter (PEF) of size $n + 1$ and prediction interval of α Wiener-Hopf system of equations is solved using the Levinson-Durbin scheme (Yilmaz, 2001)

$$\begin{pmatrix} r_0 & r_1 & \cdots & r_{n-1} \\ r_1 & r_0 & \cdots & r_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n-1} & r_{n-2} & \cdots & r_0 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix} = \begin{pmatrix} r_\alpha \\ r_{\alpha+1} \\ \vdots \\ r_{\alpha+n-1} \end{pmatrix}, \quad (4)$$

r_i – is the i -th value of autocorrelation of the trace

Autocorrelation functions of the traces are averaged across all the traces in one shot gather, then the system 4 is solved for (f_1, f_2, \dots, f_n) . Corresponding PEF is constructed as $(1, \underbrace{0, 0, \dots, 0}_{\alpha-1}, -f_1, -f_2, \dots, -f_n)$. Filter length of 50 samples and prediction gap of 1 is used. For stability, a small percent of white noise ($\epsilon = 0.01$) is added to the diagonal term in the Toeplitz matrix above. The results of this PEF deconvolution are displayed in Figure 7 for the 6001 line and the 8001 patch, respectively.

Figure 8 illustrates all of the wavelet removal techniques in one plot. It can be seen that wavelet deconvolution and PEF deconvolution have brought minimal improvement to the sharpness of the recorded wavefields. Furthermore, these techniques have introduced substantial noise at near offsets and along the cone of the surface wave. Until we can improve the results of these deconvolutions, we will interpret various events using the minimum phase corrected gathers.

WAVE MODE IDENTIFICATION

In order to get rid of the dominant linear noise present in the record, we have tried applying a FK filter to the correlated data. The following parameters for the rejection zone in the FK domain were found to be suitable for surface noise removal:

- Maximum velocity - 2000 ft/s
- Minimum velocity - 0 ft/s
- Taper size - 50 samples

Figure 9 illustrates a shot with some picked events. The dominant noise in the correlated gathers is coming from the surface waves with apparent velocity of $\approx 1300-1400$ ft/s (shown in pink in Figures 9 and 10). There is also a fair amount of shallow back scattering shown with orange dotted lines that is characterized by a V-shaped form in the record. The first breaks are drawn as a solid red line with the velocity of refracted wave being $\approx 6000 - 6600$ ft/s. The yellow line on the figure may correspond to refracted S-wave. However, it is difficult to track and therefore this event may be coming from some other phenomena.

It is very challenging to identify reflected or diffracted waves on the record (shown in solid and dotted green lines). Nevertheless, after applying FK filter they sometimes become more apparent which allows us to pick them on the seismic gathers. These additional picks can be seen in Figure 10. However, additional work must be conducted to allow further analysis. Procedures like static corrections and median filtering may help to improve the images.

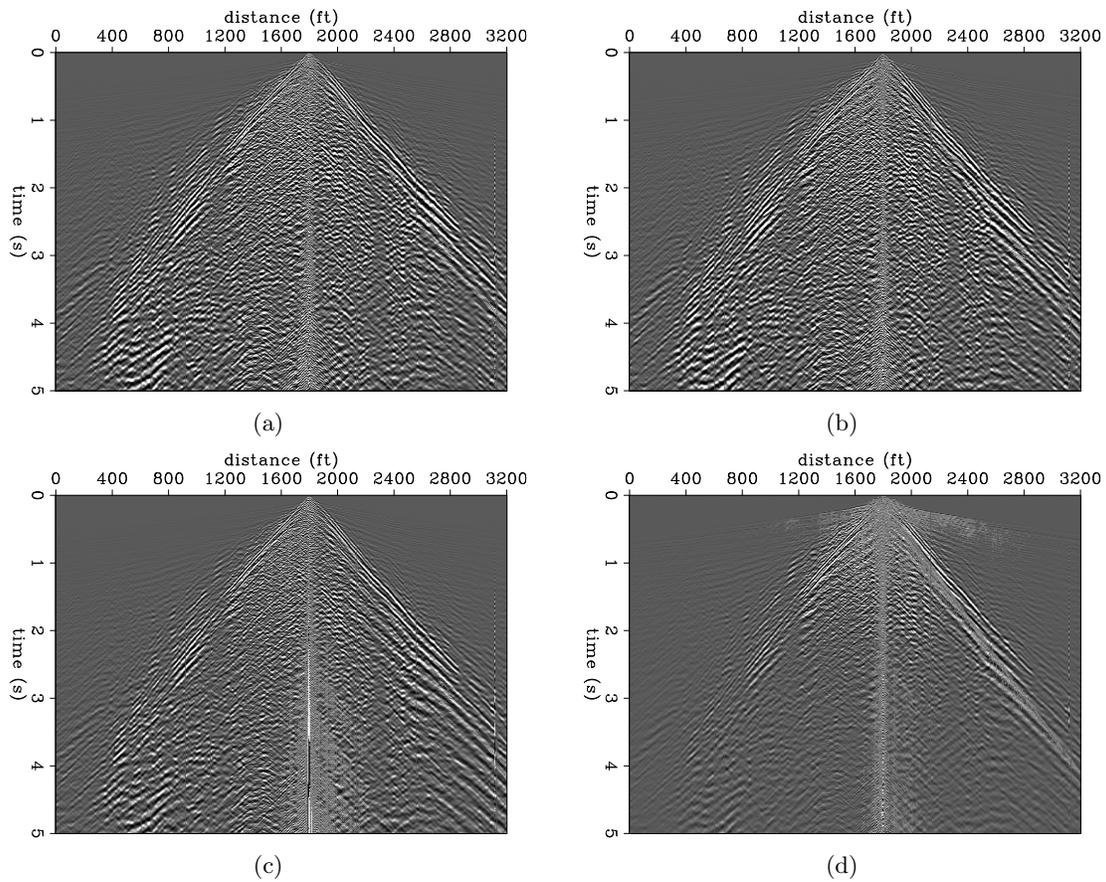


Figure 8: Shot gather from line 6001 after (a) autocorrelation, (b) autocorrelation and minimum phase correction, (c) source wavelet deconvolution, and (d) PEF deconvolution after autocorrelation and minimum phase correction. [CR]

farris1/. corr2d,corrmin2d,decon2d,wiener2d

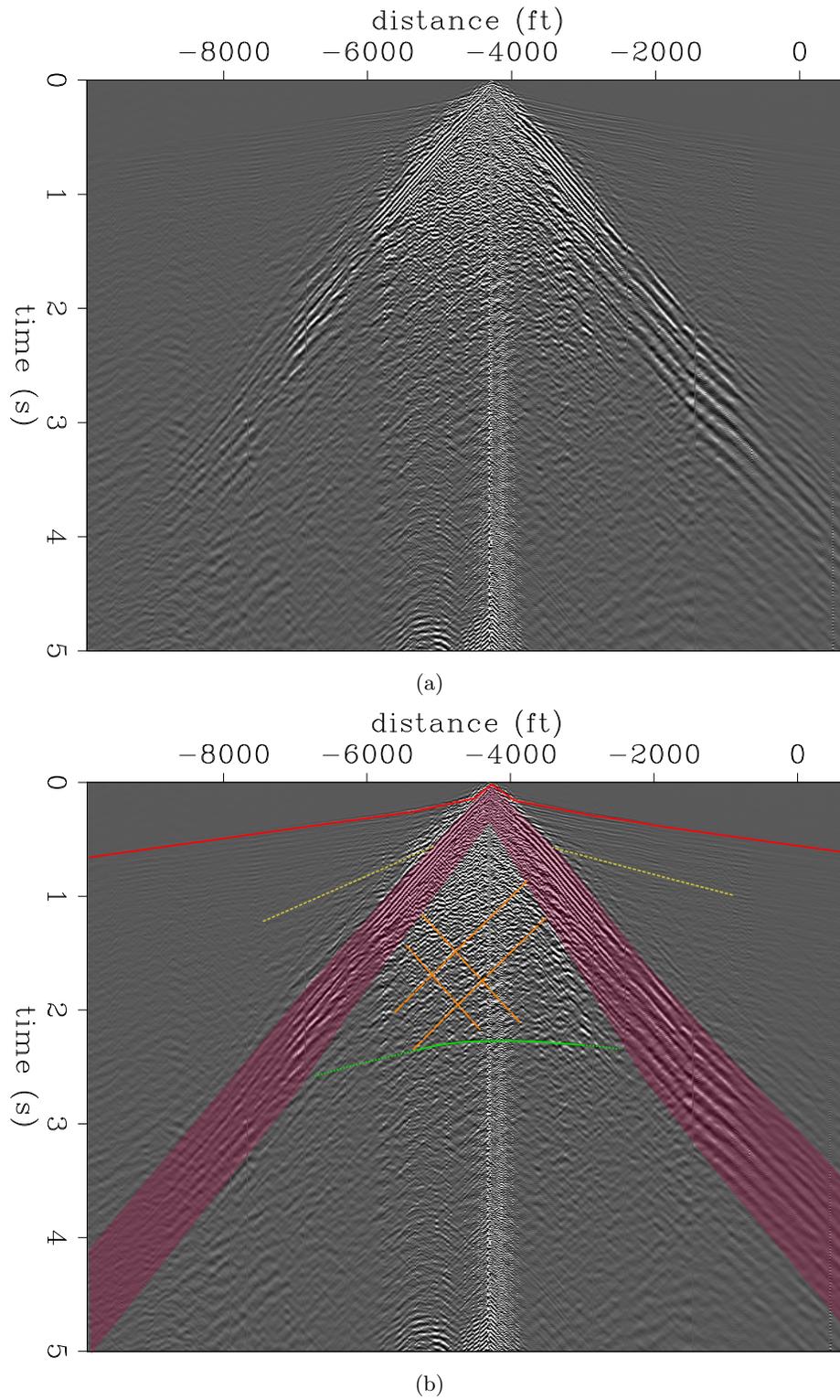


Figure 9: Identified waves in shot 33 before fk filtering: (a) without picks, and (b) with picks. [NR] `farris1/. shot33,shot33pick`

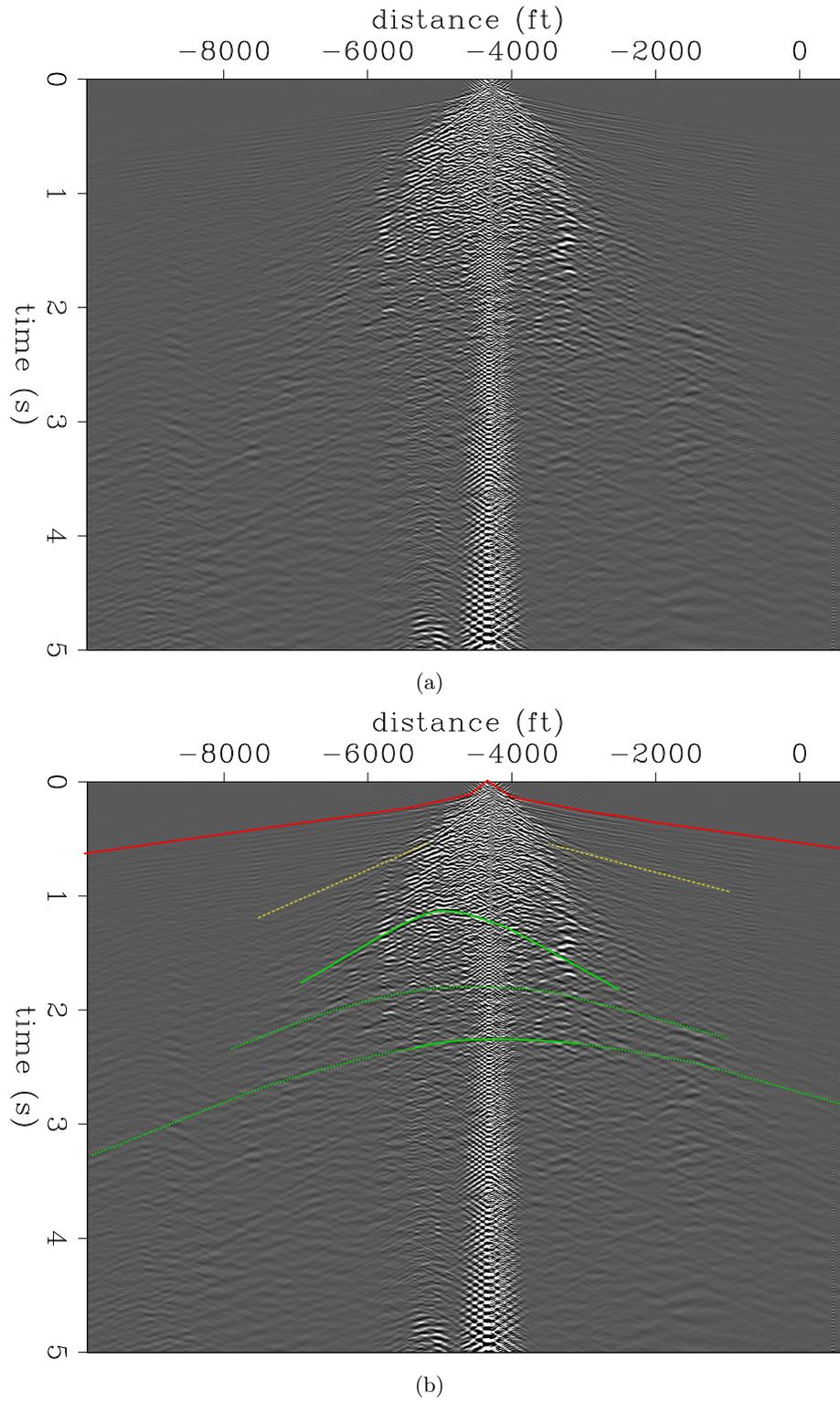


Figure 10: Identified waves in shot 33 after fk filtering: (a) without picks, and (b) with picks. [NR] `farris1/. shot33.fk,shot33fkpick`

WHAT'S NEXT

These efforts only mark the beginning of the possible investigations into this experimental dataset. There are many possible avenues for future researchers to explore. A short list includes:

1. Writing a gain program that is a function of offset.
2. Performing static corrections.
3. Performing Normal Moveout velocity analysis.
4. Comparing quality of gathers produced from linear, low frequency, and dwell sweeps.
5. Modeling surface waves.
6. Attempting full waveform inversion.

ACKNOWLEDGMENTS

We would like to acknowledge Occidental Petroleum for providing the SEP with the data used in this report. A special thanks to Biondo Biondi, Shuki Ronen, Mohamed Hadidi, and Stew Levin for guidance through the messy world of onshore seismic data. Furthermore, none of this report would have been possible without Jason Chang's extensive knowledge of SEPlib and its unique 3D data manipulation. Finally, the sponsors of the SEP should be recognized for their financial support.

REFERENCES

- Al-Ali, M. N., 2007, Land seismic data acquisition and preprocessing: an operator solution to the near-surface problem: [info:eu-repo/semantics/doctoralthesis](http://info.eu-repo/semantics/doctoralthesis).
- Claerbout, J., 2014, Geophysical Image Estimation By Example.: LULU COM. (OCLC: 986953183).
- Fagan, M. N., 1997, Resource Depletion and Technical Change: Effects on U.S. Crude Oil Finding Costs from 1977 to 1994: *The Energy Journal*, **18**, 91–105.
- Yilmaz, ., 2001, Seismic Data Analysis. Investigations in Geophysics: Society of Exploration Geophysicists. (DOI: 10.1190/1.9781560801580).

Multichannel data: separating independent causes

Jon Claerbout and Kaiwen Wang

ABSTRACT

The algorithm for blind deconvolution of a nonstationary time series of vector components (*i.e.* multichannel) has three stages: (1) Linear-least-squares multichannel prediction-error filtering, (2) Cholesky factorization of the zero-lag covariance matrix, and (3) Rotation angle scanning for maximum sparsity.

INTRODUCTION

The “blind deconvolution” problem for a vector-valued signal is shown in Figure 1. In practice $x_1(t)$ and $x_2(t)$ may be different wave types that mix and register on our two-component ($y_1(t), y_2(t)$) instruments.

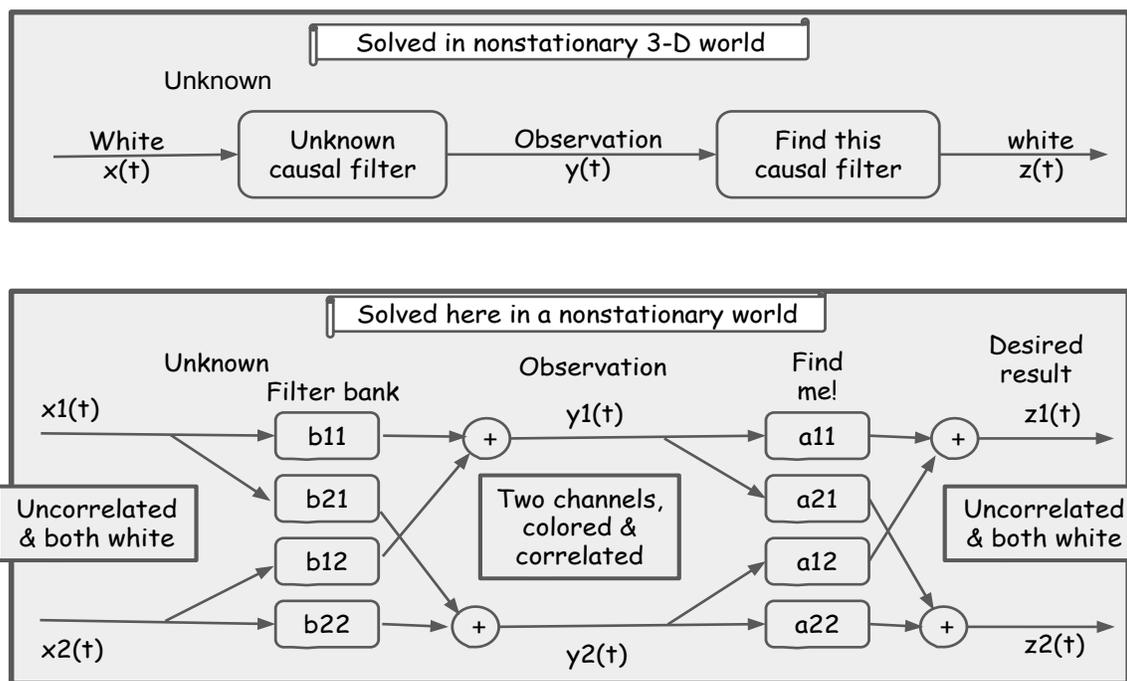


Figure 1: The left side of the diagram hypothesizes nature outside our view. Given the correlated observations \mathbf{y} in the middle we here design the causal process \mathbf{A} to create the outputs \mathbf{z} on the right. We construct uncorrelated white \mathbf{z} , then hope it approximates the underlying physical world \mathbf{x} , and that $\mathbf{B} \approx \mathbf{A}^{-1}$. [NR] `jon1/. FindMatrixFilterTransposeDone`

The multichannel structure of Figure 1 arises in diverse physical settings. For example underwater measurements may represent waves upgoing and waves downgoing, but we don't

measure them directly; we measure pressure and displacement. Or the earth may contain pressure waves and shear waves while we measure vertical and horizontal motions. Waves may arrive at our multicomponent recorders from two directions. If your multicomponent recorder records two different things, like pressure and velocity, your channels will have differing spectral characteristics. That's good. It's a central aspect of this model.

Fourier analysis suggests a crude approach to Figure 1. For scalar waves, given the spectrum $Y(\omega)^*Y(\omega)$ the solution to the problem is $A(\omega) = 1/\sqrt{Y(\omega)^*Y(\omega)}$. But this implies a symmetric function of time, not causal. Fourier space requires stationary statistics, forbids ℓ_1 -norm. The square root of a matrix of Fourier functions is easily found, but the disadvantages of Fourier space overwhelm the simplicity of the time domain. Causality is easily expressed with Z -transforms, equivalently either a matrix of them, or a polynomial of matrix coefficients.

Theory behind Figure 1 appeared in engineering literature half a century ago. I gave it little attention because I had no data of vector-valued signals. The old theory for such signals also depended on the geophysically unrealistic assumption of stationarity. My google search for terms like "adaptive multichannel filter theory" did not turn up methods recognizably suitable for my community of geophysical data analysts (although I believe it should be out there somewhere in the engineering literature). Then I stumbled onto appropriate methodology for non-stationary signals. It is much easier to put into practice than the stationary theory. Hooray! Our research group began getting multicomponent (vector-valued) data. Here I put all the pieces together and am looking for people who want to try it.

The first stage is multichannel prediction-error theory. Here the approach to matrices of filters learns from the old theory of stationary methods (but nonstationarity makes it *much* easier). PEFs (prediction-error filters) remove all lagged correlations from the data. But at zero lag there remains crosscorrelation between the channels. That's easily dealt with by the Cholesky method.

Intriguing is what comes last, something wholly unfamiliar. Even after solving the problem posed in Figure 1, the solution is unique only within an arbitrary unitary matrix. But this two channel problem, although nonlinear, is easily amenable to a one-parameter exhaustive search. That search can be done to maximize sparsity of the final signals. We humans love the simplest representation of our data. This should be it. Hooray!

In related good news it looks like this final unitary process could overcome a perennial problem in geophysical data processing. Since the processing stream is inexpensive it is easy to program a wide-ranging exploration of the process parameter space. The sparsity measure could then tell how to choose the best processing parameters. Hooray again! A complete, but untested code, for the only subtle part of the whole process, the vector-signal PEF, is found at the end of this document.

Range of applicability

This two-component signal model is not suitable for two scalar signals recorded at separate locations. If you are processing a string of multicomponent recorders (down a well, for example) each multicomponent recorder yields statistics which may be shared and averaged

with neighboring recorders, but the signals themselves would not mix, so, given these statistics, the process described here is simply a time-variable linear operator. This mathematical model is based on causality and simultaneity of the two channels responding to the outside world.

The model here naturally extends to three and more physical dimensions. Whether it is suitable for many-channel market signals I cannot say.

If the underlying model \mathbf{B} were to introduce delay, its hypothetical inverse \mathbf{A} would need to contain inverse delay (non-causality!). Then the methods of this paper must fail. In marginal cases (tiny delay) the notion of sparsity has helped for scalar signals. Its a promising area beyond our present scope.

SCALAR NONSTATIONARY SIGNALS

To my surprise and delight, I find nonstationary signal analysis simpler and more amenable to practice than textbook stationary theory.

Nonstationary prediction without mathematics(!)

Start with data (a signal of thousands of values). Take any filter \mathbf{a} of maybe ten lags. The filter will change slowly as we slide it along the data. Set the filter down on the data. The ten data values under the filter are designated \mathbf{d} . Take ϵ to be a tiny scalar (for example $\epsilon = 1/(4000 \|\mathbf{d}\|)$).

At any one time instant the filter \mathbf{a} has output $(\mathbf{a} \cdot \mathbf{d})$. We set the goal that $(\mathbf{a} \cdot \mathbf{d})$ should be something we have chosen. We might have chosen another signal (shaping), or the next incoming data value (prediction). The filter output $(\mathbf{a} \cdot \mathbf{d})$ over-shoots or undershoots this goal. We will add or subtract a teeny ϵ amount of data \mathbf{d} to the filter \mathbf{a} , then see which of the \pm does the better job. Trying the modified filter $\mathbf{a} \pm \epsilon \mathbf{d}$ two possible predictions emerge.

$$(\mathbf{a} \pm \epsilon \mathbf{d}) \cdot \mathbf{d} = (\mathbf{a} \cdot \mathbf{d}) \pm \epsilon (\mathbf{d} \cdot \mathbf{d}). \quad (1)$$

Comparing these predictions to the hoped for value reveals which sign for ϵ better improves our filter. Update the filter \mathbf{a} . Move to time $t + \Delta t$. Update \mathbf{d} . Repeat indefinitely. The filter adapts to make the best fit your goal (often prediction, more often prediction error).

The above idea can be based on conventional math. The gradient of ℓ_2 normed prediction error \mathbf{e} turns out to be $\mathbf{d} \times \text{PredictionError}(\mathbf{d})$. Let $\text{signum}(e) = e/|e|$. Then the above algorithm amounts to stepping along with $\mathbf{d} \times \text{signum}(\text{PredictionError}(\mathbf{d}))$ which smells like nonstationary ℓ_1 norm decon. Wow!

Many variations of this theme are easy, such as constrained filters, gapped filters, predicting further ahead, or predicting other signals. We mostly use these ideas for making prediction error (a white signal). We make that by constraining the first filter coefficient to be +1 so all the rest are effectively predicting negatively to try extinguish the data value under the +1. In the limit of very many iterations and $\epsilon \rightarrow 0$ the result tends to that of stationary theory.

Any color found in the prediction-error output should have been usable to enhance the prediction. Presuming it did, the result is optimal prediction-error output. This tends to whiteness, being limited only by the number of filter coefficients and the non-zero size of ϵ .

Of academic interest the relation between the ℓ_1 norm and the ℓ_2 norm methods is easy suggesting that a wide range of other norms and penalty functions are easily attained in the same way. The heart of the matter seems to be choosing any function of the components of \mathbf{d} that is polarity preserving (so when dotted into \mathbf{d} assures a positive scalar). So it seems the $\ell_{1/2}$ norm is as easy as augmenting the filter \mathbf{a} with a vector of components $\epsilon d_i/|d_i|^{3/2}$. Naturally my favorite is the softclip function, the derivative of a hyperbolic penalty function.

Pseudo code for scalar signals

In all least-squares data fitting, the residual goes into the adjoint to produce the gradient direction. Stripped of details, ℓ_2 -norm scalar-signal code is

```

a(1) = 1.0          #          Syntax:  "a+=b" means "a=a+b"
do for all time t  # e = nonstationary prediction error
  do tau= 1, na
    e(t)  +=  a(tau) * y(t-tau+1)      # forward
  do tau= 2, na
    da(tau) +=  e(t)  * y(t-tau+1)    # adjoint
  do tau= 2, na
    a = a - epsilon * da

```

Arrays in Fortran/Matlab range from $a(1)$ to $a(n)$ while in C/C++/Java they range from $a(0)$ to $a(n-1)$. Matrix operations are more naturally expressed in Fortran/Matlab while polynomial and convolution operations are more naturally expressed in C/C++/Java. While it may be natural to express each concept in its favored language, my pseudo codes were confusing until I stuck to just one language convention. Since I chose Fortran/Matlab the math idea $\int_0^\infty a(\tau)y(t-\tau)d\tau$ is rendered `do tau=1,na{a(tau)*y(t-tau+1)}`.

The `#forward` code line computes the prediction residual $e(t)$ at some time t . At that time, the first `tau` loop is performing the dot product $(\mathbf{a} \cdot \mathbf{d})$ mentioned earlier with \mathbf{d} being a backwards running chunk of data $y(t-\tau)$. The `#adjoint` loop corresponds to a matrix transposed because compared with the `forward` loop, input and output have swapped their roles. Their `tau` loops have differing ranges. This because the $a(1)=1.0$ produces the prediction *residual*, so $a(1)$ is not changed.

Linguistically, it would be more correct to call $(-1, \cdot, \cdot, \dots)$ the prediction-error filter, but nobody wants to use that filter because its output polarity is opposite that of the original data. So the PEF is defined as $(+1, a_1, a_2, \dots)$.

Gradient derivations

Formal theory underlies the idea of adding an ϵ bit of \mathbf{d} to the filter. The algebra to show that $\mathbf{d} \times \text{PredictionError}(\mathbf{d})$ is the gradient that arises from a new data point is in Appendix I.

Sergey Fomel and I developed a complicated, subtle nonstationary PEF theory. It led to an update direction with a conceptual distance parameter. Then I discovered its implementation amounts to a simple-minded step in the direction of the gradient. Operationally, the two methods had turned out to be one and the same! The “simple minded conceptual parameter” amounts to our epsilon ϵ .

Easy question: Given that seismic data is typically gained with t^2 , how should gain and decon work together?

How big is epsilon?

Epsilon ϵ is the fractional change to the filter at each stage of iteration. In a process called leaky integration, any value at time t is altered by a fractional amount ϵ during transition to $t + \Delta t$. Every smoothed value is diminished by $(1 - \epsilon)$, and then updated by ϵ times its current estimated value. After λ steps any value is reduced by the factor $(1 - \epsilon)^\lambda$. Setting that to $1/e = 1/2.718$ says $1/e = (1 - \epsilon)^\lambda$. Taking the natural logarithm, $-1 = \lambda \ln(1 - \epsilon) \approx -\lambda\epsilon$, so to good approximation

$$\epsilon = 1/\lambda \tag{2}$$

By the well known property of exponentials, half the area in the decaying signal appears before the distance λ , the other half after.

In casual discussion I think of this memory function as a rectangle function of length λ . Least squares analysis begins with the idea there should have more regression equations than unknowns. So λ should roughly exceed the number of filter coefficients \mathbf{na} . To avoid overfitting, I'd begin with $\lambda = 10 \times \mathbf{na}$.

I have more thoughts on choosing λ , but this is too early to present them. The nonstationary environment is such a strong component of many valuable applications that reports of wise and clever choices for ϵ , if they cannot be found now, they are sure to arise soon.

VECTOR NONSTATIONARY SIGNALS

In scalar signal analysis, it is known that the prediction-error signal $e(t)$ is white Claerbout (2014) (page 182). Its autocorrelation is a delta function. Something similar (but intriguingly different) happens with vector-valued signals. Vector and scalar cases are based on causality. Innovations arrive simultaneously on both channels. The two channels may show different spectra, but the method fails when one component has been delayed with respect to the other, so don't try it with two scalar channels recorded at different locations.

Nonstationary vector-valued signals require a three-stage process. The first stage mimics the prediction-error process of scalar signals. That eliminates time-lagged correlations. The next stage, the Cholesky stage, eliminates zero-lagged crosscorrelation between the two channels; and it scales the channels to unit variance. At the last stage notice the vector process defined by Figure 1 has multiple solutions. From any vector solution, others follow by any unitary matrix \mathbf{U} transformation. (With scalar-signals the arbitrariness is in a scale factor $e^{i\phi}$.) We get to choose the \mathbf{U} having minimum entropy \mathbf{z} output. Unexpected. Intriguing!

Nonstationary variance/covariance

Equation (3) defines a running variance $\sigma_y^2(t)$ of the signal $y(t)$. Such a recursive process is called leaky integration. Likewise (4) defines a running crosscorrelation between two channels.

$$\sigma_y^2(t) = (1 - \epsilon) \sigma_y^2(t - \Delta t) + \epsilon y(t)^2 \quad (3)$$

$$\sigma_{y_{12}}^2(t) = (1 - \epsilon) \sigma_{y_{12}}^2(t - \Delta t) + \epsilon y_1(t) y_2(t) \quad (4)$$

$$\sigma_{\mathbf{y}}^2(t, \tau) = (1 - \epsilon) \sigma_{\mathbf{y}}^2(t - \Delta t) + \epsilon \mathbf{y}(t) \mathbf{y}(t + \tau)' \quad (5)$$

Likewise (5) defines a 2×2 matrix of running lagged covariance where $\mathbf{y}(t)$ is a two-component column vector while $\mathbf{y}(t + \tau)'$ is a likewise a row. Stationary time-series theory actually displays the 3-D lagged covariance $\sigma_{\mathbf{y}}^2(\tau)$ (FGDP page 140).

Our 2-component PEF marching along the time axis updating prediction-error filters should chew up the lagged correlations. That provably happens with scalar signals Claerbout (2014) (page 182). Thus the PEF output $\mathbf{e}(t)$ has a 3-D covariance that vanishes at nonzero lags. We are left with the zero lag, a nice 2×2 matrix of prediction-error variances \mathbf{W} .

$$\mathbf{W}(\tau = 0) = \begin{bmatrix} \sigma_{e_{11}}^2 & \sigma_{e_{12}}^2 \\ \sigma_{e_{21}}^2 & \sigma_{e_{22}}^2 \end{bmatrix} \approx \begin{bmatrix} (\mathbf{e}_1 \cdot \mathbf{e}_1) & (\mathbf{e}_1 \cdot \mathbf{e}_2) \\ (\mathbf{e}_2 \cdot \mathbf{e}_1) & (\mathbf{e}_2 \cdot \mathbf{e}_2) \end{bmatrix} \quad (6)$$

The dot products are an oversimplification intended to clarify the meaning of leaky integration to new users. The dot products are also a handy way to initialize the update expressions.

Scalar signal scaling

A length measurement $\lambda = 1/\epsilon$ (in pixels) measures the averaging region. Thus both λ and ϵ are without physical units, though one might say λ has “units” of pixels. Prediction filters are dimensionless because from voltage, they predict voltage. Hence the units of $\epsilon \mathbf{d}$ match those of σ_d^2 . The properly scaled ℓ_2 update expression is

$$\Delta \mathbf{a} = - \left(\frac{\epsilon e}{\sigma_d^2} \right) \mathbf{d} \quad (7)$$

Replacing the prediction error e by its signum function yields an ℓ_1 -norm prediction after restoring nondimensionality by changing σ_d^2 to σ_d .

$$\Delta \mathbf{a} = - \left(\frac{\epsilon \text{signum}(e)}{\sigma_d} \right) \mathbf{d} \quad (8)$$

Let σ_e be a running standard deviation of prediction error. Now I’m feeling an ℓ_2 -norm filter update slightly more consistent than equation (7) is

$$\Delta \mathbf{a} = - \left(\frac{\epsilon e}{\sigma_e \sigma_d} \right) \mathbf{d} \quad (9)$$

Understanding physical units with scalar signals leads next to vector signal scaling.

Vector signal scaling

The leading coefficient of a vector signal PEF (prediction-error filter) is an identity matrix. The two 1's in the \mathbf{I} pass through the observed data $\mathbf{y}(t)$. Coefficients under all the other lags adapt to negatively predict it so as to get minimal output.

When components of data or model are out of scale with one another, bad things happen: The adjoint operator will not be a good approximation to the inverse. Physical units may be contradictory. Steepest descent creeps along slowly. These dangers would arise with vector-valued signals if the observations y_1 and y_2 had different physical units such as pressure and velocity recorded from up- and down-going waves. Or such as uncalibrated vertical and horizontal seismograms.

One could devise the filter updates by an effort of deep thought while inspecting Figure 1, but it's easier and more reliable to blindly base your updates on the negative adjoint of forward modeling. But, we do need to think about channels being out of scale with one another. Thus we scale each component of data \mathbf{y} and residual \mathbf{e} by dividing out their variances as we did in equation (9). Recall that any component of a gradient may be scaled by any positive number. Such scaling is merely a change in coordinates.

This is a good time to read the code at the end of this article.

Averaging in time and space

The code contains leaky integrations to assure the filter \mathbf{A} varies smoothly in time. Actually, the leaky integrations may smooth over both time and space. In other words, when updating an old filter $\mathbf{A}(t - \Delta t, x)$, we could update the old filter located at $\mathbf{A}(t, x - \Delta x)$. That would be learning over x while filtering over t . More generally, an update could leap from a weighted average over time and space. For example, we could update $\mathbf{A} \leftarrow \overline{\mathbf{A}} + \Delta \mathbf{A}$ with

$$\overline{\mathbf{A}} = \mathbf{A}(t - \Delta t, x) \frac{\lambda_t^2}{\lambda_t^2 + \lambda_x^2} + \mathbf{A}(t, x - \Delta x) \frac{\lambda_x^2}{\lambda_t^2 + \lambda_x^2} \quad (10)$$

Notice that the weights sum to unity. The averaging region is an area roughly $\lambda_x \lambda_t$ pixels squared in size. The coding requires not only saving \mathbf{A} at the previous time, it requires a saved \mathbf{A} for every time at $x - \Delta x$.

Stationary decon should remove a shot waveform. Nonstationary decon starts from there but has the added possibility of removing the waveform of the outgoing wave. That evolves with travelttime (Q and forward scattered multiples). It also evolves with space, especially receiver offset. We could build such nonstationary filters on either field data or synthetic data, then apply them to field data. The relations among pressure, velocity, upcoming, and downgoing waves vary systematically with offset. You could work out theoretical expressions for these relations, but instead you could see how this data fitting code would handle it.

How can the nonstationary PEF operator be linear?

Let \mathbf{E} be the prediction-error operator and \mathbf{e} its output. By definition

$$\mathbf{e} = \mathbf{E} \mathbf{y} \quad (11)$$

The operator \mathbf{E} may seem to be a nonlinear function of the data \mathbf{y} . But it is nearly linear, even strictly linear in a certain sense. Notice that \mathbf{E} could have been built entirely from spatially nearby data, not at all from \mathbf{y} . Then \mathbf{E} would be nonstationary, yet a perfectly linear operator on \mathbf{y} .

I am no longer focused on conjugate-gradient solutions to stationary linear problems, but if I were, I could at any stage make two copies of all data and models. The solution copy would evolve with iteration while the other copy would be fixed and would be used solely as the basis for PEFs. Thus the PEFs would be changing with time while not changing with iteration. This makes the optimization problem a linear one, fully amenable to linear methods. In the spirit of conjugate gradients (as it is commonly practiced), on occasion we might restart with an updated copy. People with inaccurate adjoints often need to restart. (Ha ha.)

CHOLESKY DECORRELATING AND SCALING

The two independent channels of unit-variance random numbers in \mathbf{x} *entering* filter \mathbf{B} in Figure 1 have the identity matrix \mathbf{I} as a covariance. Here we arrange to have the same identity covariance for the values \mathbf{z} *exiting* from \mathbf{A} on the right.

Consider the expectation (leaky sum over time) $E[\mathbf{e}\mathbf{e}']$. Theoretically it's a 3-D function of lag and the two channels. We're going to assume our PEFs are perfect so that it is no longer a function of lag. Thus we presume that $E[\mathbf{e}\mathbf{e}']$ is like the \mathbf{W} we computed with equation (6) at zero lag τ .

$$E[\mathbf{e}\mathbf{e}'] = \begin{bmatrix} \sigma_{e_{11}}^2 & \sigma_{e_{12}}^2 \\ \sigma_{e_{21}}^2 & \sigma_{e_{22}}^2 \end{bmatrix} = \mathbf{W} \quad (12)$$

Use the Cholesky method to factor \mathbf{W} into a triangular matrix \mathbf{V} times its transpose, so $\mathbf{W} = \mathbf{V}\mathbf{V}'$. (The Cholesky method is nearly trivial: Write a triangular matrix of unknown elements. Multiply it by its transpose. Notice a sequential method that unravels the unknown elements.)

$$\mathbf{W} = \mathbf{V}\mathbf{V}' \quad (13)$$

$$\mathbf{V}^{-1}\mathbf{W}(\mathbf{V}')^{-1} = \mathbf{I} \quad (14)$$

$$\mathbf{C}\mathbf{W}\mathbf{C}' = \mathbf{I} \quad (15)$$

where we have defined $\mathbf{C} = \mathbf{V}^{-1}$. Using this new matrix operator \mathbf{C} we get a new vector signal \mathbf{q} .

$$\mathbf{q} = \mathbf{C}\mathbf{e} \quad (16)$$

The expectation of this new variable \mathbf{q} is

$$E[\mathbf{q}\mathbf{q}'] = E[\mathbf{C}\mathbf{e}\mathbf{e}'\mathbf{C}'] \quad (17)$$

$$= \mathbf{C}E[\mathbf{e}\mathbf{e}']\mathbf{C}' \quad (18)$$

$$E[\mathbf{q}\mathbf{q}'] = \mathbf{C}\mathbf{W}\mathbf{C}' = \mathbf{I} \quad (19)$$

This shows Cholesky does for us two things: (1) it descales, and (2) it decorrelates \mathbf{e} at zero lag.

ROTATING FOR SPARSITY

The most intriguing part of the entire process arrives at this the last stage. As the universe marches on, things get mixed and entropy increases. We seek the opposite.

Rotations and reflections are called unitary operators. For now we are ignoring reflections (polarity changes). (Consider that to be an application labeling issue.) Scanning a single parameter θ through all angles allows us to choose the one with the most sparsity (least clutter). A general form for a 2×2 rotation operator is

$$\mathbf{U} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \quad (20)$$

We will meet our goal of finding \mathbf{A} and \mathbf{z} of Figure 1 with:

$$\mathbf{z} = \mathbf{U}\mathbf{q} = \mathbf{U}\mathbf{C}\mathbf{e} = \mathbf{U}\mathbf{C}\mathbf{E}\mathbf{y} = \mathbf{A}\mathbf{y} \quad (21)$$

A unitary operator \mathbf{U} does not change the length of any vector. It satisfies $\mathbf{U}'\mathbf{U} = \mathbf{I}$, so for any \mathbf{v} we see $(\mathbf{U}\mathbf{v})'\mathbf{U}\mathbf{v} = \mathbf{v}'\mathbf{U}'\mathbf{U}\mathbf{v} = \mathbf{v}'\mathbf{v}$. Let us check that the covariance of $\mathbf{z} = \mathbf{U}\mathbf{q}$ is constant independent of θ . Equation (19) leads to

$$\mathbf{z}\mathbf{z}' = \mathbf{U}\mathbf{E}[\mathbf{q}\mathbf{q}']\mathbf{U}' = \mathbf{U}\mathbf{I}\mathbf{U} = \mathbf{I} \quad (22)$$

This is saying the energy stays constant as we sweep through θ .

Finding the angle of maximum sparsity (minimum entropy)

Given any angle θ for equation (20) we have $\mathbf{z} = \mathbf{U}\mathbf{q}$. We can scan θ over one degree increments. Defining the entropy at any particular time as $(|z_1| + |z_2|)/\sqrt{z_1^2 + z_2^2}$ we easily choose the angle of minimum entropy for that time.

The more difficult question is dealing with noise. We want estimates based on time averages. Ultimately, we will have arrays of vector valued signals. We will also want local averages in the space of the arrays. At the deadline for this progress report, we are not certain we have properly dealt with the issue of estimating a best angle by forming averages over time and space.

Jon's theory, apparantly defective

The code below is a guess at the solution to the problem of averaging the Cholesky output $\mathbf{q}(t)$ over time to finally find a best angle for rotation. I define z_1 and z_2 by leaky integrating over time t the four quantities $q_1(t) \cos \theta$, $q_2(t) \sin \theta$, $q_1(t) \sin \theta$, $q_2(t) \cos \theta$, which contain all the parts of the product of vector \mathbf{q} multiplying the matrix of (20). Then I define entropy at time t by $(|z_1| + |z_2|)/\sqrt{z_1^2 + z_2^2}$. Finally, I scan all angles for the minimum entropy, and choose that angle θ .

```
initialize {q1cos,q2sin,q1sin,q2cos}(1:360)=0, entropy(1:360)=0
do over all time t {
  # You insert steps "q = C E y" here.
```

```

do ith= 1, 360 {
  th = 2 * 3.1416 * (ith-1)/360.
  q1cos(ith) = (1-epsilon)*q1cos(ith) + epsilon*( q1(it)*cos(ith))
  q2sin(ith) = (1-epsilon)*q2sin(ith) + epsilon*( q2(it)*sin(ith))
  q1sin(ith) = (1-epsilon)*q1sin(ith) + epsilon*( q1(it)*sin(ith))
  q2cos(ith) = (1-epsilon)*q2cos(ith) + epsilon*( q2(it)*cos(ith))
  z1 = q1cos(ith) + q2sin(ith)
  z2 = -q1sin(ith) + q2cos(ith)
  entropy(ith) = (abs(z1) + abs(z2)) / sqrt( z1*z1 + z2*z2 )
}
ithbest = 1      # Find the best theta
do ith= 1, 360
  if( entropy(ith) < entropy(ithbest)) ithbest = ith
# You put theta(ithbest) into the U matrix and make "z(t) = U C E y(t)"
}

```

Kaiwen's theory: works on easy synthetics

The code below is to find a best angle for each time step to rotation. It follows Stew Levin's suggestion to apply phase unwrapping afterwards to avoid switch or flip of trace. We define u_1 and u_2 by leaky integrating over time t the l_1 and l_2 norm of (z_1, z_2) . Then We define entropy at time t by u_1/u_2 . Finally, We scan all angles for the minimum entropy, and choose that angle for the wrapped θ . We see that the period of θ is $\pi/2$, so we choose the jump tolerance to be $\pi/4$ and correct phase change.

Matlab code:

```

u1=zeros(360,1);
u2=zeros(360,1);
entropy=zeros(360,1);
for i=na+1:nt
for ith=1:360
  theta=2*pi*(ith-1)/360;
  u1(ith)=(1-epsilon)*u1(ith)+epsilon*norm( ...
[q(1,i)*cos(theta)+q(2,i)*sin(theta) -q(1,i)*sin(theta)+q(2,i)*cos(theta)],1);
  u2(ith)=(1-epsilon)*u2(ith)+epsilon*norm( ...
[q(1,i)*cos(theta)+q(2,i)*sin(theta) -q(1,i)*sin(theta)+q(2,i)*cos(theta)]);
  entropy(ith)=u1(ith)/u2(ith);
end
[~,I]=min(entropy);
theta=2*pi*(I-1)/360;
theta_wrapped(i)=theta;
end
theta_unwrapped=theta_wrapped;
for j=2:length(theta_unwrapped)
  difference = theta_unwrapped(j)-theta_unwrapped(j-1);
  while abs(difference) > pi/4
    if difference > pi/4
      theta_unwrapped(j:end) = theta_unwrapped(j:end) - pi/2;
    elseif difference < -pi/4
      theta_unwrapped(j:end) = theta_unwrapped(j:end) + pi/2;
    end
    difference = theta_unwrapped(j)-theta_unwrapped(j-1);
  end
end
end
for i=na+1:nt

```

```

U=[cos(theta_unwrapped(i)) sin(theta_unwrapped(i)); ...
   -sin(theta_unwrapped(i)) cos(theta_unwrapped(i))];
z(:,i)=U*q(:,i);
end

```

Why the scan works

Why does this \mathbf{U} process of scanning θ lead to sparsity? Suppose the vector signal element \mathbf{q}_N at time at $t = N$ has all its energy in its first component. Say the vector signal is $[-1, 0]'$ with energy and magnitude both now equal unity. The rotated signal is now

$$\begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} -1 \\ 0 \end{bmatrix} = \begin{bmatrix} -\cos \theta \\ \sin \theta \end{bmatrix} \quad (23)$$

Let the rotation angle be 45° so sine and cosine are both $1/\sqrt{2}$. The sum of the magnitudes becomes $2/\sqrt{2} = \sqrt{2} > 1$. As expected the rotation took away the original sparsity.

3-component vector data

For 3-component vectors the scan would run over two angles so the `u(itheta)` would be expanded to `u(itheta, iphi)`.

NOT QUITE MINIMUM DELAY

We have solved the spectral factorization problem for nonstationary vector-valued signals. It might seem to be a unique solution. But it is not. If the physics \mathbf{B} includes delays, we won't find their inverse delays in our computed \mathbf{A} because our \mathbf{A} is causal. Presuming that $\mathbf{z} = \mathbf{x}$ amounts to making the so-called "minimum-phase" assumption for vector-valued signals. I don't know the general form for matrix causal all-pass (pure delay) filters but I do know it for scalar filters, and it suggests a very big world of delay possibilities.

But tiny delays are not uncommon in practice and may be overcome. Particularly instructive is my experience with scalar signals:

Deconvolution has been an industry standard for half a century. Conventionally, it is an ℓ_2 -norm process that makes a minimum-phase assumption. This means the inverse filter should be causal. It is obvious to everyone that the shot waveform is causal, but that should not be confused with its inverse. A dangerous ingredient is the marine ghost. Its heart is a second-difference operator δ_{tt} so its inverse involves a growing ramp! So, like many others, I set out to look for a causal shot waveform with a noncausal inverse. This isn't easy, but with Antoine Guitton (and earlier work with Yang Zhang and Yi Shen) we invoked the concept of sparsity. To do so takes us beyond the ℓ_2 norm. We got amazing results:

On all the data tested, four field data sets, and a fifth modeled at a sponsoring company, we were able to show images having events with signal polarity that was self evident. The first multiple would have a polarity obviously opposite the primary. A white reflection denoted a hard reflector, a black one denoting a soft one, so bottom of salt was easily recognizable by the polarity change. I found it thrilling be able to view many other events on seismic data in such a geologic manner.

MISCELLANY

Avoiding near Nyquist confusion

PEFs seem to focus equally on all frequencies up to the Nyquist. Sometimes analysts choose to focus more on the lower range. This can be done by “gapping”, namely, constraining to zero a few filter coefficients near the leading 1.

Leaky integration alternatives

The choice of an area in which to gather statistics is fairly subjective. From equation (10) and the leaky integration expression we may wonder the contours of the 2-D integration response. Contours of the product $e^{-t/\lambda_t - x/\lambda_x}$ in (t, x) are a simple triangle, so other than the sharp corner, it feels reasonable though very basic.

Sometimes I feel I'd be more comfortable if I could say statistics had been uniformly weighted before decay sets in. I could easily make exact boxes with $(1 - Z^N)/(1 - Z)$ but I don't like the sharp truncation after N lags. I also don't like the sharp corner at the beginning of the exponential. Perhaps something like a box $(1 - Z^N)/(1 - Z)$ terminated by a damped exponential $+Z^N/(1 - (1 - \epsilon)Z)$. Those two outputs could simply be added. Alternately, those two polynomial ratios could be added, then rearranged to the single ratio $(1 - \rho Z - (1 - \rho)Z^{(N+1)}) / (1 - (1 + \rho)Z + \rho Z^2)$. Multiply the numerator times the Z transform of x_t , namely $X(Z)$, and the denominator by $Y(Z)$, and identify time lags with powers of Z obtaining the recursion:

$$y_t = (1 + \rho)y_{t-1} - \rho y_{t-2} + x_t - \rho x_{t-1} - (1 - \rho)x_{t-(N+1)} \quad (24)$$

For $N = 2$, $\rho = .5$, and $x_3 = .99$, I obtain $y_t = .00 .00 .99 .99 .99 .49 .25 .12 .06 .03$.

TEST CASES

To compare inputs with outputs, display as 50 channels per sheet of wiggle trace in 10 groups of 5, the 5th being a dead trace to clarify display.

Easiest case, mixing, but no filtering

```
polarity = 1.
do i=1,1000,40 { # jump in steps of size 40
  x1(i) = 2.
  x2(i+20) = polarity
  polarity = -polarity
}
```

$$\mathbf{B} = \begin{bmatrix} 1 & -.3 \\ .2 & 1 \end{bmatrix} \quad (25)$$

The unscrambling would all be done by Cholesky and Unitary.

A case with an obvious answer

I'm not sure the method of this paper should unscramble it. I'm not really sure what these methods should be capable of, but this one should be really impressive if it works.

$$\mathbf{B} = \begin{bmatrix} 1./(1 - .6Z) & -.3/(1 - .9Z) \\ .2/(1 - .6Z) & 1./(1 - .9Z) \end{bmatrix} \quad (26)$$

For more fun, we might prefer wavelets that oscillate.

Horizontal phones, two far away signals, near each other, one stronger, the other 5× weaker coming in at slightly different angles

$$\mathbf{x} = \begin{bmatrix} S \\ W \end{bmatrix} = \begin{bmatrix} \text{random numbers} \\ \text{random numbers} \end{bmatrix} \quad \text{for } i = 1, 10000 \quad (27)$$

$$\mathbf{B} = \begin{bmatrix} 5/(1 - .6Z) & 1/(1 - .9Z) \\ 4/(1 - .6Z) & 1/(1 - .9Z) \end{bmatrix} \quad (28)$$

SUGGESTIONS AND RANDOM THOUGHTS

Gapped filters for oversampled data

Seismic data is typically oversampled on time. This implies that relatively, the upper half of the bandwidth may be mostly noise. Could we improve the wave-type separation by altering it in some way? We might design PEFs for prediction distances greater than one Δt sample.

A non-causal PEF approximation

A PEF is causal, has a white output, and is able to fit any spectrum. Extending that PEF to allow some negative lags quickly loses the white-out aspect, and we would be using more coefficients than needed to represent any spectrum. At the same time, we'd like to extend the PEF into negative lags to enable it to attack the δ_{tt} inherent to most seismograms. So, we'd like a few negative lags and we are often happy to give up a few positive lags. With non-stationary methods, testing has become much easier. We should try out the filter $(a_{-2}, a_{-1}, 1, 0, 0, a_3, a_4, a_5, \dots)$.

Four component data

Some data is sampled with four components, three velocity directions and pressure. This process would output four channels. This sounds redundant. What might we learn from the weakest channel?

Step sizes

How large should be the down-gradient hops? One first guess is to compare successive hops. The polarity of the dot product of the two successive gradients gives the indicator. If they are in the same direction, we might increase the step size. If in opposite directions, we would then decrease it. Is this a sensible strategy? If so, it could be widely used broadly in data fitting, in applications having little to do with statistics. A diverse collection of tests could be intriguing.

In fitting seismic waveforms to field data, we know the data is 95% repeatable, while we are astonished if any theory can drive the fitting residual down by as much as 50%. Does it make sense to struggle with second derivatives whose purpose is to define the location of perfect fit? Shouldn't we focus on more efficient ways of driving down the gradient? For example, gradients are functions of locations in space and frequency, so instead of looking for a global distance to hop, we should think of ways to break up the gradient into parts that can hop separately with different sized hops. Non-stationary PEFs might do some of that.

What good is a 3-D PEF?

Although it's clear how to fit 3-D PEFs to data, I doubt the utility of it. When I see 3-D data, (t, x, y) , I visualize it containing planes. A plane in 3-D looks like a line in both (t, x) and (t, y) space. It's more efficient to fit two planes each with a 2-D PEF $[a(t, x), a(t, y)]$ than with a single 3-D PEF $a(t, x, y)$. What kind of 3-D fields require 3-D PEFs? I don't know.

Segregating elastic waves

Naturally, we will attempt separation of pressure and shear waves with these multichannel methods. These methods compete, however with traditional scalar signal methods based on the idea that S waves tend to have double the stepout of P , namely namely, $v^2 = (x/t)(dx/dt)$. This opens the door to fake recording channels.

Fake channels

Might multichannel technology be made more serviceable by faking extra channels as functions of the first? For example: $y_3(t, x) = y_1(t, x + \Delta x) - y_1(t, x - \Delta x)$.

I have a feeling oceanographers are well along in this area. Do a youtube search for "perpetual ocean" to see an awesome video. Physics involves curl and divergence, both of which motivate investigating the statistics of neighboring vector-field measurements. This video might look as though they have spatially dense measurements. Perhaps so, but we might also be seeing a well-crafted vector field made from coarser measurements.

DAS string

SEP's new DAS string introduces us to the uncomfortable notion of having one component of a vector quantity but not the other. It suggests a new form of missing data problem to solve. I'm not advocating we try solving it yet on our string, but I am ready to start discussing the problem.

Consistent wavelet polarities

Greg Beroza reminds us of repeating earthquakes. Especially small quakes may repeat with the same polarity. We should think about whether and how such phenomena can be best recognized.

CONCLUSION AND OPPORTUNITIES

In theory (hopefully) the nonstationary vector spectral factorization problem is solved. Its main application is segregating wave types in multicomponent data.

Of equal or greater interest is the solution methodology. Although introduced here in 1-D, it is generally applicable to higher dimensional data. It introduces nonstationary decon to many areas including regridding, missing data, and whitening inversion residuals.

- No need for synthetics. The simplicity of the nonstationary technology invites immediate experimentation with multidimensional field data.
- Robust applications (such as using the ℓ_1 norm) will become far more common.
- Sparse model estimation is an easy extension of the usual ℓ_2 -norm regularization.
- The multidimensional filter is like a small molecule. It is built on statistics from a larger, differently shaped, region of data.
- New avenues arise for handling waveform variation over (shot-receiver) offset.
- Experimentation with ϵ will lead to deeper understanding of nonstationarity.
- Finite difference representation of differential equations (curl, divergence) might induce very weak non minimum phase that can be overcome via sparsity.

ACKNOWLEDGEMENT

I'd like to thank Enders Robinson for teaching me 55 years ago about stationary multi-channel time series. Multicomponent seismograms were rare then (and imaging soon to be more fun). I thank Kaiwen Wang for showing my initial unitary transformation was totally wrong. I thank Joseph Jennings for thoughtful comments on early versions of this manuscript. He picked up numerous errors and identified several incomplete explanations. I'd also like to thank Mohamed Hadidi who helped me take a derivative with respect to a matrix. I also thank Carl Wunsch who brought the perpetual-ocean video to my attention.

REFERENCES

Claerbout, J., 2014, Geophysical image estimation by example: Lulu.com.

APPENDIX I: GRADIENT DERIVATION

I proposed the nonstationary problem (29) to Sergey Fomel who solved it. He solved it with powerful algebraic tools. The detailed algebra is found in earlier work by Fomel and Claerbout. Then I discovered that a simple step down the gradient led to the same updating of the filter. Though differing philosophically [(1) deviate from stationarity, and (2) step down the gradient], both methods update the filter with $\Delta \mathbf{a} = -\epsilon e \mathbf{d}$. Both philosophies obligate the practitioner to choose a suitable numerical value for ϵ . Finally, while preparing a lecture, I realized what you saw near the beginning of this paper that a simple approach almost devoid of mathematics leads to $\Delta \mathbf{a} = \pm \epsilon \mathbf{d}$ which amounts to a nonstationary ℓ_1 -norm PEF. Below I derive algebraically the ℓ_2 -norm gradient.

Start with any old PEF. We are going to improve it a tiny amount by considering just one new data value d_{n+1} . Call the old PEF $\bar{\mathbf{a}} = (1, \bar{a}_1, \bar{a}_2, \bar{a}_3, \dots)$. The updated PEF \mathbf{a} will be made by moving a small distance down the gradient (opposite polarity of the gradient).

Consider the regression:

$$\begin{bmatrix} d_{n+1} & d_n & d_{n-1} & d_{n-2} \\ \lambda & \cdot & \cdot & \cdot \\ \cdot & \lambda & \cdot & \cdot \\ \cdot & \cdot & \lambda & \cdot \\ \cdot & \cdot & \cdot & \lambda \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \approx \begin{bmatrix} 0 \\ \lambda \\ \lambda \bar{a}_1 \\ \lambda \bar{a}_2 \\ \lambda \bar{a}_3 \end{bmatrix} \quad (29)$$

The top block says we seek a PEF \mathbf{a} , that should improve fit of the newly arrived data value d_{n+1} . The lower block needs a large numerical value for λ to limit the amount of filter change. Define the fitting residual \mathbf{r}

$$\mathbf{0} \approx \mathbf{r} = \begin{bmatrix} d_n & d_{n-1} & d_{n-2} \\ \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} - \begin{bmatrix} -d_{n+1} \\ \lambda \bar{a}_1 \\ \lambda \bar{a}_2 \\ \lambda \bar{a}_3 \end{bmatrix} \quad (30)$$

Let

$$\mathbf{d} = \begin{bmatrix} d_n \\ d_{n-1} \\ d_{n-2} \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

With these definitions and \mathbf{I} being an identity matrix the residual definition (30) is

$$\mathbf{0} \approx \mathbf{r} = \begin{bmatrix} \mathbf{d}^T \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{a} - \begin{bmatrix} -d_{n+1} \\ \lambda \bar{\mathbf{a}} \end{bmatrix} \quad (31)$$

We take the derivative of $\mathbf{r}^T \mathbf{r}$ to find the search direction.

$$\Delta \mathbf{a} = - (\text{some constant}) \left. \frac{\partial}{\partial \mathbf{a}^T} \right|_{\mathbf{a}=\bar{\mathbf{a}}} \mathbf{r}^T \mathbf{r} \quad (32)$$

Form the transpose of the residual (31), and then differentiate it by \mathbf{a}^T . (By \mathbf{a}^T we mean the complex conjugate transpose of \mathbf{a} .)

$$\frac{\partial \mathbf{r}^T}{\partial \mathbf{a}^T} = \frac{\partial}{\partial \mathbf{a}^T} \{ \mathbf{a}^T [\mathbf{d} \ \lambda \mathbf{I}] - [-d_{n+1} \ \lambda \bar{\mathbf{a}}] \} = [\mathbf{d} \ \lambda \mathbf{I}] \quad (33)$$

Multiply that onto \mathbf{r} from (31) keeping in mind that $\mathbf{d}^T \bar{\mathbf{a}}$ is a scalar and that the expression $(\mathbf{d}^T \bar{\mathbf{a}} + d_{n+1})$ is the prediction error e .

$$\frac{\partial \mathbf{r}^T}{\partial \mathbf{a}^T} \mathbf{r} = [\mathbf{d} \ \lambda \mathbf{I}] \left\{ \begin{bmatrix} \mathbf{d}^T \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{a} - \begin{bmatrix} -d_{n+1} \\ \lambda \bar{\mathbf{a}} \end{bmatrix} \right\} \quad (34)$$

$$= \mathbf{d}(\mathbf{d}^T \mathbf{a}) + \lambda^2 \mathbf{a} + \mathbf{d}d_{n+1} - \lambda^2 \bar{\mathbf{a}} \quad (35)$$

$$\left. \frac{\partial \mathbf{r}^T}{\partial \mathbf{a}^T} \right|_{\mathbf{a}=\bar{\mathbf{a}}} \mathbf{r} = (\mathbf{d}^T \bar{\mathbf{a}} + d_{n+1}) \mathbf{d} \quad (36)$$

$$\Delta \mathbf{a} = -\frac{1}{\lambda} e \mathbf{d} \quad (37)$$

Scale out the physical dimensions to see the filter update we've been using from the beginning of this paper.

APPENDIX II: RATFOR CODE (UNTESTED)

Compared with earlier pseudocode where the gradient is an unscaled adjoint, here the gradient has divided out the variances σ_e and σ_y . You may always scale gradient components by positive numbers.

```
# Non-stationary prediction error for vector signals in Ratfor/Fortran syntax
#
integer it, nt=1000, ia, na=10, gap=1, lambda=4000, ic, jc, nc=2
real y(nc,nt), e(nc,nt), aa(nc,nc,na), sige(nc), sigy(nc), eps
eps = 1./lambda
do ic=1,nc {
do jc=1,nc {
do ia=1,na {
aa(ic,jc,ia) = 0.
}}}
do ic=1,nc {
aa(ic,ic,1) = 1.
}
do ic=1,nc {
do it=1,nt {
e(ic,it) = 0.
}}
read input y(nc,nt)

do ic=1,nc {
sumsq=0
do it=1,nt
sumsq += y(ic,it)**2
sigy(ic) = sqrt(sumsq/nt)
sige(ic) = sigy(ic)
}
```

```

#                               Here we go! Happy streaming. Wheee!
do it= na, nt {

  do ic=1,nc
    e(ic,it) = 0.
  do ia=1,na {                  # lag axis.
  do ic=1,nc {                  # Vector into a matrix of filters.
  do jc=1,nc {                  #
    e(ic,it) += aa(ic,jc,ia) * y(jc, it-ia+1)
    }}}
                                # Running variance.
  do ic=1,nc { sigy(ic) = (1-eps)*sigy(ic) + eps*sqrt( y(ic,it)**2) }
  do ic=1,nc { sige(ic) = (1-eps)*sige(ic) + eps*sqrt( e(ic,it)**2) }

  do ia=gap+1, na {            # adjoint = e * y'
  do ic= 1, nc {               #
  do jc= 1, nc {               #
    aa(ic,jc,ia) -= eps * (e(ic,it)/sige(ic)) * ( y(jc, it-ia+1) /sigy(jc))
    }}}
}

```

Short Note: PEF swapping positive with negative lags

Stewart A. Levin

ABSTRACT

Jon Claerbout recently hypothesized that inserting negative filter lags opposite a gap in positive filter lags could produce interesting and useful filters. Here I prove that, indeed, such filters lead to controlled autocorrelation width and may provide properties beyond those of conventional gapped deconvolution.

INTRODUCTION

In SEP 150, Claerbout and Guitton (Claerbout and Guitton, 2013) found that adding a small number of anti-causal filter lags via the cepstral domain produced a pleasing deconvolution result. As Jon pointed out in PVI (Claerbout, 2004), simply adding negative lags to a time-domain PEF in order to obtain an interpolation filter is not generally a good idea as, in the limit, it produces an output which has a spectrum inverse to the input rather than white. This summer Jon speculated that moving a few positive (causal) lags to their negative (anti-causal) counterpart locations could perhaps produce nice looking PEF output, that is, output that does not arbitrarily boost all frequencies to and beyond whiteness, irrespective of their level of noise. I took up that challenge and calculated what swapping just the first lag would produce in the PEF's autocorrelation.

THEORY

Following the PEF whiteness proof of Levin et al. (2013), I minimize $\|\mathbf{r}\|_2^2$ by adjusting filter coefficients a_m in the residual

$$\mathbf{r} = a_{-1}Z^{-1}\mathbf{d} + \mathbf{d} + 0 + a_2Z^2\mathbf{d} + a_3Z^3\mathbf{d} + \dots \quad (1)$$

so that

$$\begin{aligned} 0 &= \frac{1}{2} \frac{d}{da_m} (\mathbf{r} \cdot \mathbf{r}) \\ &= \mathbf{r} \cdot \frac{d\mathbf{r}}{da_m} = \mathbf{r} \cdot Z^m \mathbf{d} . \end{aligned}$$

Let us now examine for an integer k the autocorrelation term $\mathbf{r} \cdot Z^k \mathbf{r}$.

$$\begin{aligned} k = 0 : \mathbf{r} \cdot Z^0 \mathbf{r} &= \cancel{a_{-1}\mathbf{r} \cdot Z^{-1}\mathbf{d}} + \mathbf{r} \cdot \mathbf{d} + \cancel{a_2\mathbf{r} \cdot Z^2\mathbf{d}} + \dots \\ k = 1 : \mathbf{r} \cdot Z^1 \mathbf{r} &= a_{-1}\mathbf{r} \cdot \mathbf{d} + \mathbf{r} \cdot Z\mathbf{d} + \cancel{a_2\mathbf{r} \cdot Z^3\mathbf{d}} + \dots \\ k = 2 : \mathbf{r} \cdot Z^2 \mathbf{r} &= a_{-1}\mathbf{r} \cdot Z\mathbf{d} + \mathbf{r} \cdot Z^2\mathbf{d} + \cancel{a_2\mathbf{r} \cdot Z^4\mathbf{d}} + \dots \\ k > 2 : \mathbf{r} \cdot Z^k \mathbf{r} &= \cancel{a_{-1}\mathbf{r} \cdot Z^{k-1}\mathbf{d}} + \mathbf{r} \cdot Z^k\mathbf{d} + \cancel{a_2\mathbf{r} \cdot Z^{k+2}\mathbf{d}} + \dots \end{aligned}$$

which says that the autocorrelation vanishes outside the first couple of lags. By comparison, a singly gapped causal prediction error filter has one nonzero autocorrelation lag. I leave to the reader to see that swapping the first n PEF coefficients leads to $2n$ nonzero autocorrelation lags.

DISCUSSION

As noted in the previously cited work, these calculations allow for an infinitely long prediction error filter and do not guarantee comparable behavior for finite length filters but should yield reasonable filter behavior for seismic data. Whether and when this approach might be superior to simple causal gapped prediction error filtering will be explored in future work. We do know that conventional gapped deconvolution reduces boosting of high frequency noise. The Claerbout and Guitton paper seems to suggest it will produce more symmetrical output as well and so could be an alternative to, e.g., phase-only Q compensation.

REFERENCES

- Claerbout, J. and A. Guitton, 2013, Ricker-compliant deconvolution: SEP-Report, **150**, 1–12.
- Claerbout, J. F., 2004, Earth Soundings Analysis: Processing versus Inversion: Stanford Exploration Project.
- Levin, S. A., J. Claerbout, and E. R. Martin, 2013, Shortest path to whiteness: SEP-Report, **150**, 13–16.

Short Note: Time variable prediction without mathematics

Jon Claerbout

ABSTRACT

With almost no math, a quick trick leads to ℓ_1 norm nonstationary decon.

SUMMARY AND CONCLUSION

Start with data (a signal of thousands of values). Start with any filter \mathbf{f} of maybe ten lags. The filter will change as we move it along the data. At time t set the filter down on the data. The ten data values under the filter are designated \mathbf{d} . Take ϵ to be a tiny scalar (for example $\epsilon = 1/(200 \|\mathbf{d}\|)$).

The filter \mathbf{f} has output $(\mathbf{f} \cdot \mathbf{d})$ that we may choose to be a prediction of the next data value to slide under the range of \mathbf{d} . The augmented filter $\mathbf{f} \pm \epsilon \mathbf{d}$ offers us the two predictions $(\mathbf{f} \pm \epsilon \mathbf{d}) \cdot \mathbf{d} = (\mathbf{f} \cdot \mathbf{d}) \pm \epsilon (\mathbf{d} \cdot \mathbf{d})$. Comparing these predictions to the actual incoming data value reveals which sign for ϵ better improves the prediction. Update the filter \mathbf{f} . Move to time $t + \Delta t$. Update \mathbf{d} . Repeat indefinitely. The filter adapts to best predict its incoming data.

THEORY AND POTENTIAL APPLICATIONS

The above idea can be based on conventional mathematics. The gradient of ℓ_2 normed prediction error turns out to be $\mathbf{d} \times \text{PredictionError}(\mathbf{d})$. The above algorithm marches with $\mathbf{d} \times \text{Signum}(\text{PredictionError}(\mathbf{d}))$ which smells like ℓ_1 norm decon. Wow!

Taking many small steps down a gradient has two advantages over analytic solutions: (1) it allows nonstationarity, and (2) it streams data (saving memory).

As with deconvolution on a helix, this method extends naturally to higher dimensional spaces (such as (t, x) -space).

I'm always trying to convince students to use PEFs to make their residuals IID.

I've explained how this method extends to multichannel data (vector-valued data), but it's not yet been tried. <http://sep.stanford.edu/sep/jon/VectorDecon.pdf>

C++11 non-linear solver

Robert G. Clapp, Stuart Farris, Taylor Dahlke, and Eileen Martin

ABSTRACT

Inverse problems such as velocity estimation from reflection/refraction data are inherently non-linear. We developed a library to address non-linear problems using C++11. We demonstrate the library on two simple examples.

INTRODUCTION

The last twenty years has seen an increasing percentage of SEP theses concerned with solving large inverse problems. During that period a large number of inversion libraries have been developed. Nichols et al. (1993) developed a framework using an early version of C++. For years many small inversions problems were handled by the Fortran90 library developed by Fomel and Claerbout (1996). Schwab and Schroeder (1997) developed a library using Java. For out of core and multi-node applications the python library described in Clapp (2005) was used. A second attempt at a C++ optimization library was described in Martin et al. (2014).

All of the above attempts concentrated on solving linear problems. The summer of 2015 saw the first significant attempt at building a framework for non-linear problems with a Fortran 2003 library described in Almomin et al. (2015). The library described in Almomin et al. (2015), while well designed, is limited to in-core problems and only accesible to Fortran programers. Biondi and Barnier (2017) addresses the first issue by emulating the design of Almomin et al. (2015) but written in python for out-of-core problems.

In this paper we descibe a non-linear version library written in C++11 using the same design as the Fortran library descibed in Almomin et al. (2015). We begin by reviewing the design principals of the library. We then present two simple inversion examples using the library. Finally, we discuss planned future additions to the library.

THE DOCKER ENVIRONMENT

One unique aspect of this project is the integration of the library that we've built with a portable environment system called Docker. Docker is a recent software product that holds an advantage over other environment replicating systems, and holds special promise for integrating with future SEP work in terms of reproducibility. One of the leading challenges for software reproducibility and deployment has been the difficulty in replicating the working environment. Virtual machines are a common way to circumvent this problem, but these machines can take minutes to launch, versus seconds (or less) with most Docker containers.

Creating a docker container for each SEP report would allow the reader to launch the same environment that the author used. Further, since most all SEP research is run on

linux-based systems, most containers would share the same parts of their container images. Docker is unique in that when multiple (similar) containers are run, the bulk of resources can be shared between them since they operate from the same base image. This makes the system light, and easier to deploy. For these reasons, we employ Docker in this report to showcase how the product can be used to enhance the reproducibility of SEP research.

One aspect of using Docker containers to replicate run environments is that we can easily build the environment that we want to use by means of a Dockerfile. An example of such a file shows how one can be used to setup the solver library that is used for the examples in this report, and would be used by you, the reader, to replicate the results you see here.

```

1 From rgc007/geelab:2017
2 MAINTAINER Bob Clapp <bob@sep.stanford.edu>
3 RUN yum -y install xorg-x11-server-Xvfb whichcmake boost-devel yum
  clean all
4 RUN dbus-uuidgen >/etc/machine-id
5 RUN git clone http://zapad.Stanford.EDU/SEP-external/gieeSolver.git
  /opt/gieeSolver
6 RUN git clone http://zapad.Stanford.EDU/bob/genericIO.git /opt/
  genericIO
7 RUN mkdir -p /opt/gieeSolver/build
8 RUN mkdir -p /opt/genericIO/build
9 RUN cd /opt/genericIO/build
10 RUN cmake -DCMAKE_INSTALL_PREFIX=/opt/genericIO -DSEPlib_DIR=/opt/
  SEP/lib ..
11 RUN make install
12 RUN cd /opt/gieeSolver/build
13 RUN cmake -DCMAKE_INSTALL_PREFIX=/opt/gieeSolver -DgenericIO_DIR=/
  opt/genericIO/lib -DBoost_INCLUDE_DIR=/usr/include ..
14 RUN make install

```

Listing 1: Sample Dockerfile

In 1, we first call the base image that we wish to build or install on. Afterwards, we can use the RUN operator to execute a series of commands that install programs and utilities on top of the starting image that we designate. In this case, we clone the libraries that are used for the examples in this report, and then install those libraries into the environment that we've specified. Once this container is built, we can save it as an image for later users to run without having to execute the script that holds these build commands. This interface makes creating and distributing environments (such as the one included with this report) a relatively simple task.

SOLVERS

This library was designed to have interfaces similar to the of Biondi and Barnier (2017). This means that we work with solver objects defined by a **problem**, a **stepper**, and a **terminator**. A **problem** object is associated with an objective we wish to minimize, and has methods associated with getting that objecting value for a particular model, getting gradients, data misfits/residuals, and managing the domain and range of a problem. A **stepper** decides for a given problem how far to move from the current model along a

certain update path (think of any line search). A **terminator** has a `test()` method that returns a boolean indicating whether to continue iterating when solving an inverse problem.

The solver class, called **nlsolver**, sets up a method to solve a given **problem**, using a particular **stepper**, and a specified **terminator**. All solvers have a method called `run()` that actually solves the problem using the stepper and terminator. But the solver class is abstract, so specific types of solver algorithms must be implemented. Many types of both linear and nonlinear solvers can be implemented in this framework. Two examples of solvers we have implemented are linear conjugate gradient, and nonlinear conjugate gradient. The linear conjugate gradient solver class, called **linSolverCG**, is one of the simpler optimization solver algorithms that can be implemented as a special case of a **nlsolver** because it only requires a **problem** and **terminator**, but does not need a **stepper** because the step length is predetermined in the usual way.

The nonlinear conjugate gradient solver class, called **nonlinSolverCG**, is a type of **nlsolver** that requires pointers to a **problem**, a **stepper**, a **terminator**, as well as a string indicating the method for calculating the CG update, β . As in Biondi and Barnier (2017), the user is only expected to interact with a **nonlinSolverCG** object through two methods: instantiation, and `run()`. Once `run()` is called, everything else happens behind the scenes based on what **problem**, **stepper**, **terminator** and β update were specified at the instantiation.

The steps happening behind the scenes when `run()` is called are:

1. Use `betaAssigner()` method to figure out which β update to use
2. Calculate g_0 , the gradient of the initial model guess, x_0
3. Set $d_0 = -g_0$, the first update direction
4. While **terminator**'s `test()` method says to continue iteration:
 - (a) Calculate g_{k+1} , the gradient of objective at the current model x_k
 - (b) Calculate scalar β_k using the specified β update
 - (c) Calculate the new update direction $d_{k+1} = -g_{k+1} + \beta_k d_k$
 - (d) Use the **stepper** to select α_k and update the model $x_{k+1} = x_k + \alpha_k d_k$
 - (e) $k = k + 1$

As in the library of Biondi and Barnier (2017), the nonlinear CG solver has multiple options for how to calculate β , so when a nonlinear CG solver object is instantiated, the user must provide a `std::string` referred to as its `betaMethod`, and a method called `betaAssigner` ensures the proper beta calculation happens. A user can simply use the method `run` with no additional parameters regardless of how β is calculated. Primarily following Hager and Zhang (2006), currently supported methods for β calculation are:

betaMethod	$\beta_k =$
FR	$\ g_{k+1}\ ^2 / \ g_k\ ^2$
PRP	$g_{k+1}^T y_k / \ g_k\ ^2$
HS	$g_{k+1}^T y_k / (d_k^T y_k)$
CD	$\ g_{k+1}\ ^2 / (-d_k^T g_k)$
LS	$g_{k+1}^T y_k / (-d_k^T g_k)$
DY	$\ g_{k+1}\ ^2 / (d_k^T y_k)$
BAN	$g_{k+1}^T y_k / (g_k^T y_k)$
HZ	$\left(y_k - 2d_k \frac{\ y_k\ ^2}{d_k^T y_k} \right)^T \frac{g_{k+1}}{d_k^T y_k}$
ST	$\beta = 0$

Note that ST is simply steepest descent.

RE-IMPLEMENTING GIEE

A secondary goal of the C++ nonlinear solver is to re-implement the linear solvers and operators transcribed in Geophysical Image Estimation by Example (GIEE), (Claerbout, 2014). While the ultimate goal of this solver is to address nonlinear problems, the lessons, examples, and experience in GIEE are invaluable for a budding geophysical image processor. Furthermore, many of the nonlinear problems addressed by the SEP are built around the linear ideas described in GIEE. Therefore, we deem it necessary to add GIEE to the C++ nonlinear solver.

Here we begin to re-implement GIEE by converting all of the in-text examples from Fortran to C++ one chapter at a time. These chapters are found in the solver directory `opt/gieeSolver/giee/`. Each chapter will contain a Makefile that can reproduce all of its figures. For example, Chapter One: Basic Operators and Adjoints illustrates the derivative and convolution operators using a few simple figures. To reproduce these figures, simply enter the Docker environment, as described above, move to the directory associated with chapter one, and run the appropriate make rule. Listing 2 illustrates these steps on the command line. The Figures 1 and 2 will then appear within the figure directory associated with chapter one, `/opt/gieeSolver/giee/ajt/Fig/`.

```

1 cd /opt/gieeSolver/giee/ajt
2 make stangrad90.v
3 make conv.v

```

Listing 2: GIEE Examples

We plan to re-implement all of the chapters from GIEE using the C++ nonlinear solver framework in ascending order.

INVERSION EXAMPLES

NMO operator

We demonstrate the solver library on a simple normal moveout (NMO) operator, where we invert for the slowness and time position of hyperbolic reflection events. Both examples use conjugate gradient inversion, with one example being regularized.

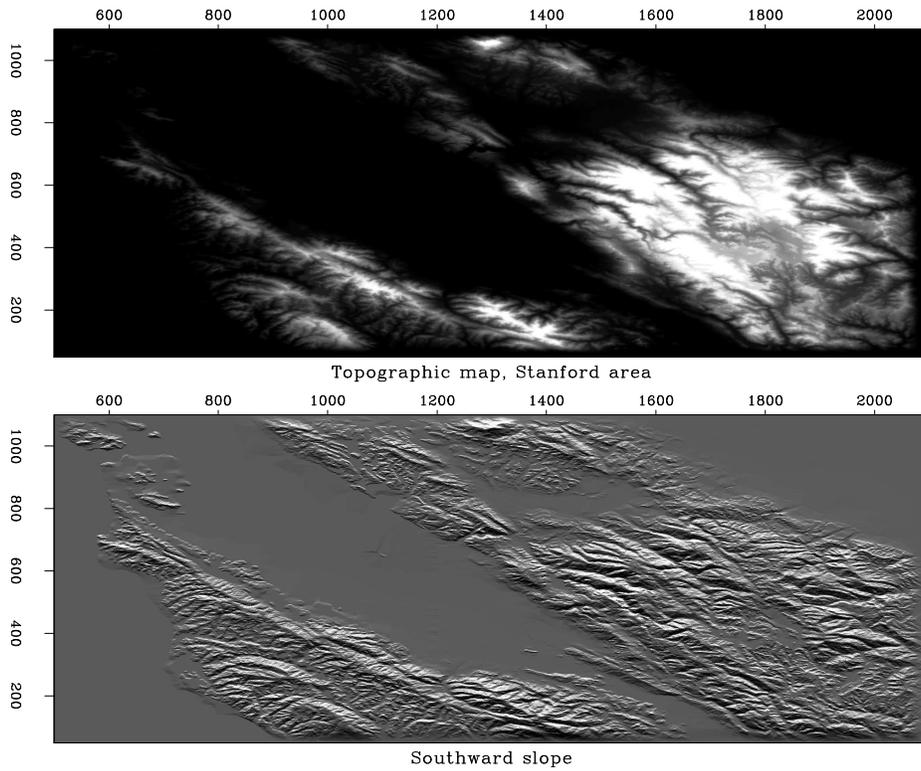


Figure 1: Illustration of derivative operator reimplemented from GIEE. [ER] bob1/. stangrad90

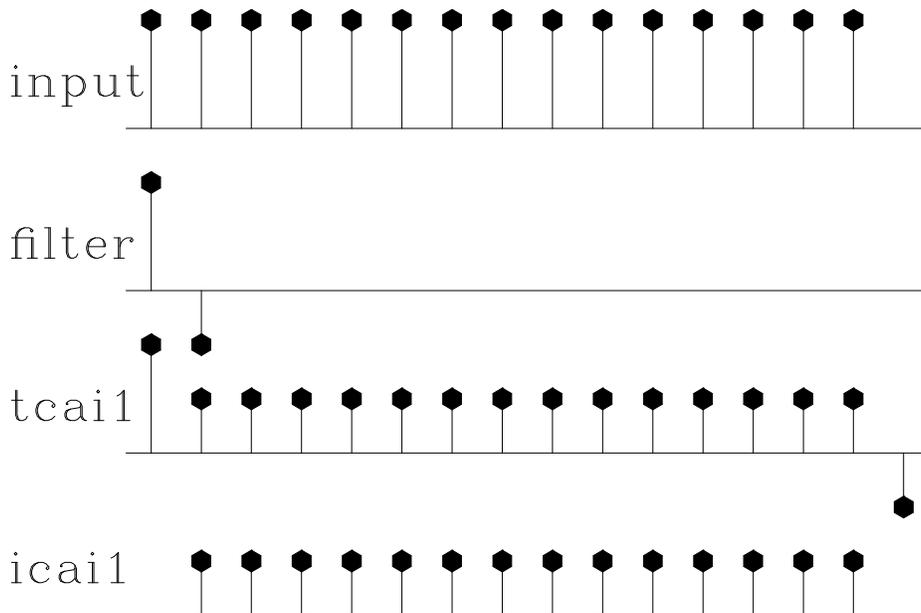


Figure 2: Illustration of transient and internal convolutions reimplemented from GIEE. [ER] bob1/. conv90

Conjugate gradient inversion

We begin with a true model as shown in Figure 3. From this we apply the forward NMO operator to create hyperbolas. We apply a smoothing to these hyperbolas to avoid inversion crime (Figure 4). We then apply 25 iterations of conjugate gradient inversion, beginning with an empty (all zeros) initial model. We get results that roughly match the true model in spatial extent as well as amplitude (Figure 5).

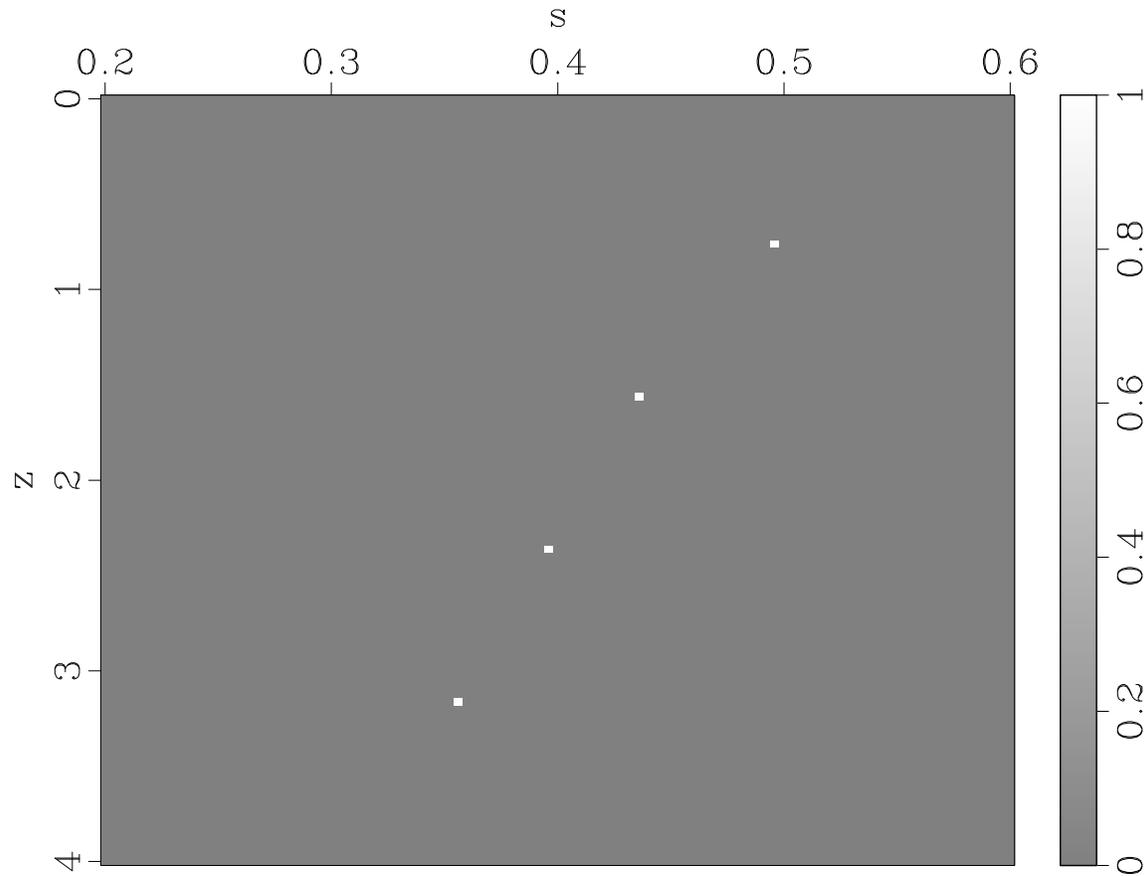


Figure 3: The true model showing the τ and s space representation of four hyperbolas.

[ER] bob1/. TrueModel

Regularized conjugate gradient inversion

For this case, we regularized the model space with a first order Laplacian smoothing operator. We set the parameter ϵ to 10.0, which controls the strength of the regularization term. Using a higher ϵ value means we will gain a smoother inverted result. Figure 6 shows the result of this regularized inversion. Note that this result is less spiky than the unregularized inversion result in Figure 5.

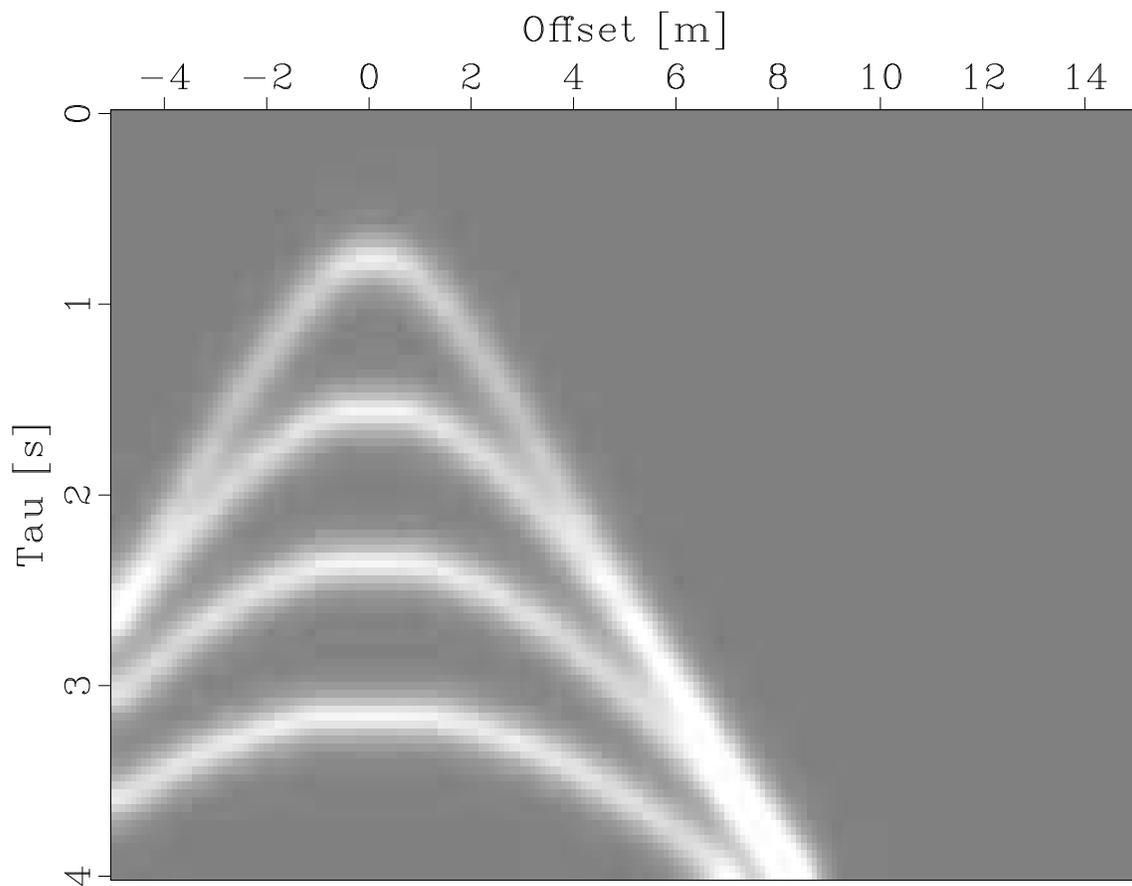


Figure 4: The initial data used for the CG inversion. In this case, we performed the forward operator on the true model and smoothed the result. [ER] bob1/. InitialData

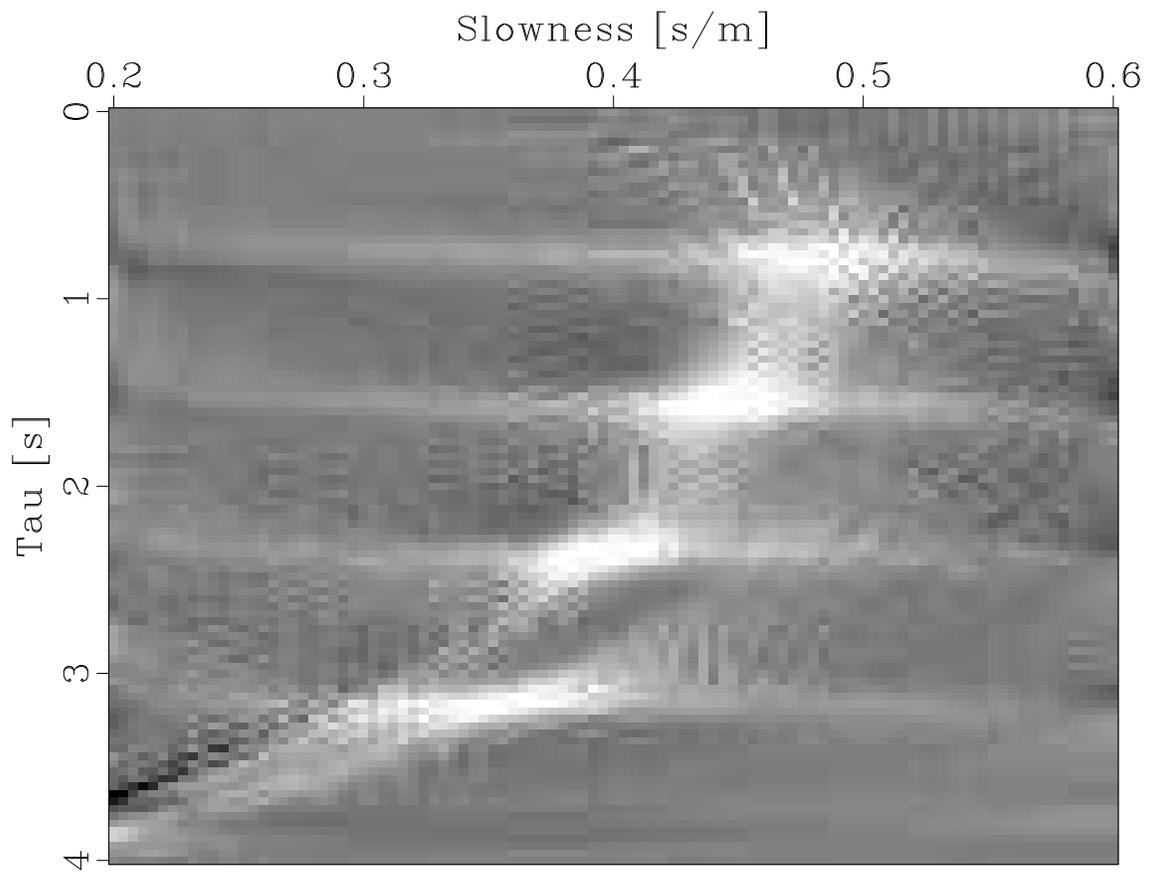


Figure 5: The result of performing 25 iterations of conjugate gradient inversion using the NMO operator. [ER] bob1/. InvertedModel

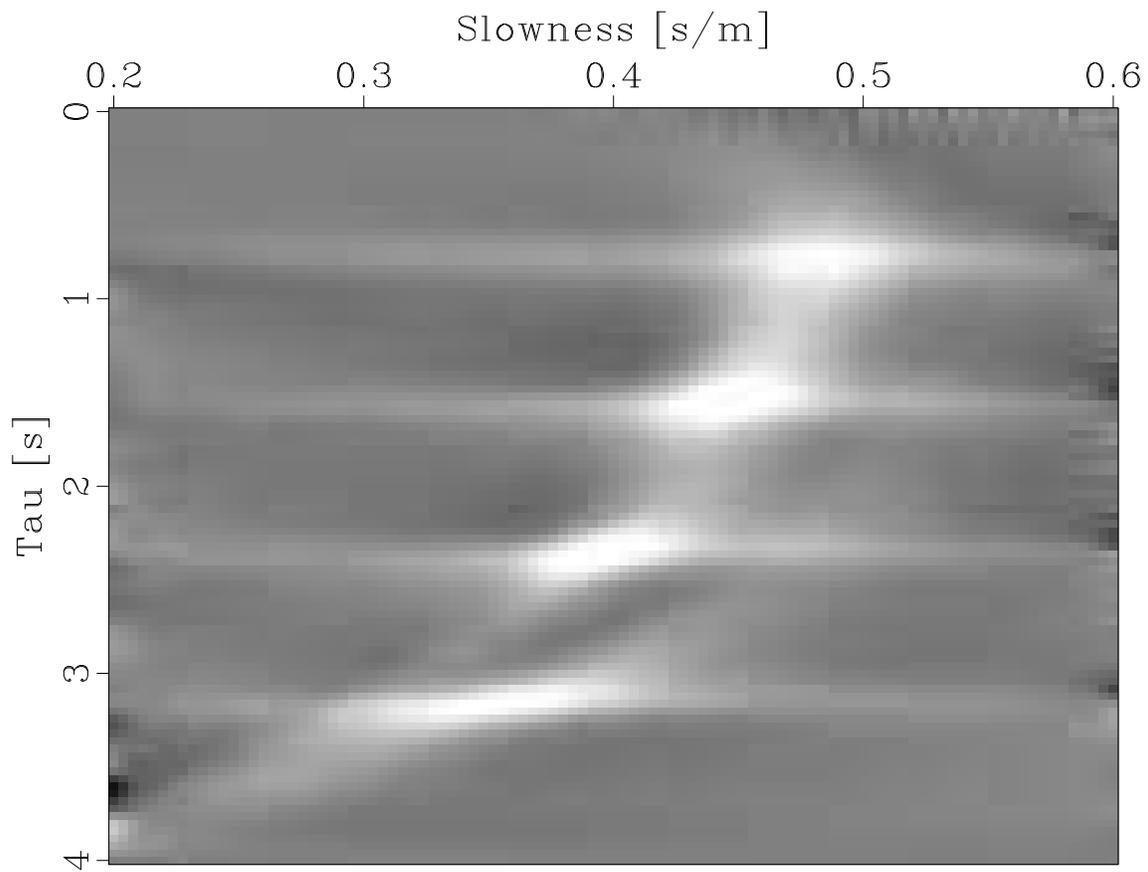


Figure 6: The result of performing 25 iterations of conjugate gradient inversion using the NMO operator and regularizing with the LaPlacian. [ER] bob1/. InvertedRegModel

FUTURE WORK

There are several ways that we want to grow this project. From a library perspective we need to add additional non-linear solvers and line search methods. We also need an expanded library of operators. This project started as way to begin the transition of both the book and class associated with Claerbout (2014) from using Fortran90 to C++. At this stage the labs for the class have been converted but many of the examples have yet to be finished. We are also considering adding a python interface to the library. This would allow the class to be taught in ipython notebooks while still introducing all of the concepts needed for students to later use the C++ library.

CONCLUSION

In this paper we described the current state of a C++ non-linear inversion library. It allows the user to solve to store in-core linear and non-linear problems using an approach similar to SEP's Fortran 2003 and python equivalents.

REFERENCES

- Almomin, A., E. Biondi, Y. Ma, K. Ruan, J. Jennings, R. Clapp, M. Maharramov, and A. Cabrales-Vargas, 2015, Seplib nonlinear solver library – manual: SEP-Report, **160**, 39–70.
- Biondi, E. and G. Barnier, 2017, A flexible out-of-core solver for linear/non-linear problems: SEP-Report, **168**, 1–15.
- Claerbout, J., 2014, GEOPHYSICAL IMAGE ESTIMATION BY EXAMPLE.: LULU COM. (OCLC: 986953183).
- Clapp, R. G., 2005, Inversion and fault tolerant parallelization using Python: SEP-Report, **120**, 41–62.
- Fomel, S. and J. Claerbout, 1996, Simple linear operators in Fortran 90: SEP-Report, **93**, 317–328.
- Hager, W. and H. Zhang, 2006, A survey of nonlinear conjugate gradient methods: Pacific Journal of Optimization, **2**, 35–58.
- Martin, E., R. G. Clapp, H. Le, C. Leader, and D. Nichols, 2014, SEPVector: A C++ inversion library: SEP-Report, **152**, 359–364.
- Nichols, D., H. Urdaneta, H. I. Oh, J. Claerbout, L. Laane, M. Karrenbach, and M. Schwab, 1993, Programming geophysics in C++: SEP-Report, **79**, 313–471.
- Schwab, M. and J. Schroeder, 1997, A seismic inversion library in Java: SEP-Report, **94**, 363–381.

SEPLib CMake update

Stewart A. Levin

ABSTRACT

SEPLib is now built with *CMake*. This report covers the further changes made to the source tree in order to fully support Linux and Mac OS X platforms.

INTRODUCTION

As discussed in Levin and Clapp (2017), our SEPLib source code was converted this year to *CMake* for more rapid and less arcane build development and maintenance. The timing of the switch proved rather awkward for me as I had only recently figured out how to build shared libraries needed for use with Java under the prior system. Diving back into *CMake* arcana, I retackled the SEPLib build process as well as fixed several open bugs. In the following I share highlights, tricks, and tips of that adventure.

CMAKE LOOSE ENDS

The unfinished portions of the SEPLib cmake-based build were

1. shared libraries,
2. symbolic links, and
3. executable scripts.

Shared libraries

Shared libraries, also known as dynamic libraries, are the basis for modern computing paradigms such as browser add-ins. They are loaded into memory at program run time rather than being statically linked into the program executable. This puts some extra burden on the developer. First, unlike traditional object file libraries, accessing one symbol, for example a subroutine, in the shared library brings all the library's symbols into the program. Usually these include undefined symbols that must be pulled in from another library. This means that

- all dependent libraries need to be known when a shared library is built,
- the same symbol should not be defined in both a shared library and any dependent shared library, and

- there should be no circular references among the library and its dependent shared libraries.

Fortunately, the shared library work I had done under the older GNU build system provided the lists for the first item and already fixed the latter two items.

As it happens SEPlib shared libraries are not needed for any SEPlib programs and, indeed, are a nuisance to deploy and use. The only part of SEPlib currently relying on shared libraries is the subset supporting Java. For this reason, SEPlib is actually built twice, once using shared libraries and a second time using static libraries. This results in static executables and library pairs, one dynamic and one static.

Shared libraries are not completely eliminated from SEPlib with static executable linkage. Quite a few system and other external libraries may only be available in shared library format and, furthermore, may be incompatible between different machines and operating system versions. For this reason, I also scan the static executables and make copies of their remaining dependent shared libraries underneath the installed library directory to be use as last resort fallback options.

Symbolic links

Symbolic links are suprisingly awkward to create in *CMake*. They are only provided for Unix-based operating environments, e.g. Linux and Mac OS X, and implemented by invoking *cmake* separately with the command line option `-E create_symlink` and arguments giving the old and new names. Somewhat confusingly, this is not the same as `ln -s` but mimics instead `ln -s -r` so that to create

```
/opt/SEP/lib/NewName.a ---> /opt/SEP/lib/OldName.a
```

one should invoke the command

```
cmake -E create_symlink /opt/SEP/lib/OldName.a NewName.a
```

either separately or, as done in SEPlib, as a subprocess during the installation.

Executable scripts

Scripts play a different role in *CMake* than executables. For one thing, they may require configuration of some variables or strings. In addition, it is a mistake to install them with the

```
install(FILEs scriptname DESTINATION bin)
```

cmake command as that relies on the script having the correct execute and read/write permissions in the source tree. One should use

```
install(PROGRAMS scriptname DESTINATION bin)
```

instead.

MISCELLANEOUS UPDATES

In addition to the three main areas discussed above, I also added support for FFTW high performance FFTs, resurrected older Motif-based 3D visualization programs, resolved a handful of vplot issues, added or modified external format converter utilities, and updated a number of program self-docs.

SUMMARY

While there are a few issues that still remain outstanding with respect to SEPlib builds and distribution, the current sources build quite cleanly on both Linux and Mac OS X and work under both GNU and Intel compilers.

REFERENCES

Levin, S. A. and R. G. Clapp, 2017, *make, schmake: CMake: SEP-Report*, **168**, 309–312.

SEP PHONE DIRECTORY

Name	Phone	Login Name
Akhmadiev, Rustam		arustam
Barnier, Guillaume	723-6006	gbarnier
Baroni, Claudia	723-5002	
Biondi, Biondo	723-1319	biondo
Biondi, Ettore	723-6006	ettore
Cabrales, Alejandro	723-9282	cabrales
Chang, Jason	724-4322	jasonpc
Claerbout, Jon	723-3717	jon
Clapp, Bob	725-1334	bob
Dahlke, Taylor	724-4322	taylor
Fabien-Ouellet, Gabriel		gfabieno
Farris, Stuart	723-6006	stuart
Ferrer, Miguel		
Huot, Fantine	723-0463	fantine
Jennings, Joe	723-1250	joseph29
Le, Huy	723-1250	huyle
Levin, Stewart	726-1959	stew
Ma, Yinbin	723-0463	yinbin
Martin, Eileen	723-0463	ermartin
Sarkar, Rahul	725-1625	rahul
Yuan, Siyuan	723-9282	syyuan

Dept fax number: (650) 725-7344

E-MAIL

Our Internet address is "*sep.stanford.edu*"; i.e., send Jon electronic mail with the address "*jon@sep.stanford.edu*".

WORLD-WIDE WEB SERVER INFORMATION

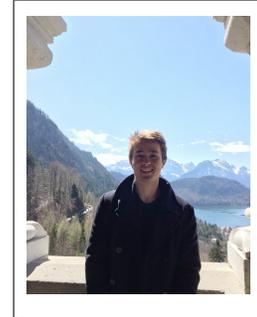
Sponsors who have provided us with their domain names are not prompted for a password when they access from work. If you are a sponsor, and would like to access our restricted area away from work, visit our website and attempt to download the material. You will then fill out a form, and we will send the username/password to your e-mail address at a sponsor company.

STEERING COMMITTEE MEMBERS, 2016-2017

Name	Company	Tel #	E-Mail
Raymond Abma	BP	(281)366-4604	abmar1@bp.com
Francois Audebert	TOTAL	(33)-6-1278-4195	francois.audebert@total.com
Biondo Biondi	SEP	(650)723-1319	biondo@sep.stanford.edu
Jon Claerbout	SEP	(650)723-3717	jon@sep.stanford.edu
Thomas Dickens	ExxonMobil	(713)431-6011	tom.a.dickens@exxonmobil.com
Carlo Fortini (Co-chair, 2nd year)	ENI	+39 02-520-61599	carlo.fortini@eni.com
Faqi Liu	Hess	(823)231-9218	fliu@hess.com
Yi Luo	Saudi Aramco	–	yi.luo@aramco.com
Dave Nichols (Co-chair, 1st year)	Schlumberger	–	nichols0@slb.com
Alejandro Valenciano	PGS	–	alejandro.valenciano@pgs.com

Research Personnel

Rustam Akhmadiev received his Bachelor's and Master's degree in Geophysics from Lomonosov Moscow State University, Russia in 2015. After that Rustam has got another Master's degree in Petroleum geophysics from IFP, France. Upon finishing his study in IFP in 2016, he was interning in Total's scientific research center in Pau, France working on higher-order finite-difference schemes used in seismic inversion. In July 2017 he started his Ph.D. program at Stanford Exploration Project. He is interested in elastic and anisotropic wave phenomena with applications in velocity models reconstruction and inversion.



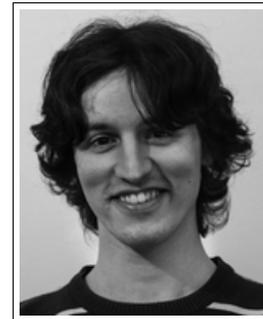
Guillaume Barnier is from Nice, France. He graduated from Telecom Paristech in 2007 with a MSc in telecommunications and signal processing. After working as a fixed income trader for JP Morgan (London) from 2007 to 2010, he decided to change career path by joining the geophysics MSc program in Colorado School of Mines (2011), where he focused his research on seismoelectric coupling in poro elastic media. In September 2013, he joined the Stanford Exploration Project to pursue his Ph.D. Guillaume focuses on improving the image quality of subsalt layers when they have been damaged by an inaccurate top salt interface delineation.



Biondo L. Biondi is professor of Geophysics at Stanford University. Biondo graduated from Politecnico di Milano in 1984 and received an M.S. (1988) and a Ph.D. (1990) in geophysics from Stanford. He is co-director of the Stanford Exploration Project and of the Stanford Center for Computational Earth and Environmental Science. In 2004 the Society of Exploration Geophysicists (SEG) has honored Biondo with the Reginald Fessenden Award. Biondo published a book, 3-D Seismic Imaging, that is the first text book to introduce the theory of seismic imaging from the 3-D perspective. The book is published by SEG in the Investigations in Geophysics series. During 2007 gave a one-day short course in 28 cities around the world as the SEG/EAGE Distinguished Short Course Instructor (DISC) . He is a member of AGU, EAGE, SEG and SIAM.



Ettore Biondi received a B.S. (2010) in geology from the University of Genoa, an M.S. (2012) in geophysics from the University of Pisa, and a diploma (2013) in computational chemistry from the Scuola Normale Superiore of Pisa. He spent almost two years as a research fellow at the University of Milan within the geophysics section of the Earth Science Department. In the summer of 2014, he joined the SEP and started his Ph.D. at the Stanford University. He is interested in multi-parameter inversion and wave-equation velocity estimation methods.



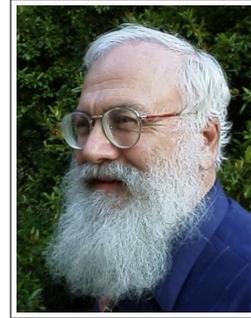
Alejandro Cabrales-Vargas obtained his Bachelors degree in Geophysics in the University of Mexico in 2002. He has been working for Petroleos Mexicanos since 2002, initially in seismic interpretation for oil and gas exploration, and more recently in the supervision of depth imaging processes. He obtained his Masters Degree in Geophysics in the University of Oklahoma in 2011. He joined SEP in the fall of 2014, and is currently working towards his PhD.



Jason Chang received his B.A. in geophysics from the University of California, Berkeley, in 2010. He joined SEP in autumn of 2011 and is currently working toward a Ph.D. in geophysics. He is a student member of SEG and AGU.



Jon F. Claerbout (M.I.T., B.S. physics, 1960; M.S. 1963; Ph.D. geophysics, 1967), professor at Stanford University, 1967. Emeritus 2008. Best Presentation Award from the Society of Exploration Geophysicists (SEG) for his paper, *Extrapolation of Wave Fields*. Honorary member and SEG Fessenden Award “in recognition of his outstanding and original pioneering work in seismic wave analysis.” Founded the Stanford Exploration Project (SEP) in 1973. Elected Fellow of the American Geophysical Union. Authored four published books and five internet books. Elected to the National Academy of Engineering. Maurice Ewing Medal, SEG’s highest award. Honorary Member of the European Assn. of Geoscientists & Engineers (EAGE). EAGE’s highest recognition, the Erasmus Award.



Robert Clapp received his B.Sc. (Hons.) in Geophysical Engineering from Colorado School of Mines in May 1993. He joined SEP in September 1993, received his Masters in June 1995, and his Ph.D. in December 2000. He is a member of the SEG and AGU.



Steve Cole is Manager of Integrated Analysis at OptaSense. He has worked in the oil and gas industry for over 30 years at companies including Fugro, Chevron, and 4th Wave Imaging, which he co-founded. He has a Ph.D. in geophysics from Stanford.



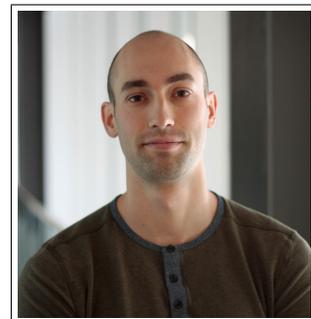
Taylor Dahlke is a sixth year student with SEP. He received his B.S. in civil engineering from the University of California, Berkeley in 2012, and joined SEP in July 2012. Currently, he is working towards a Ph.D. in geophysics with his research focused on applying level set methodologies to perform salt body segmentation. Taylor is a student member of SPE and SEG.



Nader Dutta retired from Schlumberger in May, 2015 after 15 years with the company. He held the positions of the (Global) Sr. Geophysics Advisor and Chief Geoscientist there. Prior to that, he worked for Shell, Arco, and BP for twenty five years at various capacities including both technical and management positions. Prior to joining the oil industry, he was a Sr. Research Fellow at the Harvard University. He has a Ph.D. in Physics from the University of California. He recently accepted a position at the Stanford University as a Visiting Scholar in Geophysics (Fall Quarter 2016.) and continues to advise the SEP, BPSM and SCRF / Rock Physics teams at the Stanford University. He can be reached at : Duttanc@Stanford.edu or Duttanc@outlook.com and via telephone: (+1) 832 274 1781.



Gabriel Fabien-Ouellet graduated from Laval University in 2012 with a B.S. degree in Physics and Engineering Physics and completed an M.Sc. in Earth Sciences in 2013. He received his PhD in Geophysics from INRS in Canada in 2017 and then joined SEP as a postdoc. Gabriel will take an adjoint professor position in Geophysics at Polytechnique Montreal in 2018.



Stuart Farris received his B.S. in Geophysical Engineering from the Colorado School of Mines with an added minor in Computer Science. After graduating in May of 2016, he joined the Stanford Exploration Project that summer and began his Ph.D. at Stanford University. He is interested in the unique problems associated with processing land seismic data, full waveform inversion, and utilizing parallel computing to enhance seismic imaging.



Joseph Jennings received his BS in Geophysical Engineering from Colorado School of Mines in 2014. Following graduation, he worked in land seismic processing R&D at ION Geophysical. In the summer of 2015, he joined the SEP as a PhD student where he hopes to pursue research in simultaneous source imaging and imaging of unconventional reservoirs.



Huy Le earned his B.S. degree in Geophysics from the University of Oklahoma in 2012 and his M.S. degree in Computational Geoscience from Stanford University in 2014. He is now a PhD candidate in Geophysics. His research interests include anisotropy, full waveform inversion, finite difference method, and high-performance computing.



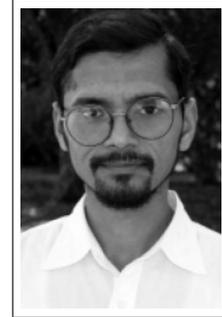
Stewart A. Levin was acting director of the Stanford Exploration Project during Jon Claerbout's 1993-4 sabbatical year. After a distinguished career in industry at Mobil and Halliburton, he has returned to Stanford as a senior research scientist in the Department of Geophysics.



Eileen Martin graduated from the University of Texas at Austin in 2012 with a B.S. in Mathematics and Computational Physics. She joined SEP in 2013. Since then, she has earned her M.S. in geophysics, and is working towards her Ph.D. in computational and mathematical engineering. During graduate school she has been supported in large part by the Department of Energy Computational Science Graduate Fellowship, as well as a Schlumberger Innovation Fellowship.



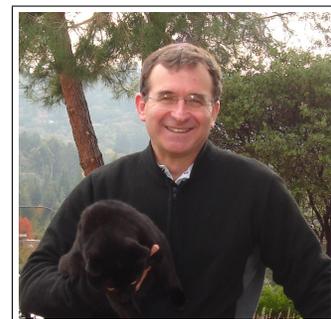
Tapan Mukerji is an Associate Professor (Research) in the Department of Energy Resources Engineering and the Department of Geophysics at Stanford University. He co-directs the Stanford Center for Reservoir Forecasting (SCRF), and is associated with the Stanford Rock Physics and Borehole Geophysics (SRB) and Stanford Basin and Petroleum System Modeling (BPSM) research groups. Tapan obtained his Ph.D. in Geophysics from Stanford University, in 1995, and M.S. (Geophysics) and B.S. (Physics) degrees from Banaras Hindu University, India. His research interests include rock physics, geostatistics, wave propagation, stochastic methods for quantitative reservoir characterization, time-lapse reservoir monitoring, and geomodeling applications. He was awarded the Karcher Award in 2000 by the Society of Exploration Geophysicists. He is a co-author of "The Rock Physics Handbook", "Quantitative Seismic Interpretation", and "The Value of Information in the Earth Sciences" all published by Cambridge University Press. In 2014 Tapan was awarded the ENI award - the so called "Energy Nobel Prize" - for pioneering innovations in theoretical and practical rock physics for seismic reservoir characterization.



Anshuman Pradhan is a Ph.D. candidate in Energy Resources Engineering at Stanford University. He received his M.S. (2014) in Applied Geophysics at Indian Institute of Technology (Indian School of Mines), Dhanbad during which he worked on seismic imaging algorithms. His current research interests include velocity modeling, basin modeling, rock physics and geostatistical and stochastic inverse methods.



Shuki Ronen is currently (2015) external and collaborative research manager of Dolphin Geophysical and a consulting faculty at SEP.



Rahul Sarkar graduated with a B.S and M.S from the Indian Institute of Technology, Kharagpur in 2011. After that he worked for Schlumberger for four years in the United States and Mexico. Since 2015, he is pursuing a M.S in the Institute for Computational and Mathematical Engineering at Stanford University.



Kaiwen Wang received her B.S. in Geophysics from Peking University in 2016. In fall 2016, she started her Ph.D. in Geophysics at Stanford University. She is currently working on applying machine learning algorithms to induced seismicity problems.



Siyuan Yuan got his Bachelors degree in Civil Engineering, Tongji University, China. After graduation in 2016, He was admitted to the master program in Civil and Environmental Engineering at Stanford. He joined SEP in the summer of 2016, and is currently working on SDASA-1 related research.



SEP ARTICLES PUBLISHED OR IN PRESS

- Almomin, A., and Biondi, B., 2012, Tomographic full waveform inversion: Practical and computationally feasible approach. SEG Technical Program Expanded Abstracts, 1–5.
- Almomin, A., and Biondi, B., 2013, Tomographic full waveform inversion (TFWI) by successive linearizations and scale separations. SEG Technical Program Expanded Abstracts, 1048–1052.
- Almomin, A., and Biondi, B., 2014, Preconditioned tomographic full waveform inversion by wavelength continuation. SEG Technical Program Expanded Abstracts, 944–948.
- Almomin, A., and Biondi, B., 2014, Generalization and amplitude normalization of tomographic full waveform inversion. SEG Technical Program Expanded Abstracts.
- Barak, O., Jaiswal, P., de Ridder, S., Giles, J., Brune, R. and Ronen, S., 2014, Six-component seismic land data acquired with geophones and rotation sensors: wave-mode separation using 6C SVD: SEG Technical Program Expanded Abstracts (Submitted).
- Chang, J., de Ridder, S., and Biondi, B., 2013, Power spectral densities and ambient noise cross-correlations at Long Beach: SEG Technical Program Expanded Abstracts **32**, 2196–2200.
- Chang, J., Nakata, N., Clapp, R.G., de Ridder, S., and Biondi, B., 2014, High-frequency surface and body waves from ambient noise cross-correlations at Long Beach, CA: SEG Technical Program Expanded Abstracts **33**.
- Chang, J., and B. Biondi, 2015, Rayleigh-wave tomography using traffic noise at Long Beach, CA: SEG Technical Program Expanded Abstracts **34**.
- Chang, J.P., de Ridder, S.A.L., and Biondi, B.L., 2016, High-frequency Rayleigh-wave tomography using traffic noise from Long Beach, California: *Geophysics*, 81, No. 2, B43–B53.
- Chang, J.P., 2016, Multicomponent ambient noise crosscorrelation at Forties: SEG Technical Program Expanded Abstracts **35**.
- Dahlke, T., Beroza, G., Chang, J., and de Ridder, S., 2014, Stochastic variability of velocity estimates using eikonal tomography on the Long Beach data set: SEG Technical Program Expanded Abstracts **33** .
- Dahlke, T., Biondi, B., Clapp, R., 2015, Domain decomposition of level set updates for salt segmentation: SEG Technical Program Expanded Abstracts **33** (Submitted).
- Grobbe, N., F.C. Schoemaker, M.D. Schakel, S.A.L. de Ridder, E.C. Slob, and D.M.J. Smeulders, 2012, Electrokinetic fields and waves: Theory, experiments, and numerical modeling. *Geophysical Research Abstracts*, Vol. **14**, EGU2012–10636.
- de Ridder, S., and B. Biondi, 2012, Continuous passive seismic monitoring of CCS projects by correlating seismic noise: A feasibility study. 74th Conference & Technical Exhibition, EAGE, Extended Abstracts (Accepted).
- de Ridder, S., and B. Biondi, 2012, Reservoir monitoring by passive seismic interferometry. SEG/SPE/AAPG Joint Workshop, La Jolla, CA, 24-29 June 2012 (Accepted).
- de Ridder, S., 2012, Ambient seismic noise correlations for reservoir monitoring. SEG Technical Program Expanded Abstracts **31** (Submitted).
- de Ridder, S., and B. Biondi, 2012, Continuous reservoir monitoring by ambient seismic noise tomography. SEG Technical Program Expanded Abstracts **31** (Submitted).
- Schoemaker, F.C., N. Grobbe, M.D. Schakel, S.A.L. de Ridder, E.C. Slob, and D.M.J. Smeulders, 2012, Experimental validation of the electrokinetic theory and development of seismoelectric interferometry by cross-correlation: *International Journal of Geophysics* (Accepted).

- Farghal, M.S., and Levin, S.A., 2012, Hunting for microseismic reflections using multiplets: SEG Technical Program Expanded Abstracts **31**.
- Guitton, A., Ayeni, G., and Esteban, D.A., 2012, Constrained full-waveform inversion by model reparameterization: *Geophysics*, **77**, No. 2, R117–R127.
- Guitton, A., and Esteban, D.A., 2012, Attenuating crosstalk noise with simultaneous source full waveform inversion: *Geophysical Prospecting* (In Press).
- Guitton, A., 2012, Blocky regularization schemes for full waveform inversion: *Geophysical Prospecting* (In Press).
- Halpert, A., 2012, Edge-preserving smoothing for segmentation of seismic images: SEG Technical Program Expanded Abstracts **31**, 1–5.
- Halpert, A., Clapp, R.G., and B. Biondo, 2014, Salt delineation via interpreter-guided seismic image segmentation, *Interpretation*: **2**, T79–T88.
- Arevalo, Humberto and Stewart A. Levin, 2014, Well and Seismic matching with ArcGIS and ProMAX via KML: Esri International User Conference Paper 901, San Diego, 15 July.
- Leader, C., and R. Clapp, 2012, Least squares reverse time migration on GPUs - balancing IO and computation: 74th Conference & Technical Exhibition, EAGE, Extended Abstracts (Accepted).
- Leader, C., and A. Almomin, 2012, How incoherent can we be? Phase encoded linearised inversion with random boundaries: SEG Technical Program Expanded Abstract **31** (Submitted).
- Levin, Stewart A. and Fritz Foss, 2014, Downward continuation of Mars SHARAD data: SEG Technical Program Expanded Abstract **33** (Submitted).
- Li, Y., Y. Zhang, and J. Claerbout, 2012, Hyperbolic estimation of sparse models from erratic data: *Geophysics* **77**, 1–9.
- Li, Y., P. Shen, and C. Perkins, 2012, VTI migration velocity analysis using RTM: SEG Technical Program Expanded Abstract **31**.
- Li, Y., Image-guided WEMVA for azimuthal anisotropy: SEG Technical Program Expanded Abstract **32**.
- R.E. Plessix and Y. Li, 2013, Waveform acoustic impedance inversion with spectral shaping: *Geophysical Journal International* **195**(1), 301–314, 2013.
- Li, Y., B. Biondi, D. Nichols, and R. Clapp, 2014, Wave equation migration velocity analysis for VTI models: Accepted for publication in *Geophysics*, 2014.
- Li, Y., M. Wong, and R. Clapp, 2014, Equivalent accuracy by fractional cost: overcoming temporal dispersion: Submitted to SEG 2014.
- Li, Y., D. Nichols, and G. Mavko, 2014, Stochastic rock physics modeling for seismic anisotropy: Submitted to SEG 2014.
- Li, Y., B. Biondi, R. Clapp, and D. Nichols, 2014, Rock physics constrained anisotropic wave-equation migration velocity analysis: Submitted to SEG 2014.
- Maharramov M., and Biondi, B., 2014, Joint full-waveform inversion of time-lapse seismic data sets. SEG Technical Program Expanded Abstracts, 954–959.
- Maharramov M., and Biondi, B., 2014, Joint 4DFWI with model-difference regularization. SEG-AGU Summer Research Workshop. Advances in Active+Passive “Full Wavefield” Seismic Imaging: From Reservoirs to Plate Tectonics.
- Maharramov M., and Biondi, B., 2015, Robust Simultaneous Time-lapse Full-waveform Inversion with Total-variation Regularization of Model Difference. EAGE Technical Program Expanded Abstracts (accepted).
- Maharramov M., 2015, Efficient Finite-difference Modelling of Acoustic Wave Propaga-

- tion in Anisotropic Media with Pseudo-sources. EAGE Technical Program Expanded Abstracts (accepted).
- Maharramov M., and Biondi, B., 2015, Simultaneous TV-regularized time-lapse FWI with application to field data. SEG Technical Program Expanded Abstracts (submitted).
- Martin E.R., B. Biondi, M. Karrenbach, and S. Cole, 2017, Ambient noise interferometry from DAS array in underground telecommunications conduits. EAGE Technical Programme Expanded Abstracts (accepted).
- Martin E.R., B. Biondi, M. Karrenbach, and S. Cole, 2017, Continuous subsurface monitoring by passive seismic with distributed acoustic sensors- the "Stanford Array" experiment. Proceedings of the First EAGE Workshop on Practical Reservoir Monitoring.
- Martin E.R., P. Wills, D. Hohl, and J.L. Lopez, 2017, Using Machine Learning to Predict Production at a Peace River Thermal EOR Site. Proceedings of SPE Reservoir Simulation Conference.
- Martin E.R., N.J. Lindsey, S. Dou, J.B. Ajo-Franklin, A. Wagner, K. Bjella, T.M. Daley, B. Freifeld, M. Robertson, C. Ulrich, 2016, Interferometry of a roadside DAS array in Fairbanks, AK. SEG Technical Program Expanded Abstracts.
- Martin E.R., J. Ajo-Franklin, S. Dou, N. Lindsey, T.M. Daley, B. Freifeld, M. Robertson, A. Wagner, and C. Ulrich, 2015, Interferometry of ambient noise from a trenched distributed acoustic sensing array. SEG Technical Program Expanded Abstracts.
- Ajo-Franklin, J., N. Lindsey, S. Dou, T.M. Daley, B. Freifeld, E.R. Martin, M. Robertson, C. Ulrich, and A. Wagner, 2015, A field test of distributed acoustic sensing for ambient noise recording. SEG Technical Program Expanded Abstracts.
- Li, Y., H. Yang, E.R. Martin, K. Ho, and L. Ying, 2015, Butterfly factorization. *Multiscale Modeling and Simulation*, 13, 714–732.
- Shen, Y., B. Biondi, R. Clapp, and D. Nichols, 2013, Wave-equation migration Q analysis (WEMQA): EAGE Workshop on Seismic Attenuation Extended Abstract 2013.
- Shen, Y., B. Biondi, R. Clapp, and D. Nichols, 2014, Wave-equation migration Q analysis (WEMQA): SEG Technical Program Expanded Abstracts.
- Li, Y., Y. Shen and P. Kang, 2015, Integration of seismic and fluid-flow data: a two-way road linked by rock physics :77th EAGE Conference and Exhibition.
- Wong, M., and Ronen, S., and Biondi, B.L., 2012, Joint imaging with streamer and ocean bottom data. SEG Technical Program Expanded Abstracts, 1–5.
- Wong, M., and Ronen, S., and Biondi, B.L., 2012, Imaging with multiples using linearized full-wave inversion. SEG Technical Program Expanded Abstracts, 1–5.
- Wong, M., 2013, Handling salt reflection in Least-squares RTM SEG Technical Program Expanded Abstracts, 3921–3925.
- Wong, M., B. Biondi, S. Ronen, C. Perkins, M. Merritt, V. Goh, and R. Cook, 2014, Robust least squares RTM on the 3D Deimos ocean bottom node dataset SEG Technical Program Expanded Abstracts, (Submitted)
- Wong, M., B. Biondi, and S. Ronen, 2014, Imaging with multiples using least-squares reverse-time migration. *The Leading Edge*, (submitted)
- Huy Le and Stewart A. Levin, 2014, Removing shear artifacts in acoustic wave propagation in orthorhombic media: SEG Technical Program Expanded Abstracts, 486–490.
- Huy Le, Biondo Biondi, Robert G. Clapp, and Stewart A. Levin, 2015, Using a nonlinear acoustic wave equation for anisotropic inversion: SEG Technical Program Expanded Abstracts (Submitted).

